DOI: https://doi.org/10.53052/9788367652124.06

Roman HAMRETSKYI[1]

Opiekun naukowy: Viktor GNATYUK[2]

WIELOKRYTERIALNA OCENA JAKOŚCI OPROGRAMOWANIA W SYSTEMACH INFORMATYCZNYCH ORAZ KOMUNIKACYJNYCH ZGODNIE Z MODELEM SQuaRE

Streszczenie: W artykule omówiono ważność oceny jakości oprogramowania w Systemach Informatycznych oraz Komunikacyjnych (ICS) stosując tzw. model SQuaRE. Omówiono uwzględnianie wybranych kryteriów oraz działania/kroki związane z wielokryterialną oceną jakości ICS.

Słowa kluczowe: wielokryterialny, jakość, oprogramowanie, Model SQuaRE

MULTI-CRITERIA ASSESSMENT OF SOFTWARE QUALITY IN INFORMATION AND COMMUNICATION SYSTEMS ACCORDING TO THE SQuaRE MODEL

Summary: The article describes the significance of evaluating software quality in Information and Communication Systems (ICS) using the SQuaRE model. It outlines the criteria and steps involved in multi-criteria software quality assessment in ICS.

Keywords: multi-criteria, quality, software, SQuaRE Model

1. Introduction

With the development of technologies and the increase in the complexity of software, there is a need to define effective methods and criteria for evaluating the quality of programs. Evaluation criteria can help determine the level of functionality, reliability, performance, and security of software. This is especially important in information and communication systems (ICS), where aspects of communication, data exchange, and system availability are inextricably linked.

Multi-criteria assessment of software quality in ICS is an important process for determining the efficiency, reliability and other functional dimensions of software. It

[1] National Aviation University, Faculty of Aeronautics, Electronics and Telecommunications, Telecommunications and radio engineering, 45391@stud.nau.edu.ua

[2] National Aviation University, Faculty of Aeronautics, Electronics and Telecommunications, Telecommunications and radio engineering, viktor.hnatiuk@npp.nau.edu.ua

allows you to take into account many different factors and criteria that affect the quality of a software product. The main purpose of multi-criteria assessment is to compare and understand the level of achievement of each criterion (metric) and their relationship.

2. Literature review

Software quality assessment is critically important to ensure the effectiveness and reliability of ICS. Multi-criteria software quality assessment in this context considers the importance of considering different aspects of quality and developing multi-criteria models for better management and improvement of software quality.
The topic of using multi-criteria software quality assessment is relevant and has been the subject of research by various scientists.
Author Jagat Sesh Challa and others in their work "Integrated Software Quality Evaluation: A Fuzzy Multi-Criteria Approach" consider the issue of measuring and evaluating software quality based on a fuzzy multi-criteria approach. The authors take into account the ambiguous nature of software quality attributes, where quality may not be clearly defined. As a basis, they use the characteristics proposed by the ISO/IEC 9126 model [1].
In the work "Preferential Selection of Software Quality Models Based on a MultiCriteria Decision-Making Approach", the authors consider an important aspect in the field of software engineering, namely the selection of an appropriate software quality model, taking into account the needs of the client and his priorities. The main goal of the study is to determine the best software quality model that meets the client's requirements and allows giving preference to certain criteria over others in accordance with specific application requirements. The proposed approach helps to choose the best alternative that is suitable for the client's application.
An important part of the methodology is the use of Multi-Criteria Decision Making (MCDM) methods, such as the Analytic Hierarchy Process, TOPSIS, and PROMETHEE II, as well as the application of fuzzy logic to solve the problem of choosing the best software quality model [2].
It is also worth paying attention to the work "Conceptual Framework For Potential Implementations Of Multi Criteria Decision Making (MCDM) Methods For Design Quality Assessment", which aims to develop a conceptual framework for potential applications of multi-criteria decision-making methods in the assessment of design quality, in particular architectural design.
The authors of the work set themselves the task of finding or adapting appropriate MCDM techniques that are already used in other fields for design quality assessment, adapting them to the construction industry, and verifying their applicability.
The paper compares and analyzes existing design quality assessment tools (such as DQI, DEEP, AEDET, HQI, LEED, BREEAM, BQA) with different MCDM methods (such as AHP, ANP, PROMETHEE, SAW, and TOPSIS). The authors explore the advantages and disadvantages of the results of these comparisons and their applicability in the field of architectural design [3].
The book "Multi-Criteria Decision Models in Software Reliability: Methods and Applications" explores the significance of software reliability in the digital era. It investigates various factors and criteria influencing software reliability, offering

decision-making methods to enhance the potency and efficiency of decision-making processes. The book also focuses on the application of multi-criteria decision models in the medical field to improve the accuracy and reliability of programs for diagnosis and monitoring, where reliability is of critical importance [4]. The book demonstrates modern practices of multi-criteria analysis that can be used in various ICS.
Multi-criteria assessment of software quality in ICS is a relevant and important field of research. Related publications consider different approaches and methods that help to consider different aspects of quality and make informed decisions to improve software quality. Research in this field is evolving and providing practical guidance for software engineers and researchers.

3. SQuaRE model

The criteria for assessing the quality of software in an ICS should be based on the needs of the environment in which this ICS functions and what tasks it performs. The generally accepted SQuaRE model can be used to determine the main criteria for the assessment.
The SQuaRE model (Software Quality Requirements and Evaluation) is one of the models for software quality assessment. This model was developed and presented at the "Software Quality Week" conference in 1996 and is described in the ISO/IEC 25010:2011 standard.
The SQuaRE model is based on the definition of eight main criteria that affect software quality. Each of these criteria characterizes different aspects of quality and can be evaluated using certain metrics or indicators. The main criteria of the SQuaRE model include [4]:

- Functional Suitability: evaluation of the compliance of the software with the functional requirements of the IC, such as the ability to process data, communication between system components, and interaction with other systems and applications;
- Performance Efficiency: this factor reflects software performance, namely system speed, response to user requests, optimal use of resources, and scalability;
- Compatibility: evaluates how a system or component can exchange information with other products, systems or components and/or perform its functions;
- Usability: the user's ease of use of the software, its intuitiveness, documentation, and support are evaluated;
- Reliability: the stability and reliability of the software, its ability to work without failures, and recovery in the event of errors or failures is evaluated;
- Security: is evaluated as a product or system protects information and data so that individuals or other products or systems have a degree of access to data that corresponds to their types and levels of authorization;
- Favorability for development (Maintainability): evaluates how easy it is to maintain, and modify the software, including its comprehensibility, modularity and documentation;
- Portability: the software's ability to work on different platforms, operating systems, and environments, including mobile devices, cloud services, etc., is evaluated;

The given criteria also have sub-criteria that can be used for more detailed quality assessment.
According to the SQuaRE model, each of these factors can be rated on a scale from 0 to 100% depending on the needs and requirements of the organization. It provides a means to systematically evaluate the quality of software and helps identify its strengths and weaknesses for further improvement.

4. Multi-criteria evaluations

The criteria for assessing the quality of software in an ICS should be based on the needs of the environment in which this ICS functions and what tasks it performs. The generally accepted SQuaRE model can be used to determine the main criteria for the assessment.
The SQuaRE model (Software Quality Requirements and Evaluation) is one of the models for software quality assessment. This model was developed and presented at the "Software Quality Week" conference in 1996. After the introduction of the model, work on standardization took place. Attempts to standardize this approach were determined by the work of committees and experts within the framework of ISO/IEC. The result of this work was the ISO/IEC 25010:2011 standard, which formally describes the SQuaRE model and includes quality attributes, processes, and assessment models. The standard was revised in 2017 and it remains current (ISO/IEC 25010:2011). The replacement is being prepared in the form of ISO/IEC DIS 25002, ISO/IEC 25010, and ISO/IEC 25019 standards, which are under development.
The SQuaRE model is based on the definition of eight main criteria that affect software quality. Each of these criteria characterizes different aspects of quality and can be evaluated using certain metrics or indicators. The main criteria of the SQuaRE model include [5]:

- Functional Suitability: evaluation of the compliance of the software with the functional requirements of the IC, such as the ability to process data, communication between system components, and interaction with other systems and applications;
- Performance Efficiency: this factor reflects software performance, namely system speed, response to user requests, optimal use of resources, and scalability;
- Compatibility: evaluates how a system or component can exchange information with other products, systems or components and/or perform its functions;
- Usability: the user's ease of use of the software, its intuitiveness, documentation, and support are evaluated;
- Reliability: the stability and reliability of the software, its ability to work without failures, and recovery in the event of errors or failures is evaluated;
- Security: is evaluated as a product or system protects information and data so that individuals or other products or systems have a degree of access to data that corresponds to their types and levels of authorization;
- Favorability for development (Maintainability): evaluates how easy it is to maintain, and modify the software, including its comprehensibility, modularity and documentation;

- Portability: the software's ability to work on different platforms, operating systems, and environments, including mobile devices, cloud services, etc., is evaluated;

The given criteria also have sub-criteria that can be used for more detailed quality assessment.

- According to the SQuaRE model, each of these factors can be rated on a scale from 0 to 100% depending on the needs and requirements of the organization. It provides a means to systematically evaluate the quality of software and helps identify its strengths and weaknesses for further improvement.

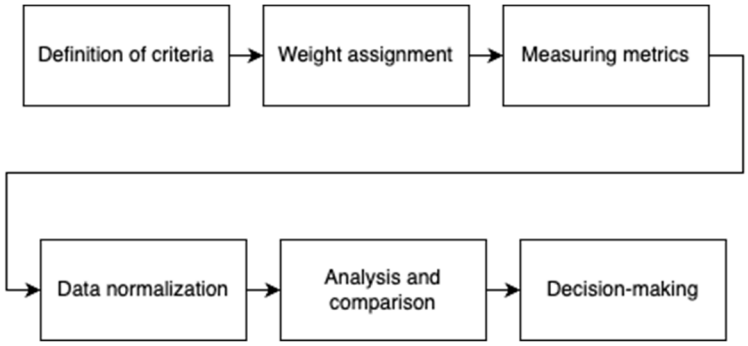

Figure 1. The main steps of the multi-criteria evaluations

A multi-criteria quality assessment based on the SQuaRE model can be carried out for different software requirements in ICS, which will reflect the importance of each criterion depending on the context and needs of the organization. The set of requirements for software that will be used for multi-criteria evaluation can be determined on the basis of additional research or data possessed by specialists conducting the analysis. Here are some possible software requirements that could be used:

Importance to the user: the extent to which each criterion of the SQuaRE model affects users and the satisfaction of their needs in ICS is assessed. For example, functionality and usability can be of great importance to users.

Business criticality: the impact of each criterion on the business processes and goals of the organization in the context of ICS is assessed. For example, reliability and performance can be business-critical, especially in real-time or transactional systems.

Technical complexity: the level of complexity of implementing each criterion is assessed from a technical point of view. Some criteria may require additional resources, effort, or sophisticated technology to achieve.

Implementation cost: the cost associated with the implementation and provision of each criterion is estimated. This may include development, implementation, testing, and support costs.

Integration risk: the risk associated with the integration of each criterion into the existing ICS is assessed. Some criteria may require changes in system architecture or have potential interactions with other system components.

The methodology of expert evaluation can be used to evaluate the criteria for compliance with the requirements. Using the example of two experts, let's consider

how their expert assessment might look. The importance of each characteristic can be established based on stakeholder consultation, consideration of business priorities, and technical complexity. These weights can be changed according to the needs and requirements of a particular organization. The assessment can be made more specific with the help of additional criteria or other parameters that may be actual to the software or ICS where it will be used.
Here is an example of a multi-criteria assessment of requirements according to the SQuaRE model for software in ICS. Each criterion can receive points from 0 to 20. Where 0 means that the criterion is not important for meeting the requirements, and 20 is very important.

Table 1. Example of assessment by expert A

Criteria	Requirements				
	Importance to the user	Business criticality	Technical complexity	Implementation cost	Integration risk
Functional Suitability	16	20	12	16	12
Reliability	20	20	16	12	16
Performance Efficiency	12	16	20	20	12
Compatibility	8	12	8	12	8
Portability	16	20	16	8	16
Maintainability	12	16	12	16	16
Security	20	20	20	20	20
Usability	12	16	12	12	8

Another expert also conducts evaluation based on the same criteria and requirements.

Table 2. Example of assessment by expert B

Criteria	Requirements				
	Importance to the user	Business criticality	Technical complexity	Implementation cost	Integration risk
Functional Suitability	20	20	12	16	12
Reliability	20	20	16	12	16
Performance Efficiency	16	16	20	20	12
Compatibility	12	12	8	12	8
Portability	20	20	16	8	16
Maintainability	16	16	12	16	16
Security	20	20	20	20	20
Usability	16	16	12	12	8

With the evaluations of all experts, it is possible to determine which criteria are most important for compliance. To do this, it is necessary to determine the overall assessment of the criteria according to each expert, and then find the mean.

$$A = \sum_{i=1}^{n} \quad C_i \tag{1}$$

$$B = \sum_{i=1}^{n} \quad C_i \tag{2}$$

$$MV = \frac{A + B}{k} \tag{3}$$

where:

- A - a total assessment of expert A;
- B - a total assessment of expert B;
- C - a criteria assessment (0-20);
- n - number of requirements;
- k - number of experts;
- MV - mean value.

Table 3. Determination of the most important criteria

Criteria	The total assessment of expert A	The total assessment of expert B	Mean value
Security	100	100	100
Reliability	84	84	84
Performance Efficiency	80	84	82
Functional Suitability	76	80	78
Portability	76	80	78
Maintainability	72	76	74
Usability	60	64	62
Compatibility	48	52	50

Based on the data from the example, we see that the criteria of safety, reliability, and efficiency are the most important to achieve compliance with the requirements important to the user, business critically, etc.

The obtained data can be used to rank the criteria according to their importance for the realization of the software characteristics. The higher the value, the more important the criterion for ensuring software quality characteristics.

This approach to determining the priority of criteria can allow developers, managers, etc. more carefully plan software development processes. The results of the assessment can be used to set priorities in software development and to make decisions about quality improvement in ICS. In this way, the necessary parameters of the customer's requirements can be achieved in the first place.

It is worth noting that the multi-criteria evaluation process itself may vary depending on the context and requirements of the organization. The quality of such an analysis depends on the level of experts who were involved in the assessment, their number, and the number of requirements with an explanation of the approach to assigning an assessment.

Expert assessment with multi-criteria analysis offers a streamlined approach for evaluating software quality, leveraging the expertise of professionals. This method provides flexibility in adapting to project-specific needs and yields detailed insights into various quality attributes. Involving experienced experts enhances the credibility of the assessment, contributing valuable perspectives for improvement.
However, expert assessments may introduce subjectivity, influenced by individual opinions and experiences. Achieving objectivity becomes a challenge, impacting result accuracy. Assigning appropriate weights to criteria poses difficulties, and outcomes heavily rely on the qualifications and objectivity of the participating experts. Analyzing a significant volume of data in multi-criteria evaluations can be intricate, adding complexity to the process. Consideration of these factors is crucial for a comprehensive and reliable software quality assessment.

5. Conclusions

The article provides an example of using the SQuaRE model for multi-criteria analysis using the expert evaluation method. The given example demonstrates the process, describing the advantages and disadvantages.
Multi-criteria assessment of software quality introduces a systematic approach to the analysis of achievements in the context of various aspects and criteria. This method allows you to rationally make decisions about the improvement and improvement of the software product, especially in ICS.

REFERENCES

1. CHALLA J. S., PAUL A., DADA Y., NERELLA V., SRIVASTAVA P. R., SINGH, A. P. (2011). Integrated Software Quality Evaluation: A Fuzzy Multi-Criteria Approach. Journal of Information Processing Systems, 7(3), 473–518. https://doi.org/10.3745/JIPS.2011.7.3.473
2. VERMA A., AGARWAL A., RATHORE M., BISHT S., SINGH D. (2023). Preferential Selection of Software Quality Models Based on a Multi-Criteria Decision-Making Approach. International Journal of Software Innovation (IJSI), 11(1), 1-13. http://doi.org/10.4018/IJSI.315739
3. HARPUTLUGIL TIMUÇIN, PRINS MATTHIJS, GÜLTEKIN A., TOPCU ILKER. (2011). Conceptual framework for potential implementations of multi criteria decision making (MCDM) methods for design quality assessment.
4. MISHRA A., LINH N. T. D., BHARDWAJ M., PINTO C. M. A. (2022). Multi-Criteria decision models in software reliability: Methods and Applications. Information Technology, Management and Operations Research Practices.
5. ISO/IEC 25010:2011. Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - System and software quality models, 1st ed. ISO/IEC, 2011.