International Journal of Engineering and Information Systems (IJEAIS)
ISSN: 2643-640X
Vol. 5 Issue 3, March - 2021, Pages: 29-34

A Survey on the Relationships between Quality Attributes of the ISO9126 Model

Abdelkareem M. Alashqar

Faculty of Information Technology, The Islamic University of Gaza
Gaza, Palestine
aashgar@yahoo.com

Abstract — *Considering quality attributes when developing software is inevitable. Quality attributes are also known as non-functional requirement that requested by software stakeholders. ISO9126 is one of the best-known quality models that includes some important quality attributes. It is not practical to develop software applications without considering the relationships between quality attributes. This is because that achieving one quality attribute will often enhance or hinder other quality attributes in the software being developed. Many research works have studied the relationships between quality attributes. In this paper we introduced a literature survey for studying and defining the relationships between quality attributes that defined by the ISO9126 quality model. We also proposed a matching mechanism between equivalent attributes that expressed differently in the current published work.*

Keywords— Software Quality; Quality Attributes Relationships; ISO9126; Tradeoffs

1. INTRODUCTION

Producing high qualified software that meets customer expectations is inevitable. Software quality is defined by ANSI Standard [1] as: "Quality is the totality of features and characteristics of a product or a service that bears on its ability to satisfy the given needs". Specific detentions for the features and characteristics of software have been introduced as quality attributes by various quality evaluation models such as the ISO/IEC 9126 model [2]. Quality attributes are sometimes called non-functional requirements that should take high consideration throughout the software development process. However, achieving the non-functional requirements is closely coupled with the design of software architecture [3]. This will encourage software developers to evaluate the achievement of quality attributes early in the architectural design phase because it is more convenient and cost effective than evaluating them after implementing the software. However, a quality attribute cannot be performed in isolation without considering its relation with other required quality attributes in the software being developed. This is because that an enhancement of one quality attribute always results in an enhancement or a hindrance of other quality attributes. The authors in [4] have provided an analysis method for understanding the descriptions for quality attributes in order to define the relationships between them. There are some research studies that explored the relationships between quality attributes. The type of research study can be dependent on the technique used whether it is experience-based, model-based or mathematical based. A limited survey on the relationships between quality attributes is found in [5]. However, to our knowledge there is no comprehensive work focus on investigating these relationships for the ISO9126 model that takes into account some important factors such as the technique used in defining the type of the relationship as well as the matching mechanism between different definitions of the quality attributes.

The rest of paper is organized as follows: Section 2 presents related works. Section 3 describes the proposed method. Section 4 discusses the relationships between quality attributes that defined by the ISO9126 model. Section 5 concludes this paper and suggests future work.

2. RELATED WORK

Although there are numerous definitions for quality found in the literature, the majority of them focus on one or two main ideas. The first one is that products and services meet definite specification, and the other is how much these products and services meet customer expectations [6]. Nonetheless the aforementioned definitions give a qualitative and flexible perspective, while the software quality models that proposed in the literature provide with more quantitative and specific perspective of quality definition. The ISO/IEC 9126 model [2] is considered a well-known software quality model that defines six quality attributes and their descriptions. The quality attributes of this model are: functionality, reliability, usability, efficiency, maintainability and portability. The sub-characteristics for each one of these quality attributes are also defined. There are other quality models found in the literature such as McCall [4] and Boehm [6]. However, the ISO9126 model is appropriate for every type of software [2]. Furthermore, it is considered as the basis for other quality models [7]. Practically, developers cannot deal with a quality attribute without considering its relationship with other quality attributes when developing software. This is because that achieving one quality attribute will often enhances or hinders other quality attribute in this software. Most of exiting works consider a 3-point scale for examining a relationship between two quality attributes; positive "+", negative "-" and neutral "0". A positive relationship indicates that an enhancement of one quality attribute will also result on an enhancement of the other, a negative relationship indicates that an enhancement of one quality attribute will result on a hindrance of the other, and a neutral relationship indicates that no effect of one quality attribute on the other and in this case they are considered independent. The negative relationship also means that there is a conflict or a tradeoff between the two quality attributes. Many works have discussed the relationships between software quality attributes. With regard to the types of techniques used, the authors in [8] have discussed three types of techniques for understanding the tradeoffs between quality attributes. These techniques are explained next.

1. Experience-Based Techniques: These techniques are considered qualitative in nature and use various methods such as empirical studies including surveys and questionnaires, expert judgment, industry experience.

2. Model-Based Techniques: these techniques use graphical representations for depicting the relationships between quality attributes. Models are often easy to communicate and understand.

3. Mathematical-Based Techniques: these techniques depend on building mathematical formulas then provide given useful inputs for them to produce best solutions about the relationships between quality attributes.

Most of research works regarding the above techniques adopts the experience-based especially by surveying the literature such as in [9] or surveying experts from academia such as in [10] and industry such as in [11]. An experience-based survey has been conducted by [12] for defining the relationships between quality attributes in the domain of web applications. The authors in [13] have asked the domain experts in the field of dependable embedded system to judge firstly the impact of architectural tactics on quality attributes in order to provide indicators on quality attributes relationships. However limited research works have used the model-based and mathematical-based techniques.

A type of model-based technique is the Non-Functional Requirements (NFR) framework that has been introduced by [14] which uses a diagram called Softgoal Interdependency Graph (SIG) in order to visualize the tradeoffs among quality attributes. Using SIG, the quality attribute is firstly shown on the diagram then connecting it to operationalizations that support or hinder other definite quality attributes. The effect of operationalizations is based on the knowledge that additionally provided in the SIG.

For the type of mathematical-based techniques, we have previously published work [15] that adopted the Choquet Integral approach for studying the relationships between the ISO9126-based quality attributes. The direct impact of architectural tactics on quality attributes are initially evaluated. The study has introduced new types of relationships such as two quality attributes interact in complementary way or substitutive way. Furthermore, the degree of relationships has been calculated. With regard to the quality attributes tradeoff, the authors in [16] have analyzed many published materials from well-known databases for examining the common approaches used by researchers in studying quality attributes tradeoffs. These approaches include Analytical Hierarchy Process (AHP), models, algorithms, Architecture Tradeoff Analysis Method (ATAM), metrics, expert opinion, Quality Function Deployment (QFD) and prototyping. However, the study does not provide collective results for quality attributes' relationships.

It important to denote that while the experience-based and model-based techniques can examine only the type of a relationship between quality attributes within a 3-points scale usually such as positive, negative or neutral as stated before, the mathematical-based technique surpasses them in that it examines the type of the relationship as well as the degree of this relationship. In other words, it can quantify it in numerical values.

3. RESEARCH METHOD

A literature survey is mainly used for understanding the relationships between the ISO9126-based quality attributes. Before conducting the survey, it is very important to formulate some hypotheses. The following formulation will be handled through the survey:

1. A relationship between two software quality attributes is often found when developing software.

2. The type of relationship can be positive or negative. The former means that enhancing one attribute will also enhances the other, while the later means that enhancing one attribute will hinders enhances the other.

3. Different methods were conducted in understating the relationships between quality attributes.

4. There is lack of research work that focus on understanding the relationships between the six quality attributes of the ISO9126.

5. We have conducted a literature survey using a sample of well-known research repositories. The survey covers published content including research papers in journals and conferences, published books, and thesis for graduate students.

International Journal of Engineering and Information Systems (IJEAIS)
ISSN: 2643-640X
Vol. 5 Issue 3, March - 2021, Pages: 29-34

4. DEFINING QUALITY ATTRIBUTES' RELATIONSHIPS

Our survey will focus only on the relationships for the quality attributes defined by the ISO9126 model. As shown in **Fig. 1** ISO9126 model encompasses six main quality attributes which are: functionality, reliability, usability, efficiency, maintainability and portability. Each one of these quality attributes also has sub-characteristics that provide more details to main quality attribute.

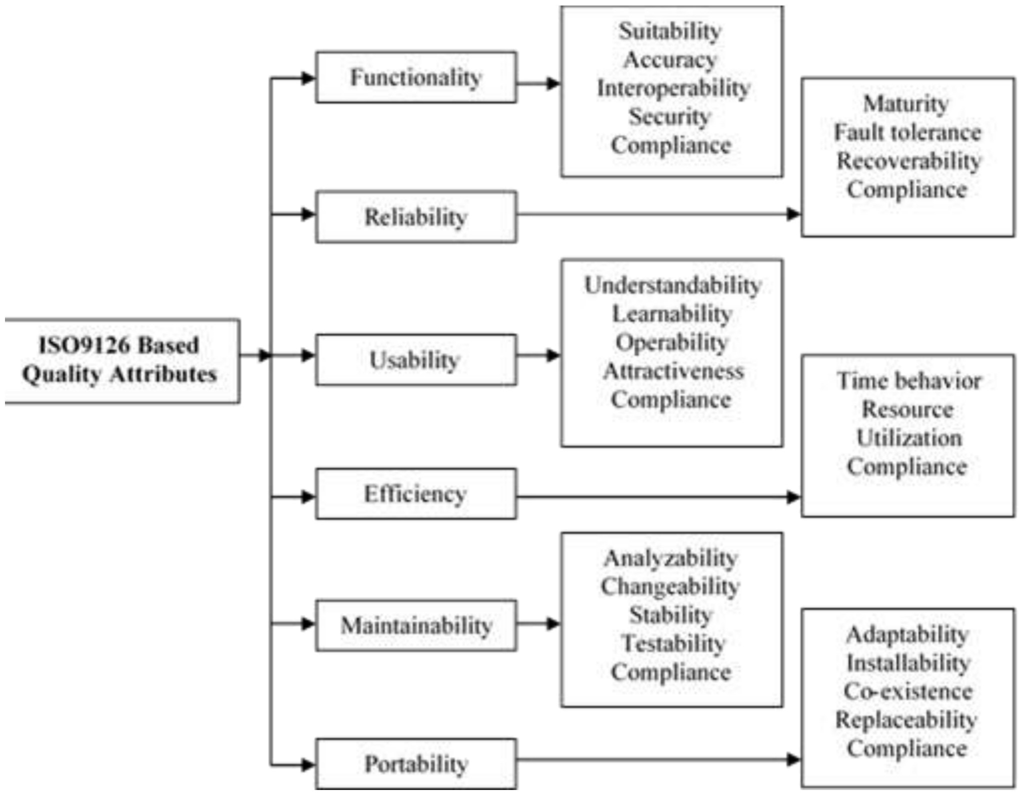

Fig. 1. The ISO9126 Software Quality Model

The descriptions for each of the main quality attributes are as follows [2][17]:

- Functionality: The capability of software for providing the needed functions while it is operational under stated conditions.
- Reliability: The possibility of software to execute without failure for a specific period of time. The average of correctly completed operations and the average of system running before fail are measurements of software reliability.
- Usability: The capability of the software to be easy to use, learned, understood and attractive effectively by users.
- Efficiency: Is a measure of how well the software utilizes the available resources such as processor capacity, disk space, memory, or communication bandwidth. Efficiency is related to performance.
- Maintainability: The capability of software to be easily understood in order to be modified for new changes or fixating existing defects. Changes often required because of business and technology change.
- Portability: The capability of the software to be migrated from one operational environment to another.

However, there are no universal definitions for software quality attributes. So, we have provided a matching mechanism between different quality attributes that have compatible descriptions. This matching mechanism is very important for finding more quality relationships with regard to ISO9126 model especially when these quality attributes handled differently by researchers. The matching results are shown in Table 1 that represents the mapping between equivalent quality attributes that expressed differently by researchers. The explanation of the mapping is as follows:

- Correctness is considered an equivalent to functionality, which means the extent to which a software meets its specifications and achieves the user expectations [4].

- Interoperability represents a sub-characteristic of functionality and means the ability of software to communicate and interact with other systems.

- Flexibility is considered an equivalent to maintainability which means extensibility and measures the capability of how easy to add new functionality and this is a main type of changeability achieved by maintenance developers and represent a sub-characteristic of maintainability.

- Performance is considered an equivalent to efficiency. Efficiency is related to performance, if software largely consumes the available system resources then users will encounter reduced performance which indicates a visible inefficiency [17].

Table 1: Mapping between equivalent quality attributes

ISO9126 Sources	Functionality	Reliability	Usability	Efficiency	Maintainability	Portability
McCall [4]	correctness, interoperability					
Wiegers [17]	interoperability				modifiability	
Zulzalil [12]						
Henningsson [9]	correctness					
Boehm [6]	interoperability			performance		
Deutsch [17]	correctness, interoperability				flexibility	
Berander [8]	correctness				flexibility, analyzability, changeability	

When the number of examined quality attribute N is six then the number of pairs' relationships for *n* can be calculated by the formula: $n*(n-1)/2$. So, the six quality attributes of the ISO9126 model will produce 15 relationships.

Based on our literature survey of seven research sources that adopted the approach of experience-based technique, in addition of considering our proposed matching mechanism in finding equivalent quality attributes, the results of the relationships between the ISO9126 quality attributes for seven research sources are shown in Table 2. In this table the "pos" abbreviation is assigned in the cell for positive relationship between two quality attributes, "neg" is assigned for negative relationship and the empty cell means that the examined source has not find a relationship between the two quality attributes.

As shown in the Table 2, there is no relationship has been examined by all surveyed sources. However, there is a considerable agreement on nine relationships which highlighted in bold text in Table 2. Each one of these relationships has been agreed by at least three examined sources despite of the approach used in defining the relationships whether it is literature review, academic or industry approach. Our survey on the examined sources reveals that functionality has positive relationship with all other quality attributes except that it has negative relationship with efficiency. In addition, efficiency has negative relationship with all other quality attributes. The results also show that the relationships of reliability and efficiency, reliability and portability, usability and portability, and usability and maintainability have been examined by only one source which are highlighted by italic text. It is important to denote that the results of our examined sources also show that there is no agreement on only two relationships which are the relationships between maintainability and each of reliability and portability.

For the mathematical-based approach, we have examined two main studies that have examined the 15 relationships completely. The first one uses the Pearson correlation and the second uses the Choquet Integral approach. The two studies use an intermediator factor in analyzing the relationships. The first one has used the overall impact of architectural styles on achieving quality attributes, and the second one has used the impact of architectural tactics on achieving quality attributes.

The two research works firstly adopts the technique of expert judgment in evaluating the impact, then use the mathematical approach in defining the type and the degree of the relationship of quality attributes. From these two surveyed research works the results show that there are considerable differences in the type of relationships as well as the degree of this relationship between quality attributes. Also, it is obvious that there is no considerable matching between the relationships' results of these both studies and the results of the experience-based technique as we have found previously.

Table 2: The relationships between quality attributes from seven sources based on experience technique

International Journal of Engineering and Information Systems (IJEAIS)
ISSN: 2643-640X
Vol. 5 Issue 3, March - 2021, Pages: 29-34

			Interaction types as defined in literature						
ID	Quality attribute	vs. Quality attribute	McCall [4]	Wiegers [17]	Zulzalil [12]	Henningsson [11]	Boehm [6]	Deutsch [18]	Berander [8]
1	**Functionality**	**Efficiency**	**neg**	**neg**	**neg**	**neg**	**neg**		
2	**Functionality**	**Reliability**	**pos**	**pos**	**pos**				
3	**Functionality**	**Usability**	**pos**		**pos**		**pos**		
4	**Functionality**	**Maintainability**	**pos**		**pos**			**pos**	
5	**Functionality**	**Portability**	**pos**	**pos**			**pos**		**pos**
6	*Efficiency*	*Reliability*		*neg*					
7	**Efficiency**	**Usability**	**neg**	**neg**	**neg**	**neg**			
8	**Efficiency**	**Maintainability**	**neg**	**neg**	**neg**				
9	**Efficiency**	**Portability**	**neg**	**neg**	**neg**		**neg**		
10	**Reliability**	**Usability**	**pos**	**pos**	**pos**	**pos**		**pos**	
11	Reliability	Maintainability	pos	pos	pos	neg		neg	
12	*Reliability*	*Portability*							*neg*
13	*Usability*	*Maintainability*	*pos*						
14	*Usability*	*Portability*		*neg*					
15	Maintainability	Portability	pos	neg	pos	neg			pos

Table 3: Relationships between quality attributes from two sources based on mathematical approach

			Interaction types as defined in literature	
ID	Quality attribute	vs. Quality attribute	Svahnberg [10]	Alashqar [15]
1	Functionality	Efficiency	-0.124	0.20
2	Functionality	Reliability	-0.566	0.06
3	Functionality	Usability	-0.404	0.33
4	Functionality	Maintainability	-0.653	0.01
5	Functionality	Portability	-0.575	0.05
6	Efficiency	Reliability	-0.421	0.15
7	Efficiency	Usability	-0.812	0.15
8	Efficiency	Maintainability	-0.462	0.04
9	Efficiency	Portability	-0.251	0.02
10	Reliability	Usability	0.721	0.09
11	Reliability	Maintainability	0.495	-0.10
12	Reliability	Portability	0.805	-0.10
13	Usability	Maintainability	0.588	0.04
14	Usability	Portability	0.404	0.02
15	Maintainability	Portability	0.719	0.01

5. Conclusion and Future Outlook

Good understanding and defining the relationships between quality attributes will increase the level of quality for the software being developed. Furthermore, it will reserve an appropriate balance of the required level of quality preferences that meet s oftware customer expectations. In this paper we have introduced a literature survey for exploring and defining the relationships between quality attributes that defined by the ISO9126 quality model. For achieving this, we proposed a mapping technique between equivalent attributes that expressed differently by researchers.

However, a lot of research work is still needed for approving whether a relationship between two quality attributes is stable or changing based on specific impact factors. The authors in [16] argue that the field of studying quality tradeoff is still not mature and there are no obvious approved approaches used in researching quality tradeoff. Although most of research studies that have examined the relationships between quality attributes from one or more of the following perspectives, literature, academia and industry there still other perspectives that have not been explored. Such as whether the relationships affected by a specific domain, type of software and stage of development. While most of studies focus on the experience-based approach in defining the types of relationships using 3-point scale, a considerable work also needed for examining these relationships with other approaches especially the mathematical-based because it helps in quantifying these relationships. The mathematical-based technique can also be used in approving the types of relationships of quality attributes that resulted from research works of the experience-based approach. The direct impact on quality attributes such as design decisions is also needed to be considered.

6. References

[1] ANSI/ASQC A3. (1978). Quality systems terminology. American National Standards Institute.

[2] ISO/IEC 9126-1. (2001). Software engineering —Product quality — Part 1: Quality model.

[3] Kazman, R. and Bass, L. (1994). Toward Deriving Software Architectures From Quality Attributes. Technical Report (CMU/SEI-94-TR-10), Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania.

[4] McCall, J. A. (1994).Quality Factors, Encyclopaedia of Software Engineering, John Wiley and Sons, 958–969.

[5] Alashqar, A. M., El-Bakry, H. M., & Elfetouh, A. A. (2017). A framework for selecting architectural tactics using fuzzy measures. International Journal of Software Engineering and Knowledge Engineering, 27(03), 475-498.

[6] Boehm, B., & In, H. (1996). Identifying quality-requirement conflicts. IEEE software, 13(2), 25-36.

[7] Gade, D. (2013). The evaluation of software quality. University of Nebraska-Lincoln, Industrial and Management Systems Engineering, Dissertations and Student Research.

[8] Berander, P., Damm, L. O., Eriksson, J., Gorschek, T., Henningsson, K., Jönsson, P., and Tomaszewski, P. (2005). Software quality attributes and trade-offs. Blekinge Institute of Technology.

[9] Henningsson K (2001). Trade-offs and conflicts between quality attributes. Master's thesis, Department of Software Engineering and Computer Science, Box 520 SE – 372 25 Ronneby Blekinge Institute of Technology.

[10] Svahnberg, M., and Wohlin, C. (2005). An investigation of a method for identifying a software architecture candidate with respect to quality attributes. Empirical Software Engineering, 10(2), 149-181.

[11] Henningsson, K., and Wohlin, C. (2002). Understanding the relations between software quality attributes -a survey approach. In Proceedings 12th International Conference for Software Quality.

[12] Zulzalil, H., Ghani, A. A., Selamat, M. H., and Mahmod, R. (2008). A case study to identify quality attributes relationships for web-based applications. International Journal of Computer Science and Network Solutions 8(11), 215-220.

[13] Al-Badareen, A. B., Selamat, M. H., Jabar, M. A., Din, J., and Turaev, S. (2011). Software Quality Models: A Comparative Study. In Software Engineering and Computer Systems, Springer-Verlag Berlin Heidelberg, 46-55.

[14] Chung,L., Nixon, B. A. , Yu, E., and Mylopoulos, J. (2000). Non-functional requirements in software engineering. Kluwer Academic Publishers, Boston, Hardbound.

[15] Alashqar, A. M., Elfetouh, A. A., and El-Bakry, H. M. (2016). Analyzing Preferences and Interactions of Software Quality Attributes Using Choquet Integral Approach. INFOS '16 Proceedings of the 10th International Conference on Informatics and Systems (May, Cairo University, Egypt), ACM, 298-303.

[16] Barney, S., Petersen, K., Svahnberg, M., Aurum, A., and Barney, H. (2012). Software quality trade-offs: A systematic map. Information and Software Technology, 54(7), 651-662.

[17] Wiegers K. E. (2003). Software Requirements, 2nd Edition, Microsoft Press.

[18] Deutsch, M. S., and Willis, R. R. (1988). Software quality engineering: a total technical and management approach. Englewood Cliffs, NJ, Prentice Hall.