

Comparison of Agile Method and Scrum Method with Software Quality Affecting Factors

Muhammad Asaad Subih¹, Babur Hayat Malik², Imran Mazhar³, Izaz-ul-Hassan⁴, Usman Sabir⁵
Tamoor Wakeel⁶, Wajid Ali⁷, Amina Yousaf⁸, Bilal-bin-Ijaz⁹, Hadiqa Nawaz¹⁰, Muhammad Suleman¹¹
University of Lahore, Gujrat Campus, Gujrat, Pakistan

Abstract—The software industry used software development lifecycle (SDLC) to design, develop, produce high quality, reliable and cost-effective software products. To develop an application, project team used some methodology which may include artifacts and pre-defining specific deliverables. There are different SDLC process models such as waterfall, iterative, spiral and agile model available to develop a quality product. In this paper we focus only on agile software development model, and Scrum model and their techniques. There are many papers and books written on agile methodologies. We will also use their knowledge in this paper. To collect data for comparison of agile method with software quality affecting factors, an online questionnaire survey was conducted. The survey sample consisted of software developers with several years of industry experience using agile methodologies. The main purpose of this study is to compare software quality affecting factors with agile and scrum model.

Keywords—Component; SDLC; Software Quality Affecting Factors; Agile methodologies; Scrum

I. INTRODUCTION

Agile methodologies have played a vital role in the development of software as compared to other methodologies. Because many companies want to implement good-quality systems, and they want to do it in a minimum period of time, at a less cost [1][2][5]. Therefore, many companies have started to follow agile methods to develop software [1], and it has been found that the extent of the organizational team's skills [1], culture of the organization [1], nature of project [1], and project constraints must be given in-depth consideration [1] when selecting an Agile method. Agile Software Development Methodologies [1][2][3] are based on both incremental and iterative development.

There are different agile methodologies such as scrum, Kanban, Extreme programming, Dynamic System Development, Feature Driven Development, [1][2][3][5]. As our research aim is focus only on agile software development model, and Scrum model and their techniques. The main purpose of this study is to compare software quality affecting factors with agile and scrum model [1]. Nowadays, many organizations are using agile methodologies because these methodologies support flexibility, changes at any stage, and light documentation.

Scrum [1][5][6] is an agile framework focuses software development. This method is based upon consistent installments, and regular collaboration among self-organizing cross-functional departments. This method is for 3 to 9

members team who break their work into actions that can be completed within time boxed iterations, called “sprints” [1]. The sprints are not more than one month and most commonly two weeks, re-plan in 15min standup meetings, called “daily scrums”. Scrum rules are product owner, scrum master and team. Scrum is easy with changes; it accommodates with changes [1]. Agile methodologies address quality issues and focusing on the role of quality.

II. RELATED WORK

It is important to work on product quality and to measuring software development projects to get a better understanding about the progress of project [2]. Agile methodologies address quality issues repeatedly and continuously. Agile method focuses on software quality through customers [2], minimum error rate, faster development and welcome to changing environment.

As Oberscheven [1] research aim is focus on agile software development model, and Scrum model and their techniques. The main purpose of this study is to compare software quality affecting factors with agile and scrum model. Nowadays, many organizations are using agile methodologies because these methodologies support flexibility, changes at any stage, and light documentation. There are different agile methodologies such as Scrum, Kanban, Extreme Programming, Crystal, Dynamic System Development Method, Feature Driven Development, and Adaptive Software Development. A Hossain, MA Kashem: [2] describe that the agile techniques are used to minimize risk factors by developing software in short period of time. During the software development the changes are welcome in agile method. In the end, this paper concludes that agile techniques are using to increase the quality of software all the way through increased customer value. It de-scribes how we can increase the quality of software by using agile techniques.

Sheetal Sharma [4] describe the factors that affect the quality of software [4] are correctness, reliability, usability, extensibility, reusability, testability, portability, maintainability, and efficiency. This research paper is about SDLC models and different scenarios which are using by developers for developing a well-engineered software. Sheetal Sharma describes some advantages and disadvantages of agile methodologies in our research paper. According to this, there are least documentation in agile method and ensure the customer satisfaction but meanwhile least documentation is also disadvantage of agile methodology.

III. SOFTWARE QUALITY AFFECTING FACTORS

Software quality is related to customer satisfaction and low error levels in software. The external and internal quality criterion are used to evaluate the quality of a software. External quality is relating to the functionality of the software. Internal quality relates to coding, and that are not visible to the end-user [2][6][8]. These Software Quality factors are shortlisted by the brief literature study and are supported by the literature review as in [2][6][8]. The factors under the study are supported by above mentioned literature.

Generally, the software qualities are of three types:

- Quality of design
- Quality of performance
- Quality of Adoption

These three qualities can be further divided into other quality attributes. Table I shows the factors affecting the software quality and their attributes.

TABLE I. SHOWS THE SOFTWARE QUALITY AFFECTING FACTORS

Software Quality Factors	
Quality of Design	Description
Correctness	If software doesn't work correctly as required then it is wasteful.
Maintainability	If software is not able to add new features or to remove error, then it has no worth.
Quality of Performance	
Efficiency	It is a factor relating to all issues in the execution of software.
Reliability	It defines how well the software meets his requirements.
Usability	If software is not user friendly then it is hard for user to use.
Testability	If testing is not done properly then it makes software errors.
Quality of Adoption	
Extensibility	If software coding is not extendable by adding new features then it has no worth.
Portability	It is the effort required to transfer the software from one configuration to another.
Reusability	If software is not reusable then it is limited product.

IV. SOME POPULAR AGILE METHODOLOGIES

After All of the agile methodologies acknowledged to produce higher quality software and more significantly the satisfaction of customer [1][7]. Here, some of agile methodologies are listed below.

A. Extreme Programming (XP)

XP [3][4][6][7] is one of the most successful agile method. This method focuses on customer satisfaction. It divides the SDLC into various number of short development cycles. At

any phase of SDLC, it allows changes or requirements from the customers. First phase of Extreme programming is collecting user requirements, and then these requirements are divided into various small no of cycles. Now the upcoming phase is iteration planning. If any new user requirement may come during the development phase then according to that, iteration plan should be tuned [4]. Next step is testing and errors will be removed in the next iteration.

B. Feature Driven Development (FDD)

FDD [3][6][7] is one of the agile development methods. Designing of the domain of software is the main feature of this method and it also focus on the building phases of the software. The first phase of the method is to get user requirements and constructing the overall model of the project. Next phase is to list down features relates to the user-valued functions [3] [6]. For example, 'calculation of company's each employee', 'calculation of tax each company's employee'. There are different groups of features are made based on their domains i.e. related features are combined into a single group. In another step make a plan for developing and assigned these tasks to development team.

C. Scrum Method

Scrum [1][5][6] is an agile framework focuses software development. This method is based upon consistent installments, and regular collaboration among self-organizing cross-functional departments. This method is for 3 to 9 members team who break their work into actions that can be completed within time boxed iterations, called "sprints" [1]. The sprints are not more than one month and most commonly two weeks, re-plan in 15min standup meetings, called "daily scrums". Scrum rules are product owner, scrum master and team. Scrum is easy with changes; it accommodates with changes. Scrum [1][5][6] is a simple framework used to organize teams and get work done more productively and with higher quality. Scrum is easy with changes; it accommodates with changes. Some key scrum practices are discussed below [1][3][4][5]. These key factors are taken in account for description one by one.

- Product Backlog – The software development team identified all tasks and makes a list called the Backlog.
- Sprints – Sprint is 3 to 9 members team who break their work into actions that can be completed within time boxed iterations, called "sprints" [1]. The sprints are not more than one month and most commonly two weeks.
- Sprint planning meeting – There are different stakeholders are involved in sprint planning meeting. They decide the functionality of the system. The stakeholders are customers, product owner and scrum team.
- Sprint Backlog – When a list of tasks is completed than a new iteration of the software product is delivered.
- Daily Scrum – These daily scrum meetings are not more than 15minutes long [9].

V. EVALUATION OF SOFTWARE QUALITIES WITH AGILE TECHNIQUES

A. System Metaphor

The system metaphor is all about that how system works. In system metaphor customers, programmers, and managers are involved. The System metaphor [2] is used to facilitate communication between customer and developer. It helps the agile development team [2] in the development of software by in-creasing communication between developers and users. So, by using system metaphor maintainability, efficiency, reliability and flexibility of the system enhance.

B. Architectural Spike

An architectural spike is technique that can re-duce the technical risk factor from the product and it comes from Extreme Programming (XP). The main purpose of this technique is to reduce the risk of a technical problem [2]. The spike is between product owner and development team.

C. Onsite Customer Feedbacks

Customer are only involved while the development of software. In agile methodologies, communication with the customers are required, which is in-tended to improve productivity [2]. Agile methodologies emphasize a lot on customer feedback.

D. Refactoring

During the development of software, we can improve the internal structure of the software without effect on external behavior. So, by using refactoring, efficiency, reliability, intra-operability and interoperability [2], testability of the system enhances.

E. Pair Programming

It is a technique in which two programmers are involved on same code. One programmer writes code while the other monitor the code and gives reviews. So, by using Pair programming correctness, verifiability [2], testability of the system enhances and reduces defects.

F. Stand-Up-Meeting

This method is for 3 to 9 members team who break their work into actions that can be completed within time boxed iterations, called "sprints" [1]. The sprints are not more than one month and most commonly two weeks, re-plan in 15min standup meetings, called "daily scrums". Stand-up-meeting is very useful for improving the quality of the software like reliability and flexibility.

G. Continuous Integration (CI)

The programmers are sharing the code in CI technique [2]. In continuous integration like the auto-mated compilation, unit test execution, and source control integration are configure by agile teams.

Table II shows the software qualities in the agile development. The software qualities contain Maintainability, Verifiability, Efficiency, Integrity, Reliability, Usability, Testability, Expandability, Flexibility, Portability, Reusability, Interoperability and interoperability. Table II further describes their use in the agile development technique.

TABLE II. SHOWING THE SOFTWARE QUALITIES IN AGILE DEVELOPMENT TEAM

Software Qualities	Agile Technique
Maintainability	Continuous Integration
Verifiability	Continuous Integration
Efficiency	Pair programming System metaphor
Integrity	Continuous Integration
Reliability	Refactoring System metaphor
Usability	Continuous Integration
Testability	Pair programming Acceptance testing Unit testing Refactoring
Expandability	Continuous Integration Onsite customer feedback
Flexibility	Stand-up meeting System metaphor
Portability	Pair programming Acceptance testing
Reusability	Refactoring Continuous Integration
Interoperability	Refactoring System metaphor
Intra-operability	Continuous Integration

VI. QUESTIONNAIRES

The following is a questionnaire developed to compare the software quality affecting factors with agile methodologies. The data on agile methodology was collected with the help of questionnaire survey. A summary of the survey and their results are given below.

A. Questionnaire Format

The questions were divided into three parts: First part deals with the respondent's position in the organization and experience on agile methodologies. Second section related to the agile methodologies and scrum method. Last section deals with the software quality affecting factors of agile methodologies.

B. Questions

Following the questionnaire survey approach, we formulated questions which help us to assess goals. The defined questions are presented in this section.

Table III shows the questions being asked and their relation to the agile development and software qualities. Table III is divided into three main sections.

First section is about the personal information of the respondent, his positions in the organization and his experience about the agile development. Second section queries about the agile methodology and the Scrum development technique and the last most and the third section inquiries about the software quality affecting factors with respect to agile methodology.

TABLE III. SHOWS THE QUESTIONS RELATING TO AGILE METHOD AND SOFTWARE QUALITIES

Questions	
Respondent's position in the organization and experience on agile methodologies.	Q1.1 What is your name? ____ Q1.2 What is your company name? _ Q1.3 What is your job title? ____ Q1.4 How long have you Experiences with Agile Methods? • <1-year • 1-2 years • 3-5 years.
Agile methodologies and Scrum method	Q2.1 What are the key reasons to start agile? • Quality • Productivity • Predictability • Team health Q2.2 Starting Scrum Meeting Practice Difficulty? • Easy • Hard Q2.3 What is the frequency of using Scrum meeting? • Never • Irregularly • Frequently • Daily Q2.4 Frequency of Using agile methodologies? • Never • Rarely • Frequently • Always
Software quality affecting factors with agile methodologies	Q3. By using agile methodologies for the development of software, then the software has following qualities: give answer of every option as "agree, disagree, strongly agree, strongly disagree". • Software does work correctly as required. • Software is error free. • Software is user friendly for use. • Software coding is extend-able for adding new features. • Software can be reused in other related applications. • Software has testability quality. • Software can be transfer from one configuration to another. • Software is able to add new features or to remove error. • Software is efficient.

C. Questionnaire Distribution

The questionnaire was distributed through e-mail. Due to shortage of time, questionnaire was distributed among 15 different software developers. Then 10 of the software professionals responded within 2 days. One of the main reasons to select these persons which are working in soft-ware development companies was, easy to access and easy communication.

D. Questionnaire Results

First questionnaire section deals with the respondent's position in the organization and experience on agile methodologies. Fig. 1 represent that 28% respondents have longer experience in agile, 27% of the respondents have 3 to 5 years' experience and only 18% respondents have less than 1-year experience.

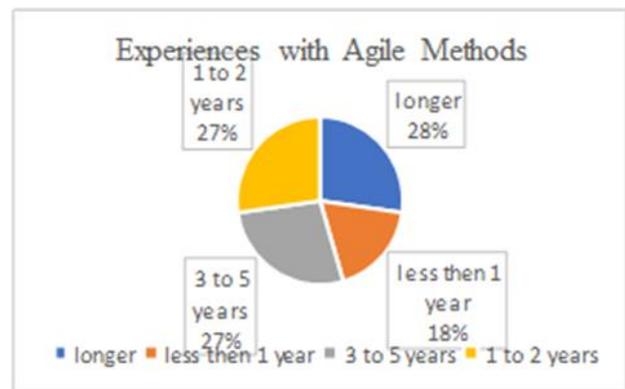


Fig. 1. Shows Agile Methods and Software Quality Factors Questionnaire Results Experiences with Agile Methods.

Second section related to the agile method and scrum method (see Fig. 2, 3, 4 and 5). Fig. 2 shows that what are the key reasons that developers are using agile methodologies. Graph show that 10 developers are using agile methodologies because of quality, 6 developers are using agile method due to team health and five dues to productivity.

Fig. 3 show that 10 respondents state that starting scrum meeting practice is not difficult.

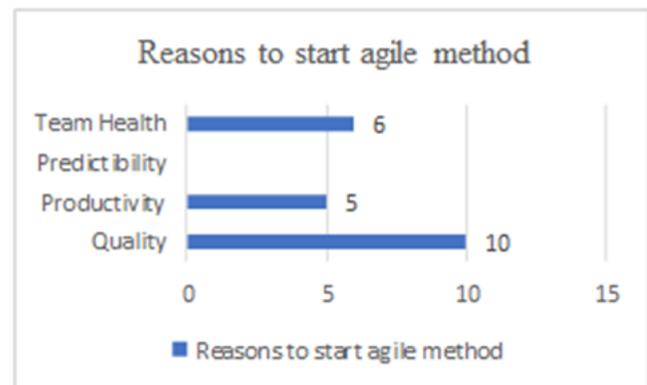


Fig. 2. Agile Methods and Software Quality Factors Questionnaire Results–Key Reasons to Start Agile methodologies.

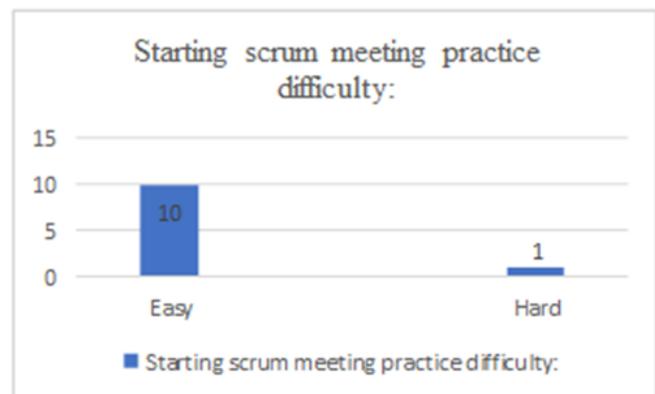


Fig. 3. Shows Agile Methods and Software Quality Factors Questionnaire Results–Starting Scrum Meeting Practice.

Fig. 4 shows that 75% of software professionals are using scrum meeting frequently and only 25 % irregularly. Fig. 5 shows that 70% respondents are always using agile methodologies for the development of software.

Last section (Fig. 6) deals with the software quality affecting factors of agile methodologies. Fig. 6 shows that the comparison of software quality affecting factors with agile methodologies. The questionnaire result indicates that almost all developers use agile methodologies to fulfill the qualities of software. In the survey, 10 respondents answer that they use agile methodologies for the development of software and the reason behind that software qualities like correct-ness, reliability, usability, extensibility, reusability, testability, portability, maintainability and efficiency.

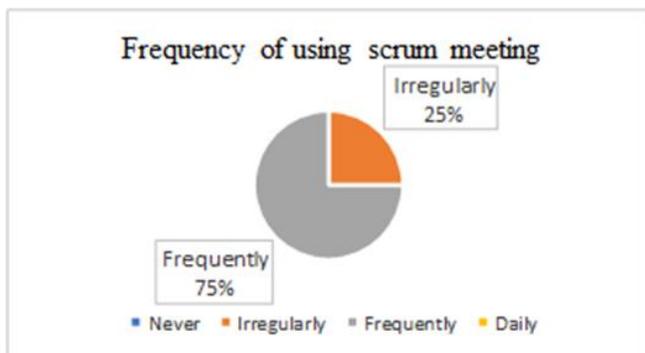


Fig. 4. Shows Agile Methods and Software Quality Factors Questionnaire Results–Percentage of using Scrum Meeting.

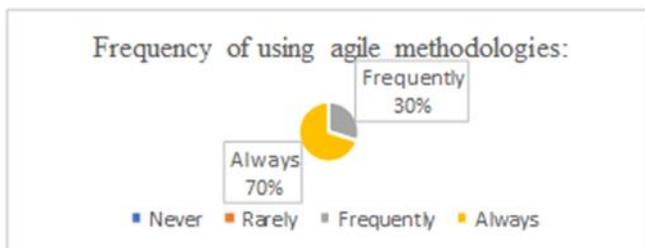


Fig. 5. Shows Agile Methods and Software Quality Factors Questionnaire Results–Frequency of using Agile Methodologies.

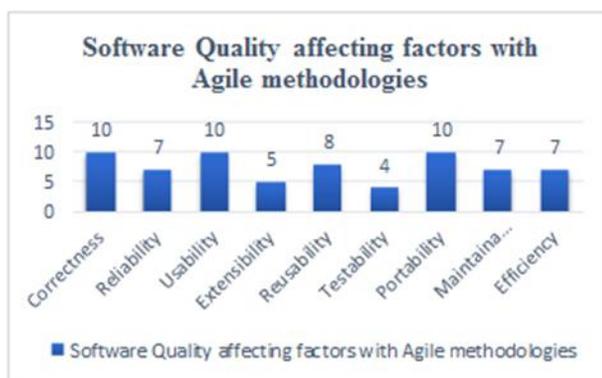


Fig. 6. Comparison of Software Quality Affecting Factors with Agile Methodologies.

E. Data Validation

For the confirmation of data validation, we interact with the respondents and asked them about the surety of their answers. Then, they inform us that, they also consulted with other team members in the company who had higher experience in the agile field. So, after sure about the opinion of the professionals, they filled in the questionnaire. That's why, respondents were confident about their answers.

VII. CONCLUSIONS AND FUTURE WORK

In this paper, a questionnaire survey was conducted. By this survey we collect data from different software development companies' employees for the comparison of software quality affecting factors with agile and scrum method. We have identified software quality affecting factors such as correctness, reliability, portability, testability, efficiency and extensibility. The main advantage of agile technique is customer satisfaction and its welcome user requirement changing at any phase. By using agile method, the software has almost all software qualities. In future work the questionnaires should be repeated with additional respondents to the results from the user feedback presented in this study. We will enhance our area of research about agile methodologies and software quality affecting factors. In current research we only target the problem that is comparison of agile methodologies with software quality affecting factors, but in future we will try to identify the all identified issues of agile methodologies and their effect on software product and customer's trust. According to the questionnaires the percent-age of software qualities increased by using agile methodologies. In future work we should work on customers satisfaction by using agile methodologies. How well agile methodologies fulfill customers' satisfaction for the development of software.

REFERENCES

- [1] Oberscheven, Falk Martin. "Software Quality Assessment in an Agile Environment." Faculty of Science of Radboud University in Nijmegen. (2013).
- [2] Hossain, Amran, Md Abul Kashem, and Sahelee Sultana. "Enhancing software quality using agile techniques." IOSR Journal of Computer Engineering 10.2 (2013): 87-93.
- [3] Awad, M. A. "A comparison between agile and traditional software development methodologies." University of Western Australia (2005).
- [4] Sharma, Sheetal, Darothi Sarkar, and Divya Gupta. "Agile processes and methodologies: A conceptual study." International journal on computer science and Engineering 4.5 (2012): 892.
- [5] Khalane, Tiisetso, and Maureen Tanner. "Software quality assurance in Scrum: The need for concrete guidance on SQA strategies in meeting user expectations." Adaptive Science and Technology (ICAST), 2013 International Conference on. IEEE, 2013.
- [6] Sirshar, Mehreen, and Fahim Arif. "Evaluation of Quality Assurance Factors in Agile Methodologies." International Journal of Advanced Computer Science 2.2 (2012): 73-78
- [7] Ullah, Malik Imran, and Waqar Ali Zaidi. "Quality Assurance Activities in Agile: Philosophy to Practice." (2009).
- [8] Sharma, Mohit Kumar. "A study of SDLC to develop well engineered software." International Journal of Advanced Research in Computer Science 8.3 (2017).
- [9] Rodríguez-Hernández, V., et al. "Assessing quality in software development: An agile methodology approach." Journal of Advanced Computer Science & Technology 4.2 (2015): 225-230.