A Literature Review on Agile Methodologies Quality, eXtreme Programming and SCRUM

Naglaa A. Eldanasory
Information Systems Department, Helwan University, Naglaa.Aly21@commerce.helwan.edu.eg

Engy Yehia
Business Information Systems Department, Faculty of Commerce and Business Administration, Helwan University, engy_yehia@commerce.helwan.edu.eg

Amira M. Idrees
Department of Information Systems, Faculty of Computers and Information Technology, Future University in Egypt, amira.mohamed@fue.edu.eg

A Literature Review on Agile Methodologies Quality,

eXtreme Programming and SCRUM

Naglaa A. Eldanasory [1, a], Engy Yehia [2, b], Amira M. Idrees [3, c]

[1] Business Information Systems Department, Helwan University, Cairo, Egypt,

[2] Business Information Systems Department, Faculty of Commerce and Business Administration, Helwan University, Cairo, Egypt,

[3] Department of Information Systems, Faculty of Computers and Information Technology, Future University in Egypt, Cairo, Egypt.

[a] Naglaa.Aly21@commerce.helwan.edu.eg, [b] engy_yehia@commerce.helwan.edu.eg,

[c] amira.mohamed@fue.edu.eg

ABSTRACT

Agile methodologies have become one of the most applied methods in the software development industry. However, agile methodologies face some challenges such as less documentation and wasting time considering changes. This review presents how the previous studies attempted to cover issues of agile methodologies and the modifications in the performance of agile methodologies. The paper also highlights unresolved issues to get the attention of developers, researchers, and software practitioners.

Keywords: Agile methodologies, Quality, Scrum, eXtreme Programming (XP)

1. Introduction

Agile methods played a significant role in software development in comparison to other methods. Because many businesses desire to implement better-quality systems in a short period and for the least amount of money [1].For traditional methodologies that have had major issues such as less user interaction, longer development time, higher costs, no adaptability, and, most importantly, no response to changing user requirements that have forced software experts to seek new directions in software development [2].

Agile combines various methodologies, including Scrum [3, 4] XP [5, 6], Feature Driven Development (FDD) [7, 8], Kanban [9], Test Driven Development (TDD) [10], etc. All of these agile models conform to agile values and principles, as well as a few key practices. These practices are not novel in the software industry, but they produce far superior results when applied by agile values and principles. Furthermore, agile is less rigid and includes more customer participation as a quality assurance practice [11]. However, it has its disadvantages such as a lesser emphasis on documentation.
The remainder of the paper is organized as follows: part 2 shows the background. Part 3 shows a related review. Part 4 shows the research design. Part 5 shows relevant papers. Part 6 shows the discussion. Part 7 shows the conclusion.

Agile software development methodologies provide a more efficient and lighter way of developing software by iteratively and incrementally building it. The goal of agile software development models was to discover new and more efficient software development methods that could overcome the limitations of traditional software development models [2]. In this paper, two methodologies are discussed, they are eXtreme Programming and SCRUM.

In terms of agile project management methodology, **Scrum** is among the most popular agile methodologies [3, 4]. This method designed for groups of three to nine people who divide their work into activities that can be completed in iterations known as "sprints." Sprints last no more than one month and are usually two weeks. They are re-scheduled in 15-minute standup gatherings known as "daily scrums." Scrum has three roles: product owner, scrum master, and development team. Despite its many well-known advantages, the scrum methodology is not without flaws. Scrum software development is best suited for small projects, but implementing Scrum in large organizations becomes more difficult.
Another famous agile methodology is **XP**. It is one of the most successful agile methods [12, 5, 6]. This method places high value on customer satisfaction. It allows changes in requirements from customers at any stage of the software development life cycle (SDLC). The first step of XP is gathering user requirements, which are then split into a small number of cycles. The iteration is the next step. XP is suitable for simple and small-scale projects, but it has weaknesses, such as a lack of documentation.

2. Related review

Agile models have gained widespread acceptance in the software industry due to features such as the ability to incorporate change, rapid development, and an emphasis on quality. In a variety of studies, researchers attempted to investigate these models. But most of them focus on showing advantages and disadvantages of agile methodologies and lacks showing some enhancement of models [1]. Listed below are previous studies.

This section includes three parts: performance, integration, and enhancement of agile development methodologies. The purpose of these parts is to improve the quality of agile methodologies as well as to discover those that have not yet been improved.

2.1 Scrum Methodology

The research was conducted by [3] proposed scrumbear to overcome the scrum issues such as not receiving changes in working scrum, sprint's release insufficient time, and consuming team time within each sprint. The implementation was making some changes, such as control master, N-sprints, and regression test to the current scrum model. Control master's rule is used to retain the modifications under control. N-sprints involve a single release for multiple sprints at the same time, as opposed to one release for one sprint. Therefore, the regression test will be run at the end of N-Sprints to ensure that all sprints have been completed.

The research was discussed by [4] proposed CESP (Cost Efficient Scrum Process) process which will aid in reducing resources waste in earlier stages by reordering the tasks that would be necessary for those stages. As a result, the whole project time and cost are reduced. The paper's major goal is to reduce the total project's cost and time, address sprint workload, and testing in the next sprint.
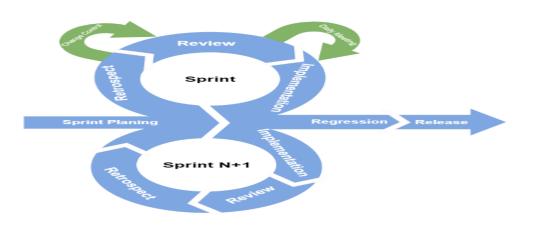

Figure 1 Scrumbear model [3]

Figure 2 CESP Framework [4]

26

The research was presented by [13] proposed a hybrid technique of scrum application with mind mapping and joint requirements document (JRD) to define, adapt, evolve, and handle system requirements progressively across the software development cycle to cover issues of requirement elicitation. These issues such as identification of incomplete requirements, implementation of vague requirements, and prioritization of secondary functional requirements that eventually lead to product failure.
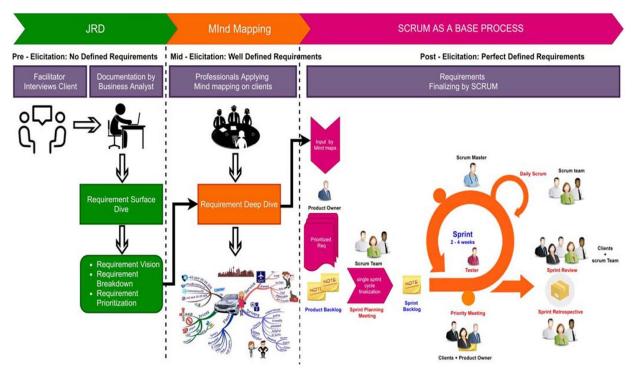

Figure 3 framework for requirements elicitation [13]

The research was discussed by [14] proposed RScrum (Refined Scrum) process model which solves the technological problems associated with scrum methodology. For example assigning, prioritizing, and integrating product backlog items, prove difficult to manage in agile methodology. The requirements are efficiently prioritized using staging priority, which has a lower standard deviation, indicating that priority is distributed evenly across modules. The developed modules are fully integrated on the basis of cyclomatic complexity, which aids in the time and monetary value savings of the regression testing mechanism. The software development process can be more agile as a result of the time savings.

2.2 eXtreme Programming Methodology (XP)

The research was discussed by [6] proposed a scaled version of the XP process model called SXP. XP has some drawbacks, such as inadequate documentation, a lack of design focus, poor architecture, limited testing, and the absence of proper change management activities. The real issue arises when XP is chosen to create a small, low-risk project, but the project scope progressively transitions from small to medium or large as requirements change.

The research was conducted by [15] examined the XP model. Drawbacks of XP eliminate by analyzing various parameters

27

(*Requirement Analysis, Reliance on Refactoring and unit testing, Elimination of bugs, Define Process, and Time and Cost*) and conversing with software developers about parameters because it is required for qualitative software development. The results of the questionnaire found that analysis of

requirements, eliminating all bugs at each phase and in each module, and defining the process before starts developing a module of software are necessary for XP. But it is not necessary to restructure the existing code because it is time-consuming.
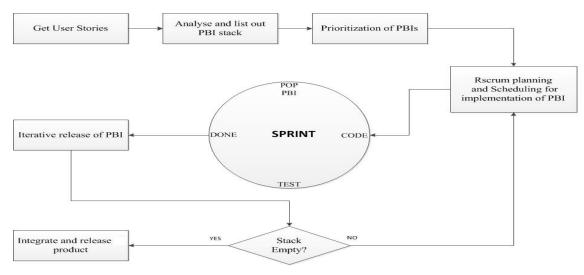

Figure 4 RScrum model [14]

Table 1 Scrum Methodology

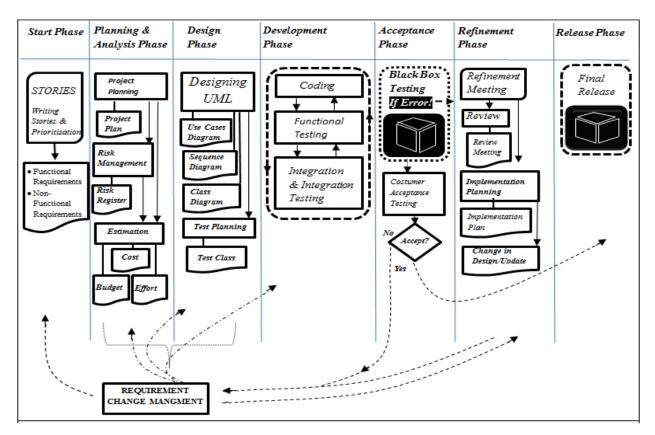| Reference	Country	Objective	Research Method	Contributions	Limitations	Quality Attributes
[3]	Saudi Arabia	To overcome the scrum issues.	An online questionnaire	Proposed Scrumbear model.	This research also lacks empirical proof.	Time
[4]	Egypt	To reduce the total project cost and time.	Experiment research.	Proposed CESP model.	A small project with few developers in the CESP will not improve project efficiency.	Time Cost Efficiency
[13]	Norway	To overcome inappropriate requirements elicitation.	A qualitative research	Proposed a hybrid technique of SCRUM.	Requirements elicitation can be used to improve requirements elicitation	Reliability
[14]	India	To solve the technical problems associated with scrum methodology.	A qualitative approach.	Proposed RScrum process.	It's limited to technical issues.	Time Cost Correctness

28

Figure 5 Scaled XP [6]

Table 2: XP methodology

Reference	Country	Objective	Research Method	Contributions	Limitations	Quality Attributes
[6]	Pakistan	To overcome XP issues	A case study.	Proposed SXP by managing the Risk Register, Adding UML Artifacts.	The number of defects in SXP is not acceptable.	Correctness Reusability
[15]	Pakistan	to eliminate drawbacks of the XP model	A questionnaire	The research analyzes parameters to eliminate drawbacks	It should apply changes to XP according to drawbacks to solve it.	Time Cost Correctness

29

3 Discussion

Agile software development models such as XP, Scrum, etc., have benefits including rapid response to changing requirements, quick feedback, early software delivery, cost reduction, and good time management. However, some major issues include a lack of management/staff control and a lesser emphasis on design and documentation.

The paper discusses the *Performance of agile methodologies, 2) integration of agile methodologies and 3) enhanced agile methodologies. Several s*tudies improved performance through modifications in methodology. *In the Scrum methodology*, [3] proposed Scrumbear by adding a control master to control changes, adding sprint N to eliminate waiting time. Furthermore, [4], proposed CESP by reordering the tasks that would be necessary for earlier stages. *In the XP methodology,* [6,] proposed SXP by organizing the Risk Register, adding UML artifacts, Effective Testing Mechanism, Formal Refinement Techniques, and an RCM Procedure. [5,] proposed SXP model by eliminating the constraints of pair programming and on-site customer to avoid unneeded conflicts, as well as adding UML and documentation.

The authors [16], presented HAQPE which provides a complete lifecycle of software development for software development companies. [17]Proposed L-Scrumban model. [18], presented 'Scrumban' to improve productivity with better business benefits. Moreover, The authors [19], proposed a more reactive and effective architecture for increasing agile's capacity to solve reusability, documentation, and, ultimately, quality concerns. [20], presented comprehensive work is also being done to suggest and verify (using I think) various quality models.

4 Conclusion

In this paper, a review of 24 papers was done. The paper focuses on showing the limitations of agile methodologies and how previous studies solved them. The paper divided previous studies into three parts according to different ways in overcoming the issues of agile methodologies. The first part showed modifications in methodologies of Scrum and XP.

References

[1] A. Hossain, M. A. Kashem and S. Sultana, "Enhancing software quality using agile techniques," Journal of Computer Engineering, pp. 87-93, 2013.

[2] F. Anwer, S. Aftab, U. Waheed and S. S. Muhammad, "Agile software development models tdd, fdd, dsdm, and crystal methods: A survey.," International journal of multidisciplinary sciences and engineering, pp. 1-10, 2017.

[3] L. Alsaber, E. Al Elsheikh, S. Aljumah and N. Jamail, ""Scrumbear" framework for solving traditional scrum model problems," Bulletin of Electrical Engineering and Informatics, vol. 10, no. 1, pp. 319-326, 2021.

[4] E. E. Hassanein and S. A. Hassanien., "Cost Efficient Scrum Process Methodology to Improve Agile Software Development," International Journal of Computer Science and Information Security (IJCSIS), 2020.

[5] F. Anwer, S. Aftab, M. S. Bashir, Z. Nawaz, M. Anwar and M. Ahmad., "Empirical comparison of XP & SXP," IJCSNS, vol. 18, no. 3, p. 161, 2018.

[6] M. Ibrahim, M. A. Shabib Aftab, A. Iqbal, B. S. Khan, M. Iqbal, B. N. S. Ihnaini and N. S. Elmitwally, "Presenting and Evaluating Scaled Extreme Programming Process Model," International Journal of Advanced Computer Science and Applications, 2020.

[7] S. Aftab, Z. Nawaz, M. Anwar, F. Anwer, M. S. Bashir and M. Ahmad, "Comparative Analysis of FDD and SFDD," International Journal of Computer Science and Network Security, pp. 63-70, 2018.

[8] S. Aftab, Z. Nawaz, F. Anwer, M. Ahmad, A. Iqbal, A. A. Jan and M. S. Bashir, "Using FDD for small project: An empirical case study.," International Journal of Advanced Computer Science and Applications, vol. 10, no. 3, pp. 151-158, 2019.

[9] M. Senapathi and M. L. Drury-Grogan, "Systems thinking approach to implementing kanban: A case study.," Journal of Software: Evolution and Process, vol. 33, no. 4, p. e2322, 2021.

[10] M. M. Moe, "Unit Test using Test-Driven Development Approach to Support Reusability," International Journal of Trend in Scientific Research and Development (IJTSRD), vol. 3, no. 3, 2019.

[11] N. A. Bonner, "Predicting leadership success in agile environments: An inquiring systems approach," Academy of Information & Management Sciences Journal, vol. 13, no. 2, 2010.

[12] F. Anwer, S. Aftab and I. Ali, "Proposal of Tailored Extreme Programming Model for Small Projects.," International Journal of

Computer Applications, vol. 171, no. 7, pp. 23-27, 2017.

[13] H. Saeeda, J. Dong, Y. Wang and M. A. Abid, "A proposed framework for improved software requirements elicitation process in SCRUM: Implementation by a real-life Norway-based IT project.," Journal of Software: Evolution and Process, vol. 23, no. 7, p. e2247, 2020.

[14] R. Anand and M. Dinakaran, "Improved scrum method through staging priority and cyclomatic complexity to enhance software process and quality," International Journal of Internet Technology and Secured Transactions, vol. 8, no. 2, pp. 150-166, 2018.

[15] H. Iftikhar and W. Azmat, "A Structured Approach to Analyze XP Model and Eliminate Drawbacks for Qualitative Development of Software," Journal of Information Engineering and Applications, pp. 6-12, 2018.

[16] L. Neelu and D. Kavitha, "Estimation of software quality parameters for hybrid agile process model," SN Applied Sciences, vol. 3, no. 3, pp. 1-11, 2021.

[17] A. Albarqi and R. Qureshi, "The proposed L-Scrumban methodology to improve the efficiency of agile software development," International Journal of Information Engineering and Electronic Business, vol. 12, no. 3, p. 23, 2018.

[18] S. P. Patil and J. R. Neve, "Productivity Improvement of Software Development Process Through Scrumban: A Practitioner's Approach," in 2018 International Conference On Advances in

Communication and Computing Technology (ICACCT), 2018.

[19] M. A. Khalid, T. Anees and A. Moeed, "LAQF: Lightweight Document Oriented, Reusable Agile Quality Framework," in 2019 International Conference on Innovative Computing (ICIC), 2019.

[20] A. Tabassum, D. S. N. Bhatti, A. R. Asghar, I. Manzoor and I. Alam, "Optimized quality model for agile development: extreme programming (XP) as a case scenario.," IJACSA) International Journal of Advanced Computer Science and Applications, vol. 8, no. 4, pp. 392-400, 2017.