Findings on System Reliability Improvements in Agile Environments

Lucas S. M. Lago, Marvin F. da Silva and Edison Spina

***Abstract* —** Software development can be considered a socio-technical system, and the choice of practices can be viewed as the design of that system. The choice of agile practices focused on human factors can change the environment where software is developed and influence the performance of the members of the team. To evaluate the impact of the choice of five practices a questionnaire was applied to 125 developers in Brazil. These practices were related to guidelines of the Ecological Interface Design, a framework used in the design of systems with reduction of human error. The use of the guidelines was related to the increase in the perception of all the guidelines with $p < 0.01$. This result goes against the opinion of many experts and shows that agile practices like "daily meeting" and "quick design sessions" can be used to increase the reliability of the software developed by the team, even without the adoption of a complete agile methodology.

***Keywords* —** agile practices, Ecological Interface Design, human factors, reliability, software engineering, socio-technical system.

I. Introduction

SOFTWARE Engineering has relied on anecdotal evidence and the opinion of experts in the study of software development practices [1] [2].

In 2001, the flag carrying the agile values was stuck by its founders [3] and three quarters of its values lay on better interpersonal communication and people relationship [4] that can reduce human communication problems that lead to knowledge-based errors due to insufficient or incorrect information [5].

Agile practices are not considered useful in development of high reliability software [6], but are still largely adopted by the industry [7]. An agile team can be considered a socio-technical system [8] and the practices can be viewed as the interfaces of this system.

The SEMAT initiative was initiated in 2009 with the main goal of changing the way people work with software development methods. One of the first results of this initiative was the KERNEL a tool that can be used in the research of software methods [9].

Human behavior is influenced by the environment [10]. Software developer errors can be correlated to changes in the environment [11]. These errors are the root cause of most of software failures [12].

There are two approaches to dealing with human error. One personal approach – more common, that focuses on the person who committed the mistake, and one systemic approach that see errors as consequences of human fallibility [13].

One implementation of this systemic approach is the Ecological Interface Design (EID) created by Rasmussen and Vicente [14].

The guidelines of Ecological Interface Design were used to evaluate if agile practices are a framework for the creation of an environment where software can be developed with low probability of human error and trying to show that agile development brings reliability unlike what is defended by others. A questionnaire was designed to identify if developers perceived the guidelines on their workplace.

The ESSENCE-kernel domains of the SEMAT initiative were used in order to map and evaluate which practices have influence on the human factors of software development.

II. Agile Practices and Reliability Concerns

Increase the reliability of software systems without increased cost is one of the concerns of software engineering [15]. Since the major difference between software and other engineering artifacts is that software is pure design, the threats to software dependability can be traced to software faults and human-error on the development phase of the software lifecycle are responsible for 60% of the faults [12], [16].

Several aspects of the work environment such as frequency of interruptions, office space and feedback have influence on software development performance and number of errors in the developed software [17].

This work was supported in part by the Institute for Technological Research of São Paulo under project 879505A and in partnership with Centro de Estudos Sociedade e Tecnologia (CEST).

L. S. M. Lago is with the Institute for Technological Research, Av. Prof. Almeida Prado 532 Cid. Universitária. 05508-901 São Paulo/SP/Brasil (+5511-3767-4128; llago@ipt.br).

M. F. Silva is with CEST and University of São Paulo (marvin@usp.br).

E. Spina is with CEST and University of São Paulo (spina@usp.br).

This fact is not considered in the most common processes to addressing the faults in software [18].

Agile practices are based on creating an environment where the development team can be productive and create high quality software [3]. These practices are being largely adopted by the industry [7] but are still heavily criticized in relation to the development of reliable software [6].

Although the experts agree that agile practices are not suitable for safety critical systems there is a lack of studies on software practices [2] and reliance on anecdotal evidence [1]. Therefore, there is a need to establish means to measure the influence of agile practices in software reliability.

III. Research Method

To measure the impact of software practices on the reliability of software directly, there would be a need to create identical software with two different sets of practices, which is highly expensive.

Therefore, the impact of the practices have to be measured indirectly, and we need a set of measurable parameters that are related to high reliability systems.

A questionnaire was designed to evaluate the perception of guidelines related to software reliability.

To build this questionnaire we choose the major agile approaches based on a large study made by Version One and described all the practices of this methods. This also enable us to map each separated practice in its respective SEMAT domain and EID guidelines.

A. Population

In June 2014, 136 software developers were contacted, and 125 took part of the study. No exclusion criteria was applied on the subjects.

The subjects had different works experience and worked on companies of distinct sizes as can be seen in Figure 1 and 2.

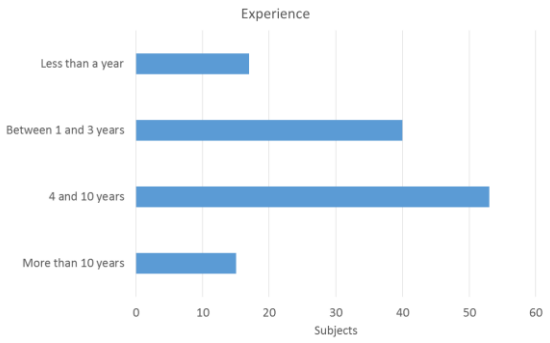

Figure 1 – Experience of subjects on software development.

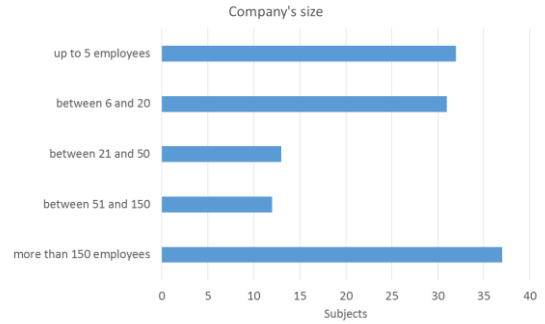

Figure 2 – size of the companies where the subjects work

On the use of agile methods of Software Development, 43.2% of the subjects did not use any method in his daily routine.

B. Parameters for the development of systems with increased reliability

Software development can be viewed as a sociotechnical system [19]. This system will have a number of different interfaces, including the ones involving the developer.

The design of the interfaces influences in the reliability of the system, through the reduction of human errors [14].

The EID (Ecological Interface Design) is an interface design based on ten guidelines. The application of these guidelines in the design of complex system's interfaces have shown positive effects on reliability [20] [21].

The guidelines are:

1. Make the limits of acceptable performance visible to the operators, while the effects are still observable and reversible;
2. Provide the actors with feedback on the effects of actions;
3. The interface should be designed so that there is a consistent and unique mapping from the signs that defines cues for action and the symbols that describe how the process functions;
4. Supply the actors with tools to make experiments and test hypothesis without having to do this on a high risk situation;
5. Make available overview displays by which "free-running" routines can be monitored by fringe consciousness;
6. Make the cues for action integrated patterns based on defining attributes and serving, at the same time, as symbolic representation necessary for functional monitoring of performance;

7. Support memory with externalization of the effective mental models;
8. Use the available data to develop consistent information transformation concepts for data integration. This will enable the interface to present information at the level which is most appropriate for decision-making;
9. Present information embedded in a structure that can serve as an externalized mental model, effective for the kind of reasoning required by the task;
10. Support of memory of items, acts, and data that are not part of an integrated gestalt can be useful.

These guidelines are not for a specific system and can be applied to any interface design. The choices of practices made by the software team are a form of design of the interfaces of the software development system that they are an integral part.

Therefore, the perception of the guidelines from the Ecological Interface Design can be used as parameters to predict the reliability of the system. We will analyze the impact of several different practices on those guidelines.

C. *Practices selection*

This work analyzed only practices increase reliability through the environment, to determine what are the most common practices that do not influence directly the reliability of the developed software a classification of the practices were classified in three different frameworks:

1. Using the works on Software Engineering Reliability we determined which practices relate to Reliability directly [22];

2. Using the Alphas of the ESSENCE kernel we determined which practices relate to humans factors of the software development [23];

3. And using the guidelines of the EID we determined which practices relate to the creation of an environment that produces software with better reliability [14].

All the classifications used references from the literature plus the knowledge of the author of each practice.

The practices selected to the classification were all of the practices mentioned in the documentation of the Agile Alliance [24], together with the practices mentioned in the State of Agile Development Survey [25].

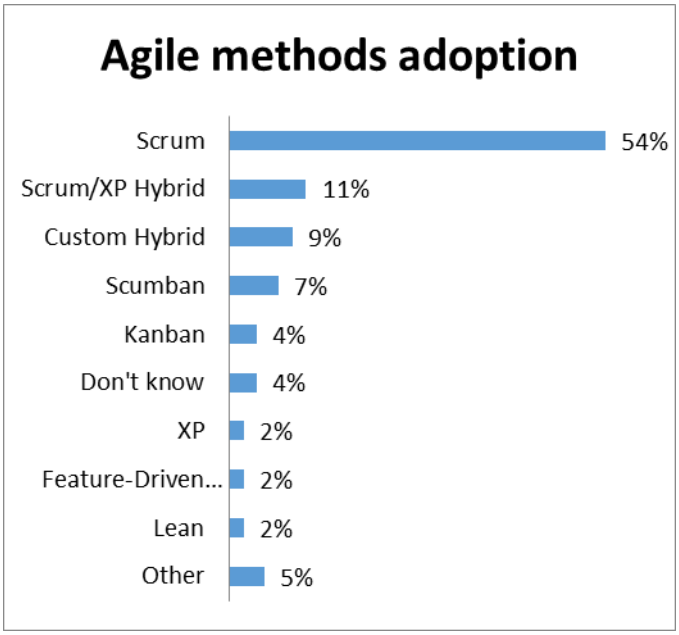

Figure 3 – Agile adoption as seen on the State of Agile Survey.

D. *Questionnaire design*

We applied an online questionnaire to 136 software developers to determine the relation of five agile practices with the increased perception of the guidelines of the EID.

The questionnaire had the following structure:

- The first section had questions that would determine the population of the study;
- The second section, evaluated the use of each of the practices selected;
- The third question looked on the perception of the guidelines;
- The fourth and fifth section, we asked for the opinions of the developers about the relation of agile practices and software reliability.

The questionnaire was developed using the tool from Google Drive® and the data was analyzed using the software SigmaPlot 13.0®.

The questionnaire used Likert scales (where one means low and seven high) to access the use of each of the practices and the perception of the guidelines.

E. Statistical Analysis

For each practice, two groups were created, one with the highest scores on the Likert scale for that practice, and a second group with the lowest score on the same scale. Each group had 31 subjects.

The results of the answers for the perception of the guidelines from the EID that were related to each practice were compared using the Mann-Whitney U test, using 0.01 as the threshold for the null hypothesis.

On a secondary comparison, the sum of the answers from practice use was used to create two groups using the same rules, high usage group and low usage group. The answers from these groups on perceptions of the guidelines were compared using the same test.

IV. Results

A. Practices selection

To select the practices that would collaborate with software reliability indirectly through human factors, we using the literature selected the practices that were:

1. Not directly related to Software Engineering Reliability;
2. Related to the following alphas of the Essence: work, way of working and team;
3. Directly related to one of the guidelines of the EID.

Five practices met that criteria: daily meetings, retrospectives, task boards, quick design sessions and niko-niko calendar. This practices are defined briefly on Table 1.

Table 1 – Definition of the practices analyzed

Practice	Definition
Daily Meeting	A daily meeting with of around fifteen minutes to share project updates [26]
Retrospectives	Informal meetings made by the team with the objective of changing or adding practices to the development process [27]
Taskboards	Place where the team keeps the situation of the project [28]
Quick Design Sessions	Two or more developers meet for a brief discussion to decide on architectural questions [29]
Niko-niko Calendar	Each member of the team draws a smile representing his humor for that day [30]

This practices relate directly to several guidelines from the Ecological Interface Design as can be seen on Table 2.

Table 2 – Relation of practice to guidelines from the Ecological Interface Design

Practice	Guidelines
Daily meeting	2
Retrospectives	2, 9
Quick Design Sessions	3
Taskboards	3, 5, 6
Niko-niko Calendar	5

B. Usage of practices

The usage of each practice is shown in the **Error! Reference source not found.**, and the developers that answered with scores under the lower quartile were compared to those who answered higher than the upper quartile in order to compare the effect of each practice on the guidelines.

Table 3 – Scores for each practice in the Likert scale of the questionnaire

Practice	Lower quartile	Median	Upper quartile
Daily meeting	0	3	6.5
Retrospectives	0	2	5
Quick Design Sessions	0	3	5
Taskboards	1	5	7
Niko-niko Calendar	0	0	0

The **Error! Reference source not found.** shows the result for the comparison of medians using the Mann-Whitney U test for the pair practice/guideline presented in the first column. The p-value is representing the confidence in which we assert that the medians are different.

Table 4 – Median from the high usage group and low usage group of each practice for the perception of each guideline

Practice / Guideline number	*Median (High usage group)*	*Median (Low usage group)*	*p*
Daily meeting / 2	6	3	< 0.01
Retrospectives / 2	6	3	< 0.01
Retrospectives / 9	5	2	< 0.01
Quick Design Sessions / 3	6	3	< 0.01
Taskboards / 3	6	3	< 0.01

Taskboards / 5	10	2	< 0.01
Taskboards / 6	6	3	< 0.01
Niko-niko Calendar / 5	9	3	< 0.01

In a secondary analysis the Likert scale score for each of the practices was added, this allowed for the evaluation of the impact of the set of practices instead of each practice independently. The result was a lower quartile of 5.5, a median of 14 and an upper quartile of 21.

Two groups of 31 developers were compared, the group with scores above the upper quartile and the ones with scores lower than the lower quartile.

The Mann-Whitney U test was applied to verify differences on the groups for every practice analyzed. The results are shown in **Error! Reference source not found.**.

Table 5 – Median from high usage group and low usage group of all practices for the perception of each guideline

Guideline number	*Median (High usage group)*	*Median (Low usage group)*	*p*
2	6	3	< 0.01
3	6	3	< 0.01
5	10	6	< 0.01
6	6	3	< 0.01
9	5	3	< 0.01

V. Discussion

The studied practices show to play a role in increasing environment factors related to them. In all analysis, the group with higher use of agile practices showed a better result in the perception of guidelines from the EID.

This positive result goes against the opinion of specialists that state that agile practices are not capable of developing software systems with high dependability and reduce software quality [6].

Some authors point out that the mechanisms used by agile methods to control quality are not adequate to the development of critic systems. These practices are not considered on this present study, but the impact of agile practices on the environment show that the adoption of these practices have positive effects on software quality [1].

Considering the set of practices studied there was a positive result for all the guidelines considered, showing that the use of agile practices can: increase the feedback, clear the perception of the state of the project and other characteristics. Environmental factors are historically related with developer performance. These agile practices solve an existent fault on traditional methods to increase reliability – the lack of human factors consideration [17], [18].

This study shows that there is no loss of quality with the use of agile practices, instead there is an increase of the adherence of guidelines that reduce errors on systems, this loss of quality is one of the major concerns in the adoption of agile methods [25], [31].

VI. Conclusions

The individual practices in this study show a positive effect on the perception of different guidelines from the Ecological Interface Design. The Ecological Interface Design has shown positive results in the reduction of human error in several different systems [32].

This shows that agile practices can foster an environment where the development of high reliability software, that goes against the belief of many experts [6].

The study shows that the practices – even individually – are effective in bringing benefits on each of the guidelines considered in the study, and that the use of agile practices can help increase the reliability of the developed software.

This shows that the adoption of agile practices can have a positive effect on the software developed even if there is no adoption of the agile methodology that fosters the practice. There can be bundle of practices that can be applied to any company independently of the methods chosen.

Furthermore, it shows that changes in workspace culture have an impact on the quality of the software developed by that team, strengthening the notion that software development should be viewed as a sociotechnical endeavor.

VII. Future work

Although it was shown to work on other environments, the guidelines of the Ecological Interface Design to reduce human errors were never applied in a software development endeavor and further empirical studies are needed to evaluate that hypothesis.

Studies with a different set of practices can help identifying which have the most impact on creating high quality software and reinforce the hypothesis that agile practices increase software quality.

References

[1] D. Turk, R. France, and B. Rumpe, "Limitations of agile software processes," *... Agil. Process. Softw. ...*, 2002.

[2] P. Johnson, M. Ekstedt, and I. Jacobson, "Where's the Theory for Software Engineering?," *IEEE Softw.*, vol. 29, no. 5, pp. 96–96, Sep. 2012.

[3] K. Beck, M. Beedle, and A. Van Bennekum, "Manifesto for agile software development," 2001. [Online]. Available: http://agilemanifesto.org/. [Accessed: 03-Sep-2013].

[4] L. Yong, J. G. Kelang, and K. Lumpur, "Communication Complexities : A Software Construction Perspective Department of Internet Engineering and Computer Science," vol. 11, no. 8, pp. 227–240, 2012.

[5] J. Reason, *Human Error*. 1990, p. 302.

[6] S. Misra, "Agile software development practices: evolution, principles, and criticisms," *Int. J. Qual. Reliab. Manag.*, vol. 29, no. 9, pp. 972–980, 2012.

[7] S. Ambler, "Survey says... agile has crossed the chasm," *DR DOBBS J.*, pp. 6–10, 2007.

[8] E. Whitworth and R. Biddle, "The social nature of agile teams," *Agil. Conf. (AGILE), 2007*, pp. 26–36, Aug. 2007.

[9] I. Jacobson, P.-W. Ng, P. E. MacMahon, I. Spence, and S. Lidman, *The essence of software Engineering: applying the SEMAT kernel*. Addison-Wesley, 2013.

[10] K. Lewin, "03 - Psychoanalysis and topological psychology," in *The complete social scientist: A Kurt Lewin reader*, 1999, pp. 67–74.

[11] T. Demarco and T. Lister, "PROGRAMMER THE EFFECTS PERFORMANCE AND OF THE WORKPLACE," pp. 268–272, 1985.

[12] A. Avizienis, J. Laprie, and B. Randell, "Fundamental concepts of dependability," *Toulouse Fr. LAAS-CNRS Res. Rep.*, 2000.

[13] J. Reason, "Human error: models and management," *BMJ Br. Med. J.*, vol. 320, no. March, pp. 4–6, 2000.

[14] J. Rasmussen and K. J. Vicente, "Coping with human errors through system design: implications for ecological interface design," *Int. J. Man. Mach. Stud.*, vol. 31, no. 5, pp. 517–534, Nov. 1989.

[15] X. Zhang and H. Pham, "An analysis of factors affecting software reliability," *J. Syst. Softw.*, vol. 50, 2000.

[16] K. Tyagi and A. Sharma, "Reliability of Component Based Systems - A Critical Survey," *SIGSOFT Softw. Eng. Notes*, vol. 36, no. 2, pp. 1–6, 2011.

[17] T. DeMarco and T. Lister, "Programmer performance and the effects of the workplace," *Proc. 8th Int. Conf. Softw. Eng.*, pp. 268–272, 1985.

[18] K. Sandhof and L. Filgueiras, "Defeitos de software como erros humanos," *II Work. WOSES*, pp. 75–84, 2006.

[19] M. Simonette and E. Spina, "Human side of software development: Teamwork as sociotechnical systems," *Softw. Eng. METHODS, …*, pp. 21–28, 2012.

[20] C. M. Burns and G. A. Jamieson, "Evaluation of Ecological Interface design for nuclear process control: Situation awareness effects," *Hum. Factors J. Hum. Factors Ergon. Soc.*, vol. 50, no. 4, pp. 663–679, 2008.

[21] D. V. C. Reising and P. M. Sanderson, "Testing the Impact of Instrumentation Location and Reliability on Ecological Interface Design: Control Performance," *Proc. Hum. Factors Ergon. Soc. Annu. Meet.*, vol. 44, no. 1, pp. 124–127, Jul. 2000.

[22] J. Musa, "Software quality and reliability basics," *Proc. 1987 Fall Jt. Comput. …*, 1987.

[23] I. Jacobson, P. Ng, P. E. Mcmahon, I. Spence, and S. Lidman, "The Essence of Software Engineering : The SEMAT Kernel," *Queue*, vol. 10, no. 10, p. 40, 2012.

[24] A. Alliance, "Guide to Agile practices," 2013. [Online]. Available: http://guide.agilealliance.org/subway.html.

[25] VersionOne, "State of Agile Development Survey." 2012.

[26] M. Paasivaara, S. Durasiewicz, and C. Lassenius, "Using Scrum in Distributed Agile Development: A Multiple Case Study," *2009 Fourth IEEE Int. Conf. Glob. Softw. Eng.*, pp. 195–204, Jul. 2009.

[27] C. T. Collins, R. Street, H. Springs, and R. W. Miller, "Adaptation : XP Style."

[28] T. P. Moran, E. Saund, W. Van Melle, A. U. Gujar, K. P. Fishkin, and B. L. Harrison, "Design and Technology for Collaborage : Collaborative Collages of Information on Physical Walls," vol. 1, pp. 197–206, 1999.

[29] W. Ju, L. Neeley, T. Winograd, L. Leifer, and W. Terry, "Thinking with Erasable Ink : Ad-hoc Whiteboard Use in Collaborative Design," *CDR Tech. Rep.*, 2006.

[30] X. Wang, D. Graziotin, J. Rikkilä, and P. Abrahamsson, "Traverse the landscape of the mind by walking: an exploration of a new brainstorming practice," pp. 1–12, 2013.

[31] C. Melo, V. Santos, and H. Corbucci, "Métodos ágeis no Brasil: estado da prática em times e organizações," *Relatório Técnico RT-MAC- …*, no. 2010, pp. 1–9, 2012.

[32] K. Vicente, "Ecological interface design: Progress and challenges," *Hum. Factors J. Hum. Factors …*, 2002.