International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-8 Issue-5, June 2019

Fuzzy Approach To The Health Application Quality Evaluation Process

Inda D Lestantri, Yahya Ahyasi, Elan Suherlan

Abstract: ***Software quality measurement aims as an effort to improve software performance. Software quality will affect the use of software. Many factors influence the quality of software, and some of these factors have a more subjective assessment. Subjective assessment of users is also influenced by the user's personal experience of something or when using software. This study aims to discuss the measurement of software quality related to user expectations or perceptions. This research is important because every user has hope in a software. The expectations of users that are fulfilled will affect subjective judgments. This study discusses the measurement of software quality from 5 variables, which are taken from the characteristics of Product Quality including truth, reliability, efficiency, integrity, usability. From each of these variables, users have their own expectations for software. The software tested is intended for the Health Office and Public Health Center, to monitor the health of pregnant women. The Mc Call method is used as a standard to measure software quality. The results obtained are Fuzzy models can be used to measure software quality. In this study also produced a level of fuzzy values in software measurement, they are very low, low, sufficient, high and very high. Before measuring software quality, software must pass functional testing.***

Index Terms: MMR, Mc Call, e-Health, graph.

I. INTRODUCTION

Maternal mortality rate (MMR) is the number of deaths during pregnancy or within a period of 42 days after the end of pregnancy. This is the result of all causes related to, or aggravated by pregnancy or treatment, but not caused by accidents or injuries [1]. MMR strongly determines the degree of public health. Based on the 2012 Indonesian Demographic and Health Survey (IDHS), the MMR was 359 per 100,000 live births. In 2007 the MMR increased sharply to 228. The maternal mortality rate in Indonesia is much higher than in other ASEAN countries such as Singapore, only 6 per 100,000 live births, Brunei 33 per 100,000 live births, and Philippines 112 per 100,000 live births. In 2015, MMR in Indonesia decreased to 305 per 100,000 based on the results of the Intercensal Population Survey [2]. The impact of high MMR shows that the ability and quality of health services in Indonesia still need to be improved. The high MMR and the slow decline in MMR are very urgent to improve health services both in terms of coverage and quality of health services.

Manuscript published on 30 June 2019.
***** Correspondence Author (s)
inda D Lestantri*, Information Technology Engineering, Yarsi University, Indonesia
Yahya Ahyasi, Information Technology Engineering, Yarsi University, Indonesia
Elan Suherlan, Information Technology Engineering, Yarsi University, Indonesia

Therefore the Application of Pregnancy Visualization is very important to help health workers to find out the amount of data of pregnant women, at risk of lack of energy, bleeding and hypertension. Pregnant women visualization application is an application developed to monitor areas or units of Public Health Centers that have pregnant women patients, especially to monitor risky pregnant women. The idea of developing applications for visualization of pregnant women began with the difficulty of finding up-to-date and real-time information on pregnant women patients, as an effort to assist in decision making and in order to improve services for pregnant women patients. This application was developed for the government (Health Office) which has a function to monitor the Public Health Center units that are under it, against pregnant women patients. This application is developed based on the web which correlates with several other applications.

Often the development of software is focused on technological sophistication, so developers often forget about users. Users with various backgrounds and characters have their own expectations for new software. Unmet expectations can become obstacles to performance. Assessment of the quality of applications is difficult to measure because there is no right size that can equalize people's perceptions of the quality of the application so that measurements of application satisfaction degrees are sometimes less accurate, ambiguous, imprecise and subjective. One approach method used to measure application quality is the Fuzzy concept. One way to find out the expectations of users of software products is to measure software in terms of quality [3]. By measuring the usability side, it will be able to know the value of the use of a software product both in terms of effective, efficient, and user satisfaction [4]. The value of the use of a product can be done by evaluating the usability side [5]. By conducting an evaluation, it will be able to assess system functions and their level of ease [6]. There are also several other factors that need to be considered to develop software that has good performance [7]. Users, in general, will tend to use software by looking at the factors how easily the software is used, how effective the processes that exist in the software support the work [8]. A device is considered easy to use for user experience, an understanding of the purpose of software and the context is also influential [9]. With all expectations fulfilled, users will tend to use software. Because of the very important measurement of software quality, Idrus et al (2018) conduct research to identify competency problems or challenges related to software testers. The results of this study are to propose a model for developing software testing competencies [10]. There are several models that can be used to measure software quality. Software Quality model was introduced by Evans and Marciniak (1987).

Another model was introduced by Deutsch and Wilis (1988). Their model has 15 factors. Another model was introduced by McCall. McCall's model has more practical advantages in classifying software requirements [11]. There are 6 distinguishing factors between the Deutsch's model and the McCall's mode. They are expandability, manageability, verifiability, safety, and survivability. And there are 2 distinguishing factors between Evans's model and McCall model, verifiability and expandability [12].

This study aims to measure user satisfaction in terms of fulfilling expectations for e-health visualization applications for pregnant women. The expectation of this study is to find out the results of the evaluation are xpected to make improvements both in terms of user interface design and process effectiveness. To evaluate, it can be done with a questionnaire, probabilistic, neural network, fuzzy, and other approaches [13]. The method used in this study uses questionnaires which are processed using the Fzzy approach.

A fuzzy approach was chosen because as a user it is possible to experience errors or things that are beyond the perception expected when using the software. A fuzzy approach is suitable to be used to measure uncertainty and tolerance to inaccuracies [14]. The use of graph scenario is used as an approach for making testing scenarios. This method was chosen because it is relatively easy to use and systematic, and does not require a long time [15].

II. LITERATURE REVIEW

Several studies related to the measurement of Fuzzy related research to measure usability are as follows: Testing using the fuzzy method is widely used to determine the level of hospital services and applications. The study was conducted by Yuen and Lau (2011). Yuen and Lau conducted research assessing the quality of software, carried out by a group of experts at different levels, consisting of 6 criteria and 27 sub-criteria used as attributes of software quality. The approach used is the Fuzzy Group Analytical Hierarchy Process. The results showed that Fuzzy can help in testing and measuring the level of software quality [16]. Pizzi (2013) conducted a study to estimate the quality of software using the Fuzzy approach. The results of the study are that the use of Fuzzy integration is superior in predicting and classification than [17].

Research related to software evaluation using Fuzzy method was also conducted by Ahmed et al (2008). The research aims to develop an integrated nd comprehensive strategy that is used to evaluate the process in software [18]. Another study was conducted by Pal and Battacherjee (2012). The research he conducted was to classify the quality of software using the fuzzy compatibility matrix and look for composition relations. The method used is the Maxmin method. The results obtained are fuzzy equivalent matrices that are formed can be used to determine the classification of software quality [19].

Setiawan et al (2018) conducted a research design development metric to improve software development. The results of the study showed that after 35 indicators were made applying which were made in the application development the number of bugs and errors found was reduced from 43% to 23% [20].

III. METHODOLOGI

Fuzzy sets represent certain conditions in fuzzy variables. In this study, we want to measure software quality in terms of fulfilling user expectations in 5 variables, and each variable in this study consists of 5 linguistic values, very low, low, enough, high, very high.

3.1. Fuzzification

Fuzzification converts input values in the form of values into a set of linguistic variables. Uncertain values and unclear values can be represented by membership functions, whose output is fuzzy value.

3.2. Graph Scenario

Black box testing is a testing method for testing software functionality. To be able to test properly, it is necessary to make a test case. In this study, the test case was made using a graph. The stages of research can be seen in Figure 1.

From Figure 1 can be seen, the stages of research conducted are: 1) Create a test scenario. 2) Testing-1. At this stage, the user tests the software in terms of functionality based on the test scenario that was made. 3) Determined the aspects of Quality Product. At this stage, variables are determined to be assessed. In this study, 5 variables were selected from 11 variables that existed in the product operation characteristics. The five variables chosen were correctness, reliability, efficiency, integrity, and usability. 4) Testing-2. At this stage, users make an assessment of the quality of software related to user expectations for a software product. 5) Fuzzy set. 6) Fuzzification. 7) Fuzzy Interference System. 8) Accurate 9) Conclusions.

Figure 1: Research Flow Diagram

Questioners are distributed to participants. Participants who tested this application were health workers in the North Jakarta area. This research is quantitative research. The quantitative approach is chosen because based on the data distributed through questionnaires to users.

The questionnaire is used to measure the product quality of the five variables. Details of a variable can be seen in Table 1. Assessments using questionnaires were conducted on a graded scale of ratings. Item questions on the questionnaire through validity testing and reliability testing. Item questions that do not meet the value, the questions are rejected

Variables	Description
correctness	all functions according to the specifications
reliability	the user assesses the application to the extent that it has failed
efficiency	the user assesses the application to the extent that it has increased resource use
integrity	the application has a high level of security so that it is not possible to be misused
usability	the user assesses the application to the extent that easy to use by the user

IV. RESULT AND DISCUSSION

4.1. Testing

To measure software quality, the first thing to do is to test the software functionality. At this stage, we also carry out validation and verification tests. Before conducting software testing, a test case scenario (TC) is made using graph. One of the makings of graph testing software can be seen in Table 2 and Figure 3.

Table 2 Scenario Test

Rules	Flow	Description
The precondition in point 2	2.1	a user enters to access all features of the application
	2.2	a user selects input data for pregnant women
	2.3	the system will display the data input form for pregnant women
	2.4	user entered data into the database successfully
	2.5	successfully
Alternative	2.4a	user enters incomplete or incorrect data

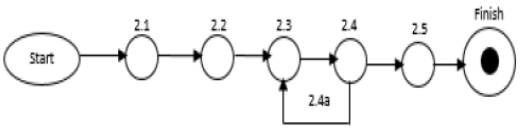

Figure 2: Graph includes data on pregnant women

Based on Figure 2, we can make TC2. Making other TCs is also done in the same way. Each TC will produce one or several test scenarios. The resulting test scenario will be used to perform software testing (testing-1). Users test software based on a test scenario by entering input and comparing the output to the reaction that should appear. The results of the data at that stage are processed. The test results are processed, which can be seen in Table 3.

Table 3 Results of Software Testing

No	Test Case	Scenario	The score of test item	Maximum Score	The Calculation on Results
1	TC1	1.1 – 1.2 – 1.2a 1.1 – 1.2 – 1.3 – 1.3a 1.1 – 1.2 – 1.3 – 1.4	416	450	92.44%
2	TC2	2.1 – 2.2 – 2.3 – 2.4a 2.1 – 2.2 – 2.3 – 2.4 – 2.5	277	300	92.33%
3	TC3	3.1 – 3.2 – 3.3 – 3.4a 3.1 – 3.2 – 3.3 – 3.4 – 3.5	278	300	92.00%
4	TC4	4.1 – 4.2 – 4.3	138	150	91.67%
5	TC5	5.1 – 5.2 – 5.3 – 5.4 – 5.5a 5.1 – 5.2 – 5.3 – 5.4 – 5.5 – 5.6	275	300	91.67%
6	TC6	6.1 – 6.2 – 6.3 – 6.4 – 6.5 – 6.6a 6.1 – 6.2 – 6.3 – 6.4 – 6.5 – 6.6 – 6.7	275	300	91,67%
7	TC7	7.1 – 7.2 – 7.3 – 7.4 – 7.5	137	150	91,33%
8	TC8	8.1 – 8.2 – 8.3 – 8.4a 8.1 – 8.2 – 8.3 – 8.4 – 8.5	275	300	91,67%
9	TC9	9.1 – 9.2 – 9.3	138	150	92.00%
10	TC10	10.1 – 10.2 – 10.3	138	150	92.00%
11	TC11	11 – 11.2 – 11.3	139	150	92.67%
12	TC12	12.1 – 12.2 – 12.3 12.1 – 12.2 – 12.2a	283	300	94.33%
Max Score			3000		
Total Score			2769		92.3%

From table 3, it can be seen that the average test case above has a percentage of 92.3%. This shows the functionality of this application to function properly.

The next step is to measure the application to see the quality of the application whether the application meets user expectations of the five variables, correctness, reliability, efficiency, integrity, and usability.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

4.2 Product Quality

The McCall model was introduced in 1997. McCall provided a method by classifying software requirements into software quality factors. The model consists of 11 quality factors. These factors are grouped into 3 categories [21]. They are product operation factors, product revisions factors, and product transitions factors. Product Operation is used to determine the operational characteristics of a system or application. Product operation factors consist of correctness, reliability, efficiency, integrity, and usability. Fuzzy processing is used to determine fuzzy set numbers, which are used for assessment. So, that the steps taken are determining fuzzy sets using TFN.

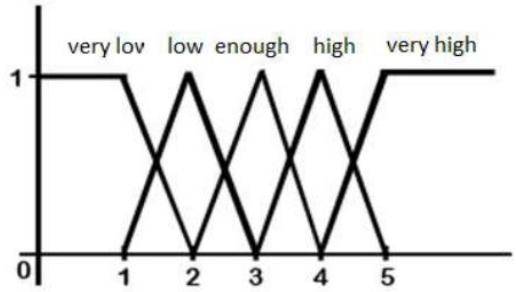

Figure 3: Graph includes data on pregnant women

Figure 4 can be used in determining the weight (score) that we use to calculate the value of fuzzification. They are very low with a value of 0, 1, 2. Value for low with the value 1, 2, 3. Value for enough with a value of 2, 3, 4. Value for the high with a value of 3, 4, 5. And value for very high with a value of 4 and 5. The way to get the composition value from fuzzy numbers is that all respondents use the arithmetic mean to obtain an average weighting score.

$$a_m = \frac{(a_{m1}+a_{m2}+a_{m3\ldots}+a_{mi})}{N} = b_m = \frac{(b_{m1}+b_{m2}+b_{m3\ldots}+b_{mi})}{N} = c_m = \frac{(c_{m1}+c_{m2}+c_{m3\ldots}+c_{mi})}{N} \quad (1)$$

Definition: Some definitions of the formula above are as follows. N is the number of population, Cm is the lower limit, αm is the middle limit, and 6m is the upper limit.

Fuzzification calculation process to get the value of fuzzy numbers, using the arithmetic mean to get the average weighting score. At the stage of testing-2, the assessment of software is carried out by the user in terms of user expectations for the software. After the questionnaire results from the respondent are collected, the next step is to recapitulate the results of the questionnaire. After that, data was done the calculation using the Fuzzy approach to get the lower limit, middle limit, an upper limit with the formula above. The results obtained after the calculation show the lower limit value (C) is 3.13; the middle limit value (A) is 4.13, and the upper limit value (B) is 4,88. The next step is to calculate the lower limit, middle limit, and upper limit of each dimension correctness, reliability, efficiency, integrity and usability. Results of quality product (QP) using fuzzy can be seen in Table 4.

Table 4 Results of QP Fuzzification Value Recapitulation

Question	Results of Fuzzification		
	TFN		
	C	A	B
C1	3.133	4.133	4.867
C2	3.1	4.1	4.833
C3	3.267	4.267	4.933
C4	3.1	4.1	4.867
C5	3.067	4.067	4.933
Correctness	3.133	4.133	4.887
R1	2.8	3.8	4.667
R2	3.133	4.133	4.9
R3	3.1	4.1	4.9
R4	3.067	4.067	4.833
R5	3.033	4.033	4.9
R6	3.2	4.2	4.833
Reliability	3.056	4.056	4.839
I1	3.067	4.067	4.767
I2	3.167	4.167	4.8
I3	2.933	3.933	4.767
Integrity	3.056	4.056	4.778
E1	3.333	4.333	4,967
E2	3.4	4.4	4,867
E3	3.3	4.3	4.967
E4	3.3	4.3	4.967
E5	3.167	4.167	4.967
Efficiency	3.3	4.3	4.98
U1	2.8	3.8	4.667
U2	3.133	4.133	4.9
U3	3.1	4.1	4.9
U4	3.067	4.067	4.833
U5	3.033	4.033	4.9
U6	3.2	4.2	4.833
Usability	3.056	4.056	4.839

The next step is made a graph based on the results in table 4. A graph was created can be seen in Figure 4.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

The graph in Figure 4 is a graph for variable correctness. From Figure 4 it can be seen that the lower limit value is 3.13, for the middle limit of 4.13 and the upper limit of 4.88 so that the application's correctness is assessed as an application that has a level of correctness high.From Table 4, we create a graph of each variable in the same way. The results obtained show that in terms of high-reliability variables with a value of the lower limit is 3.05, for the middle limit of 4.05 and the upper limit of 4.83. The integrity variable is high with a value of the lower limit is 3.05, for the middle limit of 4.05 and the upper limit of 4.78. The efficiency variable is high with a value of the lower limit is 3.3, for the middle limit of 4.3 and the upper limit of 4.98.and usability variable has a value of The efficiency side is high with a value of the lower limit is 3.28, for the middle limit of 4.28 and the upper limit of 4.95.

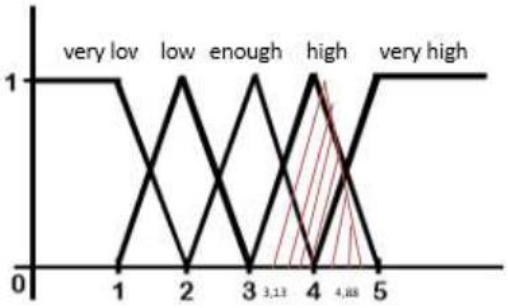

Figure 4: Correctness Variable Graph

After the graph is formed, the next step is to determine the rule base. This rule base determination is part of the stages of testing and accuracy. Rule base determination is based on fuzzy rules, fuzzy set theory concepts, and fuzzy logic concepts. Sample rules base can be seen in Table 5.

Table 5 Sample Rules Base

No	Variables Input					Output (QP)
	C	R	I	E	U	
1	very low	very low	vey low	very low	very low	very low
2	very low	very low	vey low	very low	low	very low
3	very low	very low	vey low	low	low	very low
:						
:						
n	very high	very high	vey high	very high	very high	very high

In the last stage, testing is carried out to see the level of accuracy. At this stage, the calculation is still done manually. Based on table 5, the output of the quality of the overall quality product of the e-health application product is high or good. To measure the accuracy of the results performed calculations using the formula:

$$\text{accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (2)$$

Definition: Some definitions of the formula above are as follows. TP is the number of correct results, TN is the number of a correct absence of a result, FN is the number of a missing result, and FP is the number of unexpected results

V. CONCLUSION

Based on the above research, fuzzy methods can be used to measure the level of software quality. There are 5 levels of fuzzy values produced in this study, very low, low, sufficient, high, and very high. In this study, overall e-health applications have a value in the range of 3,4,5, indicating that the quality of this application is good so that it can be used/implemented. Before measuring software quality, functional testing must be done first to ensure that all software functions properly.

REFERENCES

1. WHO, “World Health Statistics,” WHO, vol. 8, no. 33. WHO Press, Switzerland, p. 44, 2014.
2. Kementrian Kesehatan RI, Profil Kesehatan Indonesia 2015. Jakarta: Kementerian Kesehatan Republik Indonesia, 2016.
3. J. Offutt, “Web Software Applications Quality Attributes,” Qual. Eng. Softw. Technol. (CONQUEST 2002), pp. 187–198, 2002.
4. V. Venkatesh, H. Hoehle, and R. Aljafari, “A usability evaluation of the Obamacare website,” Gov. Inf. Q., vol. 31, no. 4, pp. 669–680, 2014.
5. M. Xenos, “Usability Perspective in Software Quality Usability as a Software Quality Factor,” Perspective, vol. 2, pp. 523–529, 2001.
6. H. Yahya and R. Razali, “Usability-Based Framework for Electronic Government,” ARPN J. Eng. Appl. Sci., vol. 10, no. 20, pp. 9414–9423, 2015.
7. Roger S. Pressman, Software Engineering : Practitioner ’ S Approach, 5th ed. McGraw Hill, 2001.
8. N. Jazdi, N. Oppenlaender, and M. Weyrich, “Quantification of the quality characteristics for the calculation of software reliability,” IFAC-PapersOnLine, vol. 49, no. 30, pp. 1–5, 2016.
9. A. Dix, J. Finlay, G. D. Abowd, and R. Beale, “Evaluation Techniques,” in Human-Computer Interaction, 3rd ed., Harlow, England: Person Education Limited, 2004.
10. H. M. Idrus and N. Ali, "Towards Development of Software Testing Competency Framework to Empower Software Testers ' Profession," vol. 7, pp. 749–754, 2018.
11. R. Fitzpatrick, “Software quality: definitions and strategic issues Part of the Computer and Systems Architecture Commons,” pp. 0–34, 1996.
12. B. Youness, M. Abdelaziz, B. Habib, and M. Hicham, “Comparative Study of Software Quality Models,” no. November 2013, 2016.
13. R. Nagpal, D. Mehrotra, and P. K. Bhatia, “The State of Art in Website Usability Evaluation Methods,” in Design Solutions for User-Centric Information Systems, no. November, USA: IGI Global, 2017, pp. 275–296.
14. N. Abbasy, “Fuzzy Linear Regression Approach for Uncertainty Modeling in Power System Fuzzy Linear Regression Approach for Uncertainty Modeling in Power System State Estimation,” Proc. 6th WSEAS/IASME Int. Conf. Electr. Power Syst. High Voltages, Electr. Mach. Tenerife, Spain, no. December 2006, pp. 142–147, 2006.
15. J. Ryser and M. Glinz, “Using Dependency Charts to Improve Scenario-Based Testing Depicting and Managing Dependencies between Scenarios,” TCS‘2000 Washington, D.C.
16. K. K. F. Yuen and H. C. W. Lau, “A fuzzy group analytical hierarchy process approach for software quality assurance management: Fuzzy logarithmic least squares method,” Expert Syst. Appl., vol. 38, no. 8, pp. 10292–10302, 2011.
17. N. J. Pizzi, “A fuzzy classifier approach to estimating software quality,” Inf. Sci. (NY)., vol. 241, pp. 1–11, 2013.
18. F. Ahmed, L. F. Capretz, and J. Samarabandu, “Fuzzy inference system for software product family process evaluation,” Inf. Sci. (NY)., vol. 178, no. 13, pp. 2780–2793, 2008.
19. J. Pal and V. Bhattacherjee, “Application of Fuzzy Clustering on Software Quality using Max-min Method,” Procedia Technol., vol. 6, pp. 67–73, 2012.
20. R. Setiawan, Z. E. Rasjid, and A. Effendi, “Design Metric Indicator to Improve Quality Software Development (Study Case: Student Desk Portal),” Procedia Comput. Sci., vol. 135, pp. 616–623, 2018.
21. L. A. Zadeh, “Fuzzy sets.” Information and Control, California.
22. D. Galin, Software quality assurance: from theory to implementation, 1st ed. Harlow, England: Pearson Addison Wesley, 2004.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)