FQSCO: A FLEXIBLE MODEL FOR MEASURING CUSTOMER SATISFACTION ON SOFTWARE BASED PRODUCTS AND SERVICE

EZEKIEL U. OKIKE

University of Botswana, Gaborone , University of Botswana
E-mail: euokike@gmail.com

Abstract - Many software products and services deployed in user environments at times fail to meet user needs satisfactorily. This may be due to the fact that the product or service failed to meet user requirements from the outset (inception) of the Information Systems (IS) project. This study proposes a Flexible Qualifier Weighted Customer Opinion with Safeguard Estimates (FQWCOS) model for measuring the satisfaction of users of software products and services. The FQWCOS model is a variant of the Qualifications Weighted Customer Opinion with Safeguard questions (QWCOS). The FQWCOS model was verified with empirical data using samples from 40 users of ASAS software product. Descriptive statistics were also used to obtain the frequencies, mean values, relative frequencies, standard error, and standard deviation. From these values, it was possible to compute the normalized score of customer opinion Oi and the external measures E for QWCOS and Ei (i=1-4) for FQWCOS were computed. Results from the study reveal that there was no difference between the external measures for QWCOS and FQWCOS. However, the result suggest that external measures were higher when standard error (SE) was used to obtain the measures at different levels 31.58, 19.79, 21.76, 35.69 and 31.06 than when external measure was computed using standard deviation (STD) which yielded the values 4.99, 3.13, 3.44, 5.64 and 4.07. We conclude that FQWCOS and QWCOS yield the same values probably due to small sample used. However, FQWCOS provides a flexible and simple approach, and reveals the need to use the standard error instead of standard deviation since this yields higher magnitude values appropriate for expressing external measuresin percentages.

Index Terms - Software quality, external measurement, customer satisfaction, flexible model.

I. INTRODUCTION

Software metrics presently preferred to be called software measurement is concerned with measurements in Software Engineering. The preference of the term “measure’ to the term ‘metric’ is due that the fact that the term ‘measure’ supposedly emphasizes that the collection of measurements preserve relationships that exist between the entities being measured. This is the approach of the science of measurement otherwise called measurement theory[okike, 2007 Thesis; Okike, 2007 JAIST].

Software applications today are becoming more complex, and organizations cannot afford to neglect the likely consequences of the failure of their Information Systems (IS). Among these consequences are economic losses, and even threats to human lives. Therefore, a means of effectively measuring the quality and reliability of software products is needed. On the other hand, effective use of measurement instruments in software products and services benefit an organization in the following ways [okike, 2007 phd]:

1. Estimating the cost and schedule of future projects
2. Evaluating the productivity impacts of new tools and techniques
3. Establishing productivity trends over time
4. Improving software quality
5. Forecasting future staffing needs
6. Anticipating and reducing future maintenance requirements
7. Setting design standards for an organization

A. Statement of the Problem

The need to continuously measure customer satisfaction as a dominant quality in use factor for software products and services cannot be over emphasized. Although there are models for measuring customer satisfaction such as …. , these models are not based on measurement theory and may not be appropriate to software products and services. For instance, the Qualifications Weighted Customer Opinion with safeguard questions (QWCOS) is statistically good but does not allow for flexible customerqualifiers, which should have been indeed quantifiers. Therefore, it becomes very imperative to evolve a model formeasuring customer satisfaction which allows for adjustable quantifiers.

B. Aims and Objectives

The aim of this study is to propose a model for measuring satisfaction as a quality indicator in software products and services, hence, measure the effect of customer satisfaction in delivered software products or services using empirical data. The rest of this paper counts of four sections. Section 2 presents a review of related work. Section 3 presents the methodology of this study. Section 4presents the empirical study, while section 5 is the summary and conclusion.

II. LITERATURE REVIEW

A. Software Quality Measurement

Measuring quality in software products or service may be done in two ways namely:

1. Measuring internal attributes of software at code

level.

2. Measuring external attributes of software in a user environment.

A software quality model that represents these approaches is evident in figure 1.

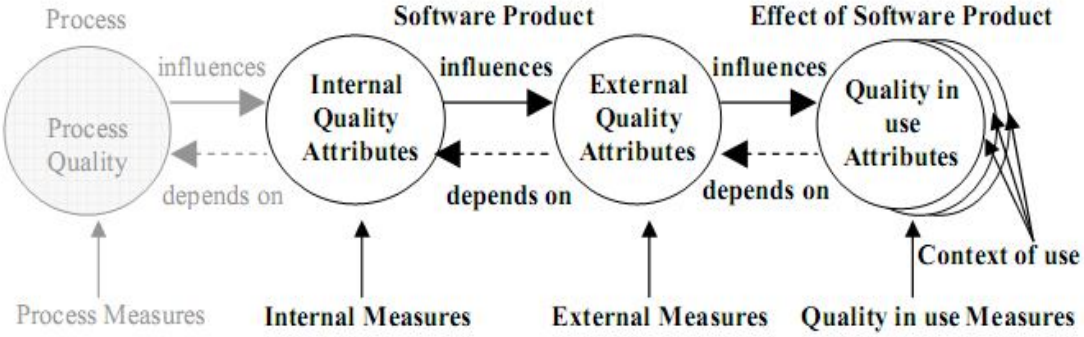


Figure 1: Internal and external Software quality model.

Internal Software Attributes (Code Level measurement).

Measuring Software Product Quality from Code level perspectives usually takes into account the internal software attributes which make up the structural complexity of a software program . Software complexity has multiple facets, including algorithmic complexity and structural complexity [7]. The structural complexity of a program has been defined as the organization of program elements within a program (Gorla and Ramakrishnan, 1997) . Measures of structural complexity are internal attributes of software, that is, they are specific to the software artifacts. Figure 2 identifies seven sources of software complexity namely: software, module cohesion, module coupling, control structures, algorithm, data structures, and nesting level (ibid). These are the recognizable and measurable internal software quality attributes.

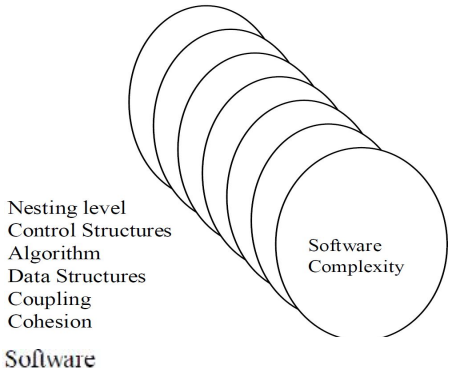


Figure 2: Different levels of software complexity.

External Software Attribute Measurement

External software quality attributes are those attributes of the software observable from use in the user environment as opposed to attributes inherent in the software codes. External product attributes depend on both the product behavior and the environment. Examples of external software attributes are usability, integrity, efficiency, testability, reusability, portability, interoperability, understandability, maintainability, reliability e.t.c. The ISO 9126 quality model in figure 3 is an example of external product measures.

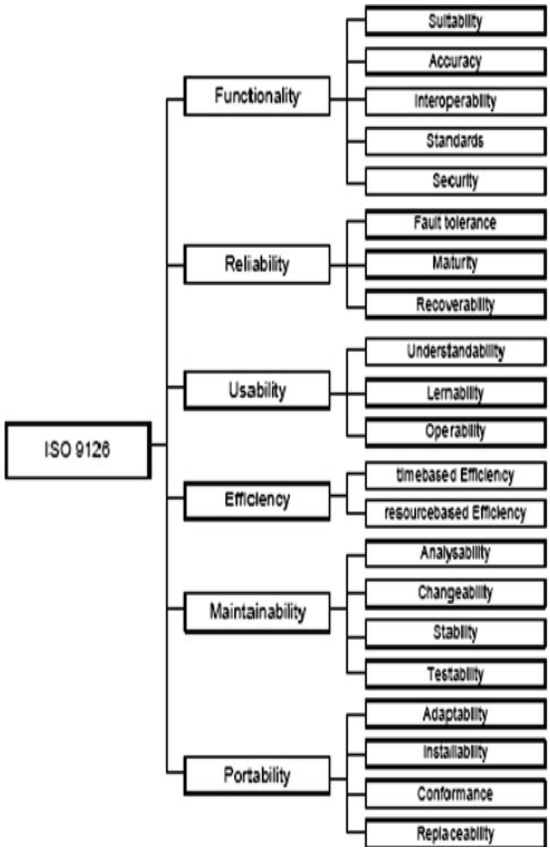


Figure 3: External Software Quality attributes ISO/IEC 8126

As illustrated in Figure 1, internal quality attributes influence external quality attributes which in turn influence quality in use. On the other hand, quality in use depend on external quality attributes, which in turn depend on internal quality attributes. This study is concerned with measuring customer satisfaction

based on external quality attributes, that is, based on the users perception of the software in the user environment.

B. Measuring Customer Satisfaction

Customer (User) Satisfaction in the context of this paper means the act of satisfying the needs of the user of a software product or service. Hence, satisfying a user implies fulfilling the desires or needs of the user with respect to the product or service. In general the business domain views customer satisfaction as the core of human experience which reflects our liking of a company's business activities (or products). Therefore, a high level of customer satisfaction are strong predictors of customer retention, customer loyalty, and product or (service) repurchase [15].

Providing customer satisfaction could be a quality criteria or requirement for software products or services. In this case, customer satisfaction means meeting all the customer requirements. According to Asher[10], customer satisfaction for a product or service may be measured by answering the following questions:

1. Did the provider keep all promise about the product or service to the user?
2. Did the user get good service from the product or service?
3. Did the provider do everything to meet the user needs?
4. Was the product or service delivered on schedule?
5. Did the provider start for the users' problem rather than the product or service?
6. Are the providers aware of the all the user needs, current and potential? Did they ask the users for their needs?
7. Did the provider notify the user of any changes to the product or service?
8. How many providers team members understand the needs of the users?
9. Can the users depend on the quality of the product or service?
10. Are providers interested in how users use the product or service?
11. From the forgoing, it is clear that the end product of every service rendered, or product usage is customer (user) satisfaction. Customer satisfaction is thus a form of evaluation of a product, in this case software. Such an evaluation, which among other factors measures user's perceptions of a product, could be useful in generating information that could eventually lead to improvement of such a product. To measure customer's satisfaction, this paper proposes a customer satisfaction model, which takes into account ISO quality models as shown in figures 2-4 above. We demonstrate the usefulness of the proposed model with an empirical study of customer satisfaction on software product for academic management and administration of students grades from the various courses done in a university.

C. Academic Students Administration System (ASAS)

The Academic Student Administration System (ASAS) is a generic software product. Generic products are sold on the open market to any customer who is able to buy the product, as opposed to bespoke (or customized) software products which are commissioned and developed for a particular customer or user. With generic products such as the ASAS software tool, purchasing organizations may sometimes reconfigure the software to suite the environment where it must be used. However, one peculiar challenge with generic products that require further configurations before use is that the IT personnel usually would do the configuration based on their technical knowledge without having to involve the users for appropriate user requirements input. This approach should create a knowledge gap between requirements analysis and software design as usual. Therefore, there is often the need for measuring user (customer) satisfaction on generic products such as the ASAS system to ascertain their usefulness to users or otherwise in the environment from time to time.

III. THE EMPIRICAL STUDY: MODELING AND MEASURING CUSTOMER SATISFACTION ON ASAS SOFTWARE

A. The Method

The study employs two models for measuring customer satisfaction namely the Qualifications WeightedCustomer Opinion with Safeguards (QWCOS) model and a Flexible Qualifier Weighted Customer Opinion with Safeguards (FQWCOS) model proposed in this study. QWCOS techniques estimate the result of the external measurement (EM) , which is based on O_i (the normalized score of customer opinion), the E_i (qualifications of customer i) and the use of a number of control questions (safeguard questions S where S_T is the total number of control questions, from which the customer i (from a total number of n customers) has responded correctly as at S_i [5]:

$$EM = QWCOs = \sum_{i=1}^{n}\left(Oi \,.\, Ei \,.\frac{Si}{St}\right) / \sum_{i=1}^{n}\left(Ei \,.\frac{Si}{St}\right) \qquad (1)$$

In this study, we define a flexible external measure M_E having a flexible Qualifier Weighted Customer Opinion with Safeguard estimates (FQWCOS).

Formally, define External Measure (M_E) as a quadruple (Σ, O_i, α , β). Hence,

$$M_E = \sum_{k=1}^{n}(Oi.\alpha j \,.b) / \sum_{k=1}^{n}(\alpha j \,.b) \qquad (2)$$

Where:

Oi = normalized score of user i opinion using mean

Standard Error (S.E) or Standard Deviation (STD)
α = Flexible qualifier of user
β = relative frequency
Hence $M_E = Oi$ (best measured using S.E).

From Eq.(2), external measure M_E may be applied in any number of steps as desired by varying the flexible qualifier α_j (α_j;j=1- n). In this study, quantity (α_j is applied in four steps as shown in equations (3) to (6):

Step 1 (using the highest academic qualifications of users as in Eq 1)

$$M_{E1} = FQWCOs = \sum_{i=1}^{n} (Oi . \alpha 1. \beta) / \sum_{i=1}^{n} (\alpha 1. \beta) \quad (3)$$

Step 2 (using IT technical competence of users)

$$M_{E2} = FQWCOs = \sum_{i=1}^{n} (Oi . \alpha 2. \beta) / \sum_{i=1}^{n} (\alpha 2. \beta) \quad (4)$$

Step 3 (usingyears of experience in using the product)

$$M_{E3} = FQWCOs = \sum_{i=1}^{n} (Oi . \alpha 3. \beta) / \sum_{i=1}^{n} (\alpha 3. \beta) \quad (5)$$

Step 4 (using users over all satisfaction with the product)

$$M_{E4} = FQWCOs = \sum_{i=1}^{n} (Oi . \alpha 4. \beta) / \sum_{i=1}^{n} (\alpha 4. \beta) \quad (3)$$

The essence of the four steps is to observe the effect of variable α_j in term of the over all magnitude of quantity M_E as the highest measure of customer satisfaction of the product or service in a userenvironment. Both the QWCOS and the FQWCOS models were verified with data collected from 40 users of ASAS software product using a questionnaire instrument. Results of both model were comparatively analyzed to determine which of the models best measures customer satisfaction based on some threshold.

B. Result and Discussion

The results of measuring user satisfaction using the QWCOS and the FQWCOS models are shown in tables 1 and 2. From Table 1, using 40 respondents and the safe guard questions S where S_T is the total number of control questions, from which customer i, (from a total number of n customers) has responded correctly as at S_i [5], the value of O_i was computed using either the mean standard error or the standard deviation. The magnitude of O_i using mean standard error (S.E) is higher than O_i using the standard deviation. Furthermore, from Table 2, the external measures for QWCOS and FQWCOS remain the same. Hence, QWCOS and FQWCOS are comparatively the same.

TABLE 1: Measuring users satisfaction with QWCOS and FQWCOS

QUESTIONS	N = 40	FREQ. 4-5 1-3	MEAN (μ)	REL. FREQ	S.E / STD	$Oi = \frac{\mu i - \mu}{S.E}$	$Oi = \frac{\mu i - \mu}{Std}$
1 Level of satisfaction with ASAS	40	6 34	2.6000	1.00	.17097 1.08131	31.58	4.99
2 Pretty Satisfied with ASAS	40	19 21	2.4250	1.000	.28168 1.7849	19.79	3.13
3 Pretty certain I can Use ASAS without assistance	40	14 26	2.2750	1.00	.26309 1.66391	21.76	3.44
4 I have truly enjoyed using ASAS	40	7 33	2.7000	1.000	.14850 .93918	35.69	5.64
5 I can recommend ASAS to other institutions	40	15 25	3.2308	1.000	.15351 .95866	31.06	4.97

6 Experience in years in ASAS	40	30 10	3.9250	1.00	.15766 .99711	25.85	4.09
7 IT competence	40	11 29	1.4750	1.000	.12904 .81610	50.57	8.00

TABLE 2: Comparative analysis between QWCOS and FQWCOS

QUESTIONS	QWCOS	FQWCOS	$Oi = \frac{\mu i - m}{S.E}$	$Oi = \frac{\mu i - m}{Std}$
1 Level of satisfaction with ASAS	EM	$M_{E1}/M_{E2}/M_{E3}/M_{E4}$	31.58	4.99
2 Pretty Satisfied with ASAS	EM	$M_{E1}/M_{E2}/M_{E3}/M_{E4}$	19.79	3.13
3 Pretty certain I can Use ASAS without assistance	EM	$M_{E1}/M_{E2}/M_{E3}/M_{E4}$	21.76	3.44
4 I have truly enjoyed using ASAS	EM	$M_{E1}/M_{E2}/M_{E3}/M_{E4}$	35.69	5.64
5 I can recommend ASAS to other institutions	EM	$M_{E1}/M_{E2}/M_{E3}/M_{E4}$	31.06	4.97
6 Experience in years in ASAS	EM	$M_{E1}/M_{E2}/M_{E3}/M_{E4}$	25.85	4.09
7 IT competence	EM	$M_{E1}/M_{E2}/M_{E3}/M_{E4}$	50.57	8.00

CONCLUSION

Both models QWCOS and FQWCOS are comparatively the same. However, FQWCOS computation appear more straight forward and also flexible with the explanation provided in this study. FQWCOS approach suggest that the impact of the external measure of satisfaction on the selected software product yields higher values when the normalized score of customer opinion $O_i$is computed using the mean standard error (S.E) than using the standard deviation as shown in table 2.

REFERENCES

[1] B. Berenbach, J. Kazmeier, D. J. Paulish, and A. Rudorfer, "Software and Systems Requirements Engineering in Practice," New Delhi: Tata McGraw Hill, 2009, pp. xxv+321.

[2] B. Boehm, "Software Risk Managements and Principles," IEEE Software 8(1): 32-41, 1991

[3] C. Y. Lin, M. Wu, J. A. Bloom, I. J. Cox, and M. Miller, "Rotation, scale, and translation resilient public watermarking for images," IEEE Trans. Image Process., vol. 10, no. 5, pp. 767-782, May 2001.

[4] D. Duggan. "Enterprise Software Architecture and Design: entities, Services, and Resources." USA:Wiley, 2012, pp. xxiii+481

[5] D. Stravrinoudis, and M. Xenos, "Comparing internal and external software quality measurements," Proceedings of the 8th Joint Conference on Knowledge-Based Software Engineering, Piraeus Greece: IOS Press, August 25-28, 2008, pp. 115-124.

[6] IEEE, "Glossary of Software Engineering", USA:IEEE, 1990

[7] E. U. Okike, "Measuring Class Cohesion in Object-Oriented Systems Using Chidamber and Kemerer Metrics and Java as Case study," P.HD Thesis, Department of Computer Science, University of Ibadan, xvii+133 pp, 2007.

[8] I. Jacobson, G. Booch, I Rumbaugh, "The Unified Software Development Process," Addison-Wesley Longman Publishing Co Inc., Boston, 1999.

[9] J. U. Duncombe, "Infrared navigation—Part I: An assessment of feasibility," IEEE Trans. Electron Devices, vol. ED-11, pp. 34-39, Jan. 1959.

[10] M. Asher. Measuring customer satisfaction, The TQM magazine. Vol. Iss. 2. Pp93-97. 1989.

[11] M .I. Kamala, and T. Tamai, " How Does Requirements Quality Relate to Project Success or Failure?" Proceedings of the International Requirements Engineering Conference (RE'07), 2007.

[12] N. E. Fenton, and S. L. Pfleedger, " Software Metrics: A rigorous & Practical Approach," 2nd ed., London:PWS, 1997, pp. xii+638.

[13] N. Gorla, and R. Ramakrishnan, " The Effect of Software Structure attributes on Software Development PProductivity," Journal of Systems and Software, 36, 2:191-199, 1997.

[14] P. Ralph, " The illusion of Requirements in Software Development," Requirements Engineering (2013) 18, pp. 293-296.

[15] S. M. Smith, " Customer Satisfaction Survey Questions: 5 Sample Templates you can use right away," www. Qualtric.com retrieved 26 July, 2018.

★★★