

Gap Analysis between State of Practice & State of Art Practices in Agile Software Development

Abdul Rauf

Department of Computer Science, College of Computer & Information Sciences, Al-Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh, KSA, Saudi Arabia
rauf.malik@ccis.imamu.edu.sa

Mohammed AlGhafees

Department of Computer Science, College of Computer & Information Sciences, Al-Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh, KSA, Saudi Arabia
maalghafees@ccis.imamu.edu.sa

Abstract— Agile software development can be considered as a development methodology which suits the situations where requirements are changing frequently. Most of the literature about agile is specific to one or two practices. Lack of a comprehensive discussion on all the commonly used agile practices in literature and their adoption results by industry is very much evident. This work is an ongoing effort, where we are going to discuss some of the literature, we have gone through to solicit common agile practices, score them on the basis of frequency of referral and then we have tried to get an opinion from industry regarding our findings. Our aim is to have a comprehensive review of all the literature presented for Agile in last decade and have an analysis of these results based on feedback from industry of at-least 2-3 continents. In this short paper, we are presenting some of our findings and we hope to extend our work gradually. (*Abstract*)

Keywords— *Software development; agile methods; agile practices; XP, Scrum (key words)*

I. INTRODUCTION

Software development is considered to be successful if it is completed within time and budget while satisfying customers' needs. Software development is always associated with ambiguity due to changing requirements, change in technology or early delivery, hence plan-driven, engineering based methodologies are not suitable for such dynamic environment [1][2]. Agile development methodologies provides flexibility to handle changing requirements, improved communication and coordination mechanisms, and improved quality while enhancing the time-to market speed [3]. Many empirical studies report the successful adoption and execution of agile methods [4]. Software companies are not following any agile method as is; rather they are using the most suitable agile practices. Hence they should have the knowledge of the benefits and issues associated with the use of each practice. Empirical studies on agile methods are mainly intended towards the findings of adopting any specific agile method with the benefits and challenges in adoption. The impact of applying each practice of the method is not addressed in detail. Also the literature addressing the benefits and challenges of each practice is scattered, hence it is difficult to investigate each practice with respect to the experiences associated with their usage. In order to address this gap, an industrial survey is conducted to capture the experience of practitioners to

investigate the agile practices regarding their benefits and challenges.

The purpose of this study is to accumulate the experience of industry practitioners in order to better understand the benefits and challenges of applying agile practices in projects. The Agile methods: XP and Scrum are found to be the most widely used methods [5] [6] [7], hence the practices of XP and Scrum are considered for this study. By benefits we mean that how the application of agile practices in projects can be valuable. Similarly by challenges we mean the factors which create hurdles for implementing agile practices in projects. To the best of our knowledge, no research effort has been done with the focus on accumulating benefits and challenges associated with the use of agile practices. The industrial survey investigated the strength of benefits of agile practices by posing close-ended questions and explored the challenges of agile practices by posing the open-ended questions. 79 responses are collected from 45 companies to investigate the strength of advantages of the practices and exploring their challenge. Hence the main contribution of this work is to provide guidance for the software teams aiming to apply the agile practices by showing the merits and demerits of each practice.

Findings indicate that besides agreeing with the benefits of agile practices, the practitioners have identified many issues with agile practices; hence more research efforts should be directed to examine the practices critically. The results indicate that the issues with agile practices are elevated when applied in large companies as agile is intended to be implemented in small companies with team size less than 10. Survey results indicate that the agile practices are being used by larger teams as well hence new issues with agile practices arise which need to be resolved to encourage the use of agile practices.

In section 2 discusses the related literature, section 3 details the research methodology describing the research questions, research aims, context, data collection and data analysis. Section 4 demonstrates the results to answer the research questions. Section 5 contains conclusion and future work.

II. RELATED WORK

A large number of studies have been conducted on agile methodologies with a focus on agile transformation; agile adaptation, benefits and challenges of applying agile methodologies but there is a lack of empirical studies with a

true focus on investigating agile practices in detail. Here is a list of empirical studies assessing some agile practices in detail. These studies are mostly based on application of agile methods in large organization, hence the challenges of applying agile practices is also addressed besides their benefits.

Chen et al. [8] performed a case study at Intel to investigate how agile methods can be employed at large organizations. The study highlighted many issues as a result of applying agile practices in large companies.

Begel and Nagappan [9] performed a survey at Microsoft to evaluate their experience of adopting agile methodologies. 487 respondents contributed in the study to make it a valuable study. It was found that Scrum and XP was being used at Microsoft. This study provided valuable findings in evaluating the widely used, most effective and most problematic agile practices while focusing of agile adoption challenges in large companies. The study was a good initiative in directing the research efforts to focus on the challenges of agile practices besides their benefits.

Fruhling et al. [10] investigated the challenges faced in agile transformation in a government organization. Case study was conducted to explore the challenges faced in adopting agile practices. The study was good initiative in exploring the challenges but the results cannot be generalized.

Salo and Abrahamsson [11] focused on applying XP and Scrum practices in embedded software development. The study provides generalized results on the application of agile methods without detailed exploration of result of applying each practice individually. The results cannot be generalized as the study is limited to the application of agile methods in embedded industry.

Petersen and Wohlin [12] investigated the experience of applying of XP and scrum in a large organization. They conducted a case study at Ericsson to investigate the benefits and challenges of agile adoption. The study was a good effort towards investigating the agile practices in detail by comparing the literature findings with case study findings to generalize the literature discoveries.

M. Laanti et al. [13] conducted a large survey comprising of 1000 respondents from different geographic locations regarding the agile adoption experiences and views. The problematic areas and challenges were discussed but the challenges with a focus on adopting each agile practice was not covered in detail.

III. RESEARCH METHOD

This research is conducted using industrial survey is conducted to assessing the strength of benefits and exploring issues related to agile practices adoption. We have selected survey as it is the suitable research method for getting multi-disciplinary view and to draw generalized conclusions about the benefits and issues of agile practices [14].

A web-based industrial survey is conducted as it is convenient for the respondents instead of filling a paper survey.

Also the data obtained as a result of web-based survey is easier to manage and compiled. The questionnaires consist of multiple: (1) demographic information, (2) benefits of practices section and (3) challenges of practices. Demographic information includes designation and experience of the respondent to get assured that the respondents are the right people to get the desired information. Respondents are asked whether they follow agile development in their companies, the name of agile methodology being used to make sure that they are using agile methodologies in their organizations. The strength of benefits of agile practices is assessed by asking close-ended questions with 5-level likert scale (1= Strongly Disagree, 2= Disagree, 3=Neutral, 4=Agree, 5=Strongly Agree). Open-ended questions are used to explore the issues of agile practices experienced by industry practitioners.

Software companies do not follow any specific agile method completely; rather they use a mix of agile practices. Hence it is useful to capture the views of industry practitioners behind using the specific practices and abandoning the other ones. The experience with agile practices varies from project to project and team to team, hence it is worth accumulating the benefits and challenges associated with individual practice usage. This research motivation leads us to the following research question:

What are challenges and benefits of agile practices in state of practice? This question is addressed by conducting an industrial survey in software industry of Pakistan including the main cities: Islamabad, Rawalpindi, Lahore and Karachi. In this way, a current view of issues and benefits faced by industry is presented.

The sample for this study consists of 79 responses from 45 software companies in Pakistan. We conducted an online industrial survey over a period of 2 months.

IV. RESULTS

Among the survey respondents 33% represent managers including (Project managers, team leads), 52% represent development staff (developers, architects, software engineers and test analysts) and 15% represent other staff. The average experience of survey respondents in agile software development is 2.9 years having minimum experience of 0.5 years and maximum 7 years with standard deviation 1.7. Team size of respondents is also enquired in order to get insights for the trends for adopting agile practices in small and large teams. It is found that agile practices are widely applied in projects with small team sizes. 28% of respondents have reported their team size greater than 15.

Survey results show that Scrum is the most widely used agile method as 57% of survey respondents have reported Scrum as their software development methodology. 27% respondents use the practices of both XP and Scrum. Only 5% respondents have reported to use XP solely. This trend of agile methods usage appears the same as reported in literature studies [15]

The extent of adoption of agile practices is enquired by providing a list of practices so that the respondents can select

the practices being used in their companies. The prioritized list of agile practices according to their extent of usage is shown in Table 1. Sprint planning meeting, refactoring and daily Scrum meeting are the most widely used agile practices. The least used practices are acceptance testing, pair programming and onsite customer.

A. Benefits of Agile practices

The benefits of agile practices obtained from literature are categorized based on commonality to create a short list of benefits related to each practice. The strength of the benefits is evaluated by asking close ended questions using a likert scale:

(1= Strongly Disagree, 2= Disagree, 3=Neutral, 4=Agree, 5=Strongly Agree).

The benefits of delivering in iterations/sprints have received the highest acceptance from industry as shown in Figure 1. Iterative development has also reported to be the most widely used agile practice, hence respondents also agree with its benefits.

The sprint planning meeting and refactoring are a widely used agile practice, hence a large number of respondents also agree with their benefits as shown in Figure 2 and Figure 3. The respondents giving neutral and negative findings avoid the practice of refactoring as it needs care and coordination. Refactoring helps in maintaining code quality by eliminating errors at early stages.

The responses for accepting the benefits of daily Scrum meetings results have shown positive results. Only 1% of respondents have disagreed with the benefits and 16% respondents have shown. The neutral responses are due to the fact that some respondents are not using this practice as they perceive the daily conduct of meetings depends on the complexity of projects.

Sprint review meeting, product backlog, incremental design, collective code ownership and test-driven development (TDD) are not widely being used in industry but still the practitioners strongly agree with their benefits.

Burndown charts are used by only 54% of industry but still the strength of positive responses is larger. Only 3% of respondents have shown negative responses. 15 % responses are neutral. Although 52% of participants use continuous integration, still more are inclined to accept the benefits. Negative responses are 3% and neutral responses are 28%. The adoption rate is lesser due to the extensive care and control required for using the practice.

Sprint backlog and collective code ownership are used by only 48% and 43% but still respondents recognize their benefits. Collective code ownership has received only 5% negative response and 23% neutral response.

The practice of onsite customer is used by only 34% of respondents. This practice is more suitable in complex projects hence it is not widely used as the neutral responses against the practice is 31%. Negative response is only 5% which shows that instead of limited usage of the practice, respondents still agree with the benefits associated with its usage.

The least used practices in industry are: pair programming and acceptance testing. 29% of respondents use pair programming (PP) and 21% of respondents use acceptance testing. Besides least usage in industry, the percentage of negative responses against the practices is also high. Only 3% of respondents agree with the benefits of PP which complies with its low usage in industry as shown in Figure 4. PP is preferred only in complex situations.

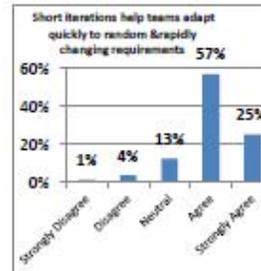


Figure 1. Benefits of Short Iterations



Figure 2. Benefits of Sprint Planning Meeting

B. Challenges of Agile practices

Challenges of each agile practice are explored by asking open-ended questions. The results obtained against each practice are grouped into relevant themes to create a list of challenges against each practice. The data was tabulated against each practice by assigning them a unique ID. The list of issues against each practice is shown in Appendix A.

Challenges reported by the practitioners indicate the reason for abandoning the agile practices instead of realizing their benefits. The challenges should be addressed in detail in order to devise the strategies and suggestions for overcoming the challenges.

V. CONCLUSION

Software development organizations are not following any agile methods completely. They adopt a mix of agile practices according to their projects. The motivation behind the study is to provide industrial insights regarding the use of different agile practices, instead of studying any specific agile method. An industrial survey is conducted to gain insights regarding the application of agile practices in practice. The strength of benefits against each agile practice are investigated and challenges against each agile practice are explored. First the extent of agile practices adoption in industry is investigated in order to identify the widely accepted and least used practices. A prioritized list is provided to analyze the extent of their application in industry. Second, the practices are evaluated with respect to their extent of adoption and perceived usefulness in industry. Although the practices are not adopted widely, but the respondents still agree with their benefits which show that they are willing to adopt the practices. The reason for abandoning some agile practices is the challenges faced by the organizations in adopting them. Third, the challenges and issues faced while adopting each practice are explored. A list of challenges against each practice is provided in order to gain insights for the factors hindering the use of agile practices.

It would be interesting to carry out another survey for a larger population to find any new challenges of the agile practices. The findings can be generalized by comparing the survey results with the literature findings. The issues with agile practices can be categorized in context dependent issues and general issues. This work is an initiative to direct the research efforts towards exploring and eradicating issues with agile practices in detail. Further work is required to evaluate the strength of evidence for agile practices.

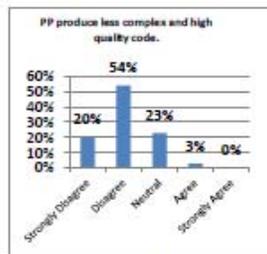
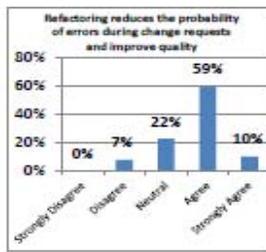


Figure 3. Benefits of Refactoring

Figure 4. Benefits of PP

TABLE 1. %AGE OF AGILE PRACTICES USAGE

	Agile Practices	%age usage
1	Iterative Development	90
2	Sprint Planning Meeting	76
3	Refactoring	73
4	Daily Scrum Meeting	71
5	Sprint Review Meeting	63
6	Product Backlog	62
7	Incremental Design	61
8	Single Code Base	60
9	Iterations/Sprint	56
10	Burndown charts	56
11	Test Driven Development (TDD)	56
12	Continuous Integration	52
13	Sprint Backlog	48
14	Collective Code Ownership	43
15	On-Site Customer	34
16	Pair Programming (PP)	29
17	Acceptance Testing	21

REFERENCES

[1] Serena, An introduction to Agile Software Development, [online] June 2007. [Cited: January 18, 2012.] www.serena.com/docs/repository/solutions/intro-to-agile-devel.pdf

[2] J. Cho, Issues and Challenges of Agile Software Development With Scrum, *Issues in Information Systems*, VOL IX, No. 2, pp. 188-195, 2008.

[3] P. Abrahamsson, J. Warsta, M.T. Siponen, J. Ronkainen, *New Directions on Agile Methods: A Comparative Analysis*, Proceedings of 25th International Conference on Software Engineering, pp. 244 –254, 2003.

[4] Abrahamsson, P., Salo, O., Ronkainen, J., 2002. *Agile Software Development Methods – Review and Analysis*. VTT Publications 478. <http://www.inf.vtt.fi/pdf/publications/2002/P478.pdf> (accessed December 2005).

[5] Scrum Alliance, *Scrum Alliance Membership Industrial survey Shows Growing Scrum Adoption and Project Success*, [online] September 28, 2007. [Cited: January 18,2012.] Available: www.scrumalliance.org

[6] Versionone, *State of Agile Survey:2011, The State of Agile Development*, [online] june 2011, [Cited: January 20, 2012.] http://www.versionone.com/state_of_agile_development_survey/10/

[7] Versionone, *State of Agile Survey:2011, The State of Agile Development*, [online] june 2012, [Cited: January 20, 2012.] http://www.versionone.com/state_of_agile_development_survey/11/

[8] J. Q. Chen, P. Dien, B. Wang, and D. R. Vogel, "Light-Weight Development Method: a Case Study," presented at Service Systems and Service Management, 2007 International Conference on, 2007.

[9] Begel, A. and N. Nagappan, "Usage and Perceptions of Agile Software Development in an Industrial Context: An Exploratory Study," presented at Empirical Software Engineering and Measurement, 2007. ESEM 2007. First International Symposium on, 2007.

[10] A. Fruhling, P. McDonald, and C. Dunbar, "A Case Study: Introducing eXtreme Programming in a US Government System Development Project," presented at Hawaii International Conference on System Sciences, Proceedings of the 41st Annual, 2008.

[11] O. Salo and P. Abrahamsson, Agile methods in European embedded software development organisations: a industrial survey on the actual use and usefulness of Extreme Programming and Scrum, *IET Software*, Vol. 2, No. 1, pp. 58-64, 2008.

[12] K. Petersen and C. Wohlin, "A comparison of issues and advantages in agile and incremental development between state of the art and an industrial case," *Journal of Systems and Software*, vol. 82, pp. 1479-1490, 2009.

[13] M. Laanti, O. Salo, and P. Abrahamsson, "Agile methods rapidly replacing traditional methods at Nokia: A survey of opinions on agile transformation," *Information and Software Technology*, vol. 53, pp. 276-290, 2011.

[14] P. Shari Lawrence and A. K. Barbara, "Principles of industrial survey research: part 1: turning lemons into lemonade," *SIGSOFT Softw. Eng. Notes*, vol. 26, pp. 16-18, 2001.

[15] Zaigham Mushtaq and M. Rizwan Jameel Qureshi, "Novel Hybrid Model: Integrating Scrum and XP", *IJITCS*, vol.4, no.6, pp.39-44, 2012.

APPENDIX: CHALLENGES OF AGILE PRACTICES REPORTED FROM INDUSTRIAL SURVEY

Iterative Development
S1. Clients' unavailability or Lack of cooperation
S2. Clients' willingness to get the system in pieces.
S3. iterative development for large projects result in scope management issues
Incremental design
S1 Integration and prioritization of features
S2. Difficult to manage for large projects
S3. Client involvement for prioritizing requirements, hence Timelines can suffer
Time Boxed Iterations
S1. Time limitations enhances pressure on developers which reduces productivity
S2. The items not completed in an iteration are moved to next iteration with a result in delayed schedule
S3. High planning and control is required
Sprint Planning meeting
S1. High responsibility on PM for involving client and Team.
S2. Team availability for sparing time for meeting in tight schedules as meetings gets long.
S3. Finalization of points due to suggestions conflicts as each member has different view
Daily Scrum meeting
S1. Hard to manage in terms of finalization of points, time, prioritizing issues, conducting with updated status.
S2. Long meetings wastes developer time and divert mind of developers from their coding tasks.
S3. Daily reporting discourages developers as there is not enough work to show daily. .
Sprint Review meeting
S1. Team member unavailability
S2. Developers are unwilling
S3. Take extra time of project
Product backlog
S1. Difficult to manage/update the product backlog for rapid changes.
S2. Time consumption in planning for product backlog in tight deadlines.
Sprint backlog
S1. Continuous updating requires extra resources and time
S2. Backlog take time of other tasks due to tight deadline and this cycle continues.
S3. People sometimes add new stories to backlog during the sprint
Burn down charts
S1. Difficult to update daily as some tasks are partially done/dependent tasks
S2. In tight deadlines, it is cumbersome to update daily
S3. Daily updation can take time of other tasks
Refactoring
S1. Not possible every time because it is time consuming and take extra time of project
S2. Stability of code can be affected
S3. Risk of external system crash
Test-Driven development & Unit Testing
S1. Developers assume it time consuming as they use to test only some common errors and thorough testing.
S2. To meet tight deadlines, this is often ignored
S3. Developers think it as hectic and extra work
Pair programming
S1. Coordination issues in developers
S2. Problem occurs if one of those or both leave the organization
S3. Takes more time, if two developers work on same machine
Collective code ownership
S1. Code is not well-designed as no one is responsible individually
Continuous integration
S1. Time overhead to integrate everyday
S2. Close communication and coordination is required
S3. Documented design is required
Onsite Customer
S1. Client unwillingness /client unavailability
S2. Communication gaps from customers results in delayed feedback.
S3. Continuous change request leads to scope creep