International Research Journal of Mathematics, Engineering and IT
ISSN: (2349-0322)
Association of Academic Researchers and Faculties (AARF)
Impact Factor- 7.132 Volume 10, Issue 04, April 2023
Website- www.aarf.asia, **Email** : editoraarf@gmail.com

A STUDY ON THE QUALITY ASSESSMENT FOR AGILE SOFTWARE PRODUCTS

Deepika
Research Scholar, Department of Computer Science Engineering, OPJS University

Dr Vijaypal Singh
Associate Professor, Department of Computer Science Engineering, OPJS University

Dr Suman Rani
Associate Professor, Department of Electronics and communication Engineering, OPJS University

ABSTARCT
Software testing is the most important approach for determining whether or not a product is of a satisfactory quality. In spite of this, testing software using an agile methodology is extremely difficult and presents several obstacles. An investigation into the quality assessment for agile software products is going to be carried out as the primary objective of this research. Based on the findings of this research, an improved version of the agile quality model was recommended to be used in order to evaluate the agile software development process. The Fuzzy Step-Wise Weight Assessment Ration Analysis (Fuzzy-SWARA) method is a multi-criteria decision-making approach that is used for calculating the weight coefficients of the identified quality factors. This helps in the quality estimation of various software-based projects that are carried out in an agile setting. The end result demonstrates a good correlation of quality metrics, which indicates that the proposed method is being accepted in the software business.

Keywords: *Agile Software Products; Quality assessments; Agile environment; Fuzzy – SWARA Method; Multi-Criteria Decision-Making Approach.*

INTRODUCTION

Agile approach is becoming increasingly popular among businesses all over the world as a means of producing their software products. This is owing to the fact that Agile methodology promises to create products that are both faster and of higher quality. As more organisations implement practises of the agilelike software development, trends for testing software development methodologies show that the practises of agility are adapted to the workplace setting (Atawneh, 2019). Software testing makes sure that the final product is what you intended to create in accordance with the system requirements. Additionally, it is possible to spot flaws and defects in the system that might improve the software's quality and it checks to see if there are any errors that could render the software useless (Sophocleous &Kapitsaki, 2020). Testing is becoming a crucial part of other development phases because to agile methodologies, which also guarantee consistent product quality (Honest, 2019).

The research suggests that an improved version of the agile quality model should be implemented so that it may be used to evaluate the agile software development process. The Fuzzy Step-Wise Weight Assessment Ration Analysis (Fuzzy-SWARA) method is a multi-criteria decision-making approach that is used for calculating the weight coefficients of the identified quality factors. This helps in the quality estimation of various software-based projects that are carried out in an agile setting. The result demonstrates a significant correlation of quality values, which suggests that the software industry will adopt the proposed approach.

LITERATURE REVIEW

The authors **Sinha & Das (2021)** recognised that "change is the only constant" applies to the software industry. Most IT businesses now use Agile to produce software quickly. Few firms use the Traditional Waterfall Model for software development. Quality assurance and testing are crucial phases in software development. This discussion included software testing implementation and agile approach. The authors also compared Agile testing to Waterfall testing.

The **Barraood et al., (2021)** study examines Agile software testing practises. Acceptance criteria, iteration modification, and quality assurance activities are used to compare. Identifying these methodologies' similarities and differences, especially in test case creation, is the goal. The study examines XP, Scrum, and Kanban. The review showed no difference in test case design methodologies between these three methods. This result might assist Agile developers and testers create test cases using the same strategy in different Agile techniques.

According to **Poth et al. (2020),** established domains in highly regulated environments require a methodical approach to scale agile approaches and ensure regulatory compliance. This paper provides a structured way to building a systematic strategy for small agile teams. It scales to larger teams or even subsidiaries and is independent of the underlying approach like Scrum or Kanban. Its compliance and quality risk dimensions meet regulatory criteria. Over 100 developers in one subsidiary validated the methodology in financial IT.

According to **Jain, Sharma & Ahuja (2019),** Agile outperformed traditional methods. Rapid development, review, demonstrable progress, and continual delivery satisfy client needs. Thus, agile development must be measured. In this regard, the present research effort explores the inter-relationships and inter-dependencies between the identified quality factors (QF), identifying which QF have strong driving power and dependence power, indirectly helping agile development process success. This study offers a new agile quality model using interpretive structural modelling (ISM) and MICMAC to classify the observed elements. Understanding how these QF interact and how to use them can substantially improve agile development.

From this detailed research, it is apparent that adopting agile practices in an organization can provide several challenges that may affect software product quality. To evaluate agile software development, the research advises implementing an improved agile quality model.

METHODOLOGY

The research suggested that an improved version of the agile quality model should be implemented so that it may be used to evaluate the agile software development process. The Fuzzy Step-Wise Weight Assessment Ration Analysis (Fuzzy-SWARA) method is a multi-criteria decision-making approach that is used for calculating the weight coefficients of the identified quality factors. This helps in the quality estimation of various software-based projects that are carried out in an agile setting. In order to evaluate the quality of every given agile project, the agile quality model is first transformed into a more hierarchical structure. In addition, the investigation revealed that functionality, maintainability, and reliability (FMR) were the three elements that had the most significant impact on the creation of software. This was discovered after it was determined which factors had been investigated. It is required to define the selected elements and to build methods for assessing these aspects in order to meet the quality standards in order to evaluate the overall quality of a software product that was built using an agile development approach. Only then can the overall quality of the software product be evaluated. Next, an MCDM technique known as the Fuzzy Step-Wise Weight Assessment Ratio Analysis (FuzzySWARA) approach is utilised in order to conduct the evaluation.

RESULTS AND ANALYSIS

The study was conducted in an agile setting, and the agile manifesto was taken into account throughout the process. Due to the fact that the projects were conducted in classrooms, only limited testing was done on them. However, the institute is currently utilising an attendance monitoring system as well as an online complaint management system. Both of these systems are operating flawlessly with very little restrictions. The procedure for preparing the end-of-semester results is now being used, but its efficiency is not particularly high due to the fact that it is repeated twice a year when the annual results are announced. In addition, the charge module software system was just recently constructed, and the account department is currently testing it as well as using it. The metric values of each of the criteria that are being considered for weight assignment are compiled for each of

the four projects, and then the overall quality of each classroom-based project is determined by applying the middle relative fuzzy weights to the results of the calculation.
For the purpose of determining the overarching quality of the software projects, it is necessary to conduct an analysis based on the hierarchical agile quality model that has been proposed. These four factors include the effectiveness of defect eradication, the correctness and completeness of functional operations, and the productivity of software development initiatives. In order to obtain the values of these factors, we need to obtain the values of the other factors that have been defined. All of the values were derived from the information that was gathered for each of these four software projects, as can be seen in the figure that follows.

Project Metrics Data

Project #	Project #	User Stories Planned	User Stories (Delivered)Completed	User Stories Accepted	Project Goal (%)	User Stories Acceptance (%)	Story Points Planned	Story Points Actual	Project Goal (%)	Efficiency (%)	DRE: Pre-Delivery defects (defects from all QC / QA activities) (E)	DRE: Post – Delivery Defects (D)	Defect Removal Efficiency (%) (E/ (E+D))*100: Project Goal (%)	Defect Removal Efficiency (%) (E/ (E+D))*100: Actual (%)
Attendance Monitoring System	1	18	18	18	93	100	25	31	0	0.24	35	2	95	95
End Semester Result Preparation	2	12	12	4	93	33	23	28	0	0.2173913	18	4	95	82
Fee Module System	3	10	10	10	93	100	18	21	0	0.16666667	25	6	95	81
Online Complaint Management System	4	19	19	28	93	147	26	33	0	0.26923077	15	3	95	83

Sheet

DD: No. of Story Points completed	DD: Total Defects Found(defects removed per sprint)	Defect Density (Defects/Story Points): Project Goal	Defect Density (Defects/Story Points): Actual	SSF: Planned Effort (in Hrs)	SSF: Actual Effort (in Hrs)	Sprint Stretch Factor (%): Project Goal (%)	Sprint Stretch Factor (%): Actual (%)	Productivity: No. of Story Points Completed	Productivity: Actual Effort (in Hrs)	Productivity (Story Points/Hr): Project Goal	Productivity (Story Points/Hr): Actual
31	37	0.15	1	500	726	0	0.452	31	726	0.2	0.04
28	22	0.15	1	1063	1425	0	0.340545626	28	1425	0.2	0.02
21	31	0.15	1	1240	1350	0	0.088709677	21	1350	0.2	0.02
33	18	0.15	1	1308	1695	0	0.29587156	33	1695	0.2	0.02

Figure 1: Data Collection for Four Web Based Projects
Calculations can also be made on the quality value of the other three projects. The following table presents an overview of the overall quality results for each of the four projects:
Table 1: Agile Projects Quality Evaluation

PROJECT	FUCTIONAL COMPLETENESS	FUNCTIONAL CORRECTNESS	DEFECT REMOVAL EFFICIENCY	EFFICIENCY	OVERALL QUALITY VALUE	PROJECT QUALITY (IN %)
Project 1: Attendance monitoring system	0.244	0.141	0.261	0.055	0.701	70.1%
Project 2: End semester result preparation system	0.097	0.029	0.002	0.050	0.402	40.2%
Project 3: Fee Module Software System	0.014	0.096	0.222	0.039	0.371	37.1%
Project 4: Online Complaint Management System	0.241	0.0	0.228	0.062	0.531	53.1%

CONCLUSION

The final quality values on a scale of zero to one and the FMR for each project can be calculated based on the aforementioned data. Project 1: Attendance Monitoring System came out to be an excellent quality project with high values for functionality, maintainability, and dependability in accordance with the overall ideals. Project 4: Online Compliant Management System appears to be the second-best quality project, with 53.1%, when the projects are arranged in increasing order. Then came Project 2: A system for preparing end-of-semester results, and finally Project 3: The fee module software system, which had the lowest quality rating. Projects 1 and 4 appear to be doing better than the other two projects when these four projects are used in a real-world setting. The overall quality of each of the four projects is assessed and contrasted with their actual results. The quality attained using the suggested strategy was discovered to be compatible with the projects employed. This demonstrates strong association and suggests that the proposed quality model is accepted for use in assessing the general quality of any product created in an agile environment.

REFERENCES

1. Atawneh, S. (2019). The Analysis of Current State of Agile Software Development. Journal of Theoretical and Applied Information Technology, 97(22).
2. Barraood, S. O., Mohd, H., &Baharom, F. (2021). A comparison study of software testing activities in agile methods. In *Knowledge Management International Conference (KMICe)* (Vol. 2021).
3. Honest, N. (2019). Role of Testing in Software Development Life Cycle Nirali. International Journal of Computer Sciences and Engineering, 7(5), 886–889.
4. Jain, P., Sharma, A., & Ahuja, L. (2019). A customized quality model for software quality assurance in agile environment. *International Journal of Information Technology and Web Engineering (IJITWE)*, *14*(3), 64-77.
5. Poth, A., Jacobsen, J., & Riel, A. (2020). Systematic agile development in regulated environments. In *Systems, Software and Services Process Improvement: 27th European Conference, EuroSPI 2020, Düsseldorf, Germany, September 9–11, 2020, Proceedings 27* (pp. 191-202). Springer International Publishing.
6. Sinha, A., & Das, P. (2021, September). Agile methodology Vs. traditional waterfall SDLC: A case study on quality assurance process in software industry. In *2021 5th International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech)* (pp. 1-4). IEEE.
7. Sophocleous, R., &Kapitsaki, G. M. (2020). Examining the Current State of System Testing Methodologies in Quality Assurance. Stray V., Hoda R., Paasivaara M., Kruchten P. (Eds). Agile Processes in Software Engineering and Extreme Programming. XP 2020., 240–249. https://doi.org/https://doi.org/10.1007/978- 3-030-49392-9_16