Towards a Modern Quality Framework

Martin Glinz
University of Zurich
Dept. of Informatics
Zurich, Switzerland
glinz@ifi.uzh.ch

Norbert Seyff
University of Applied Sciences and Arts Northwestern Switzerland
Windisch, Switzerland
norbert.seyff@fhnw.ch

Stan Bühne
IREB GmbH
Karlsruhe, Germany
stan.buehne@ireb.org

Xavier Franch
Universitat Politècnica de Catalunya
Barcelona, Spain
franch@essi.upc.edu

Kim Lauenroth
University of Applied Sciences and Arts Dortmund
Dortmund, Germany
kim.lauenroth@fh-dortmund.de

2023 IEEE 31st International Requirements Engineering Conference Workshops (REW) | 979-8-3503-2691-8/23/$31.00 ©2023 IEEE | DOI: 10.1109/REW57809.2023.00067

***Abstract*—Quality frameworks have been used in requirements engineering (RE) for a long time to help elicit and document quality requirements. However, existing quality frameworks have major issues that hamper their applicability, particularly in RE, but also in other fields such as the design of digital systems.**

In this paper, we discuss the issues of existing quality frameworks and propose a new quality model, which has been designed for application as a quality framework in RE as well as in the design of digital systems. We present the rationale and requirements for our new model, introduce the model and sketch its application. Our work contributes to the improvement of quality frameworks used in RE and Digital Design.

***Index Terms*—Quality framework, Quality model, Requirements Engineering, Digital Design**

I. Introduction

Dealing with quality in digital, software-based systems is notoriously difficult. According to [1], we define quality as *"The degree to which a system satisfies stated and implied needs of its stakeholders"*. Quality models have been developed as frameworks for classifying, assessing and measuring systems and software quality [2], [3], [4]. In Requirements Engineering (RE), we have to deal with *quality requirements, i.e., requirements that pertain to a quality concern that is not covered by functional requirements* [1]. Eliciting, documenting and validating quality requirements is a major challenge [5].

Elicitation of quality requirements can be supported by using a framework of typically required qualities as a guideline or checklist for both requirements engineers and stakeholders. Popular frameworks are the overall quality model given in the ISO/IEC standard 25010 [4] and the Volere requirements specification template [6].

In this position paper, we argue that the existing quality frameworks face several issues and need an update to reflect the quality needs of today's digital systems. We will particularly shed light on the weaknesses of ISO/IEC 25010 and Volere and present our vision of a modern quality framework. Such a framework will be useful not just in RE. We envision a broad application of a modern quality framework for designing and realizing digital systems that meet the quality expectations of their stakeholders.

II. Quality Requirements and Frameworks

According to [1], we classify requirements into functional requirements, quality requirements and constraints. The latter two are also called non-functional requirements. Differentiating between functional and quality requirements as well as ways how to elicit, structure and represent quality requirements have been a widely discussed topic in the requirements engineering community for many years [5], [7], [8], [9].

Quality models as well as requirements specification templates provide a structured view on qualities that are frequently required when specifying or designing systems and software. Such models or templates can serve as checklists for elicitation and as aids for documenting quality requirements [10]. In the subsequent two subsections we briefly present two popular representatives: the ISO/IEC 25010 quality model and the Volere requirements specification template.

A. The ISO/IEC 25010 Quality Model

Software quality models have a long history, starting with the models of Boehm et al. [2] and McCall and Matsumoto [3]. The first international standard on software quality models was ISO/IEC 9126, originally published in 1991 [11] and revised in 2001 [12]. In the course of a major restructuring of all ISO quality standards, ISO/IEC 9126 was superseded in 2011 by the new standard ISO/IEC 25010 [4].

Both ISO/IEC 9126 and ISO/IEC 25010 distinguish between product quality and quality-in-use and present two corresponding quality models. The ISO/IEC 25010 product quality model includes: *functional suitability*, *performance*, *efficiency*, *compatibility*, *usability*, *reliability*, *security*, *maintainability* and *portability*. The characteristics included in the quality-in-use model are: *effectiveness*, *efficiency*, *satisfaction*, *freedom from risk* and *context coverage*. For each of these characteristics, several subcharacteristics are defined.

The ISO/IEC 25010:2011 standard is currently under revision. The plan is to separate the two quality models into two separate standards, the revised ISO/IEC 25010 [13] with the product quality model and the new standard ISO/IEC 25019 [14] with the quality-in-use model. The revised product quality model will include *safety* as a main characteristic, which is missing from the current model. Also, some subcharacteristics will be added or renamed. The new ISO/IEC

DOI 10.1109/REW57809.2023.00067

25019 standard will feature a totally revised quality-in-use model, departing from a user's perspective toward covering the stakeholders' needs. The revised model only keeps *freedom from risk*. The other characteristics of the 2011 model (see above) are replaced by two new characteristics: *beneficialness* and *acceptability*.

B. The Volere Template

We have selected the Volere Requirements Specification Template [6] for our discussion as it contains a wide range of possible quality requirements organized into main categories and subcategories. Main categories in Volere are: *look and feel requirements*, *usability and humanity requirements*, *performance requirements*, *operational and environmental requirements*, *maintainability and support requirements*, *security requirements*, *cultural requirements* and *compliance requirements*.

Volere is designed not only as a specification template, but also as a guideline for specifying requirements. For each of the categories mentioned above, Volere provides subcategories, together with guiding advice and examples.

III. Issues of Existing Frameworks

Unfortunately, both the ISO/IEC 25010 standard and the Volere template have major issues that hamper and limit their use as a quality framework in RE – and we are not aware of any other frameworks that would be better suited. We briefly discuss these issues the in the subsections below.

A. Issues of ISO/IEC 25010

The separation of product quality and quality-in-use into two distinct quality models is one of the major issues of ISO/IEC 25010. It goes back to the 2001 revision of ISO/IEC 9126, where a separate quality-in-use model was introduced in order to foster the users' view of the quality of a system or product [15]. Unfortunately, this separation does not work as intended: (1) Many of the qualities in the product quality model cannot be assessed based on the product alone, but only when using the product in some context. For example, *reliability* belongs to the product quality model of ISO/IEC 25010, but it is defined as the "degree to which a system, product or component performs specified functions under specified conditions for a specified period of time", which can only be assessed when the product is in use. (2) The quality-in-use model has been changed radically both in the transition from ISO/IEC 9126 to ISO/IEC 25010 in 2011 and in the currently ongoing transition from ISO/IEC 25010 to ISO/IEC DIS 25019. These changes indirectly prove that the quality-in-use models of ISO/IEC 9126 and ISO/IEC 25010 did not really work. (3) When looking at the new quality-in-use model with the characteristics *beneficialness*, *freedom from risk* and *acceptability* in ISO/IEC DIS 25019, we are afraid that this model does not constitute an improvement. For example, *usability* as a subcharacteristic of *beneficialness* in ISO/IEC DIS 25019 is defined differently from *usability* as a characteristic of product quality in ISO/IEC DIS 25010. This will be confusing for any users of these standards. As another example, the characteristic *freedom from risk* lumps together rather different things such as freedom from life or health risks (a sub-topic of safety), from environment or societal risks (a sub-topic of sustainability), and from economical risks. The latter again lumps together rather different things such as risks "to financial status, efficient operation, commercial property, reputation, or other resources in the intended contexts of use" [14]. A characteristic with such a low cohesion will be rather difficult to use in practice.

Furthermore, important qualities such as *safety*, *privacy* and *sustainability* are missing from the ISO/IEC 25010:2011 product quality model. *Safety* will be added in the ongoing revision, but *privacy* and *sustainability* are still missing. Also, *explainability* is missing in both ISO/IEC 25010:2011 and ISO/IEC DIS 25010, despite its importance for AI-based systems. The revision cycle time of ISO/IEC standards is about ten years, so the missing quality characteristics will not appear in ISO/IEC 25010 in the near future.

B. Issues of Volere

Volere discusses a broad range of quality attributes. However, similarly to ISO/IEC 25010, the Volere template misses important qualities such as *safety*, *sustainability* or *explainability*. In addition to the problem of incomplete coverage, there is further the problem of inadequate classifications. For example, treating privacy as a subcharacteristic of security is inadequate in our view.

The Volere template generally treats quality attributes as a single section and provides only limited granularity and detail. While it makes sense to specify globally crosscutting quality requirements in a top level chapter, this is inadequate for quality requirements that are confined to some subsystem or component. Hence, we argue that the strict separation of functional and quality requirements on the top level of the specification is inadequate, particularly when specifying a system that has subsystems or components. When following the principle of separation of concern, the primary structure of the detailed requirements section of a requirements specification should be by component and not by requirement type.

Finally, Volere is not applicable beyond the specification and documentation of requirements.

IV. General Considerations for New Quality Frameworks

The identified limitations and our ideas to overcome them have led to a discussion among the authors about high-level considerations for future quality models. In this section, we briefly present three key insights.

Overcoming existing limitations: Although overcoming existing limitations of current frameworks is a key goal for novel quality frameworks, we recognize that it can be challenging to completely overcome all existing limitations. Therefore, the expectation is that novel quality frameworks will have a specific scope and target audience. This would allow for focused efforts on addressing specific needs and challenges.

Including what matters: Ideally, a quality model would include exactly those characteristics that actually matter in practice. However, to our knowledge, there are no studies investigating which quality characteristics are used in practice to which extent. So we decided to base our new quality model on (1) the popular existing frameworks, and (2) our insights about the problems of the existing frameworks. As a sanity check, we looked at the quality characteristics identified in two systematic literature reviews [8], [16].

Open for extension: As software development evolves, new quality characteristics emerge, and frameworks need to be able to accommodate these characteristics. In our opinion, future frameworks should be easier to evolve than today's standards are, by following the principle of "Open for extension, closed for modification". This includes adding subcharacteristics that make the characteristics of the model more precise and allow measurement where possible.

V. A New Quality Framework

Based on the analysis of the problems with existing quality frameworks and following the considerations discussed above, we sketch a new quality model that, in our opinion, better reflects the needs of contemporary requirements engineering and design of digital solutions.

We first present our requirements and rationale. In Subsection V-B, we introduce our new model. The subsequent two subsections illustrate the envisaged use of the new quality model as a quality framework in RE and in Digital Design.

A. Requirements and Rationale

Our rationale for coming up with a new model is three-fold: (1) As shown above, the existing frameworks have deficiencies which hamper their application in RE. (2) For requirements engineers and stakeholders, quality is a means to an end. Hence, they need a conceptually simple quality framework that is easy to learn and apply. (3) As the authors are involved in the creation of IREB's Digital Design Professional certification program (https://www.digitaldesign.org), we want a framework that is also useful as a guideline for dealing with quality in Digital Design.

So we state the following requirements: The new quality model shall be *conceptually simple* (R1), *adequate* (R2), and *easy to apply* (R3). Furthermore, as we are in the early stage of model development, we also want to *keep the model on the level of characteristics* (R4).

To make these requirements more concrete and tangible, we define the following acceptance criteria for them:

- R1-AC1: The model can be fully described on less than ten pages.
- R1-AC2: Its essence is captured in no more than two tables.
- R2-AC1: Consistent terminology, based on the existing terminology in RE [1].
- R2-AC2: Inclusion of those characteristics that are relevant in today's RE and Digital Design.
- R3-AC1: Applicable as-is as a guideline for eliciting requirements, in particular quality requirements and constraints.
- R3-AC2: Applicable as-is as a framework for designing the quality of digital solutions or digital systems in Digital Design.
- R4-AC1: At most two levels of characteristics in the model.
- R4-AC2: No details about measuring the characteristics.

B. The Proposed New Quality Model

Table I shows an overview of the basic characteristics and their subcharacteristics. While there are obvious similarities to the product quality model of ISO/IEC 25010:2011 [4], there are also major differences. *Sustainability*, *safety* and *privacy* do not occur in the ISO/IEC 25010 product quality model. We have added these three characteristics with our acceptance criterion R2-AC2 (see Sect. V-A) in mind: these characteristics do occur when eliciting requirements or designing digital solutions.

TABLE I
A new quality model: Basic properties

Characteristic	Subcharacteristics
Functionality	Structure and data, Function and flow, State and behavior, Context and boundary
Performance	Time, Volume, Frequency, Throughput, Resource consumption
Usability	Learnability, Ease of use, User assistance, Explainability
Sustainability	Environmental, Social
Reliability	Availability, Fault tolerance, Recoverability
Security	Confidentiality, Integrity, Non-repudiation, Accountability, Authenticity, Resistance
Safety	Preventability, Resilience
Privacy	Data sovereignty, Data minimization, Anonymity, Non-disclosure
Maintainability	Modifiability, Reusability
Portability	Adaptability, Installability, Scalability
Compatibility	Co-existence, Interoperability

In accordance with IREB's RE glossary, we consider the term *system* as an umbrella term that particularly includes *products*, *services* and *components* [1], p. 20. Our choice and naming of the characteristics has been guided by R2-AC2 and R2-AC1. We briefly discuss some of our decisions here.

Functionality and *performance* and their subcharacteristics have been changed or renamed to reflect their usage in RE. The terminology follows the one used by IREB [1], [10]. For *usability*, we have reduced the subcharacteristics from ISO/IEC 25010 to the three core ones: *learnability*, *operability* and *accessibility*. We have renamed the latter two to *ease of use* and *user assistance*, which characterize these properties better in our view[1]. We have added *explainability*, which is particularly important for any system that contains an AI component. When defining the subcharacteristics of

[1]ISO/IEC DIS 25010 will also rename accessibility to user assistance.

TABLE II
A NEW QUALITY MODEL: EMERGENT PROPERTIES

Characteristic	Mainly emerges from
Compliance	Functionality, Performance, Sustainability, Reliability, Security, Safety, Privacy
Dependability	Reliability, Security, Safety, Privacy
Efficiency	Functionality, Performance, Usability
User experience	Functionality, Performance, Usability, Dependability

sustainability, we were inspired by the five sustainability dimensions in the Karlskrona Manifesto [17]. We are still discussing different ways to include these dimensions in the model. For this first version, we have decided that we need to explicitly include environmental and social sustainability. We argue that technical and also individual sustainability, at least to some extent, are covered by other quality characteristics within the model. Neither do we see a strong need to include the economic dimension. For *safety*, we have chosen two subcharacteristics, which are particularly relevant with R2-AC2 in mind. *Preventability* is the ability to prevent or mitigate potential hazards associated with the use of the system or being caused by failures. *Resilience* is the ability of the system to recover from a failure or adverse event and return to a safe state. For *privacy*, we have chosen *data sovereignty* as a subcharacteristic. This is the ability of a person or organization to control and autonomously use the data produced by them or collected about them. The meaning of the other subcharacteristics should be obvious from their names. In comparison to ISO/IEC 25010, we have streamlined *maintainability* and *portability*[2].

As we have illustrated in Sect. III, the distinction between product quality and quality-in-use in ISO/IEC 25010:2011 is flawed, so we do not make this distinction. Actually, *functionality*, *performance*, *usability* and *sustainability* pertain to both the system and its efficient use. *Reliability*, *security*, *safety*, and *privacy* primarily pertain to the system in use, while *maintainability*, *portability* and *compatibility* pertain to the system as such.

We model *compliance*, *dependability*, *efficiency* and *user experience* separately as emergent properties, see Table II. *Compliance* emerges when the system's *functionality*, *performance*, *sustainability*, *reliability*, *security*, *safety* and *privacy* comply with legal, regulatory, ethical, etc. constraints. *Dependability* emerges from the trustworthiness of the stated *reliability*, *security*, *safety* and *privacy* properties. *Efficiency* emerges when the *functionality* of the system can be used efficiently under the stated *performance* and *usability* properties. Finally, *user experience* emerges when the combination of a system's *functionality*, *performance* and *usability* yield a joyful experience of working with the system, and the users can depend on the stated *reliability*, *security*, *safety* and *privacy*.

[2]ISO/IEC DIS 25010 renames portability to flexibility. We believe that flexibility is a too general term here, so we stick with portability.

C. Using the New Quality Model in RE

The new model, accompanied by a glossary with definitions of all terms used, can be used as is by requirements engineers as a *memory aid* when eliciting quality requirements from stakeholders. It can also serve as a checklist that requirements engineers can use for assessing the completeness of the quality requirements that have been elicited and documented so far.

However, for making the model truly useful as a framework and guideline for eliciting, documenting and validating requirements, the model would have to be extended with advice and examples, similar to what is available today in Volere. However, a new documentation template for quality requirements should allow a multi-level structure by components instead of the flat structure used in Volere (see Sect. III-B).

D. Using the New Quality Model in Digital Design

In Digital Design, form, function and quality are the three basic dimensions of design [18], [19]. We envision that our new model, together with some advice, will help digital designers to master the quality dimension of Digital Design. As Digital Design considers quality on the abstraction levels of solution, system and elements, a mapping of the quality model to these levels will be needed. This is beyond the scope of this paper and is subject to future work.

VI. DISCUSSION

A. Requirements and Acceptance Criteria Revisited

In this subsection, we revisit the acceptance criteria stated in Sect. V-A. We believe that our new model satisfies all acceptance criteria for our four requirements.

- The two tables of our model, together with textual definitions of all characteristics and subcharacteristics plus some application guidance will not require more than about six pages. So R1-AC1 is satisfied.
- The essence of the model is captured in Tables I and II, thus satisfying R1-AC2.
- Only a few terms currently are not included in the IREB RE Glossary [1]. So R2-AC1 is mostly satisfied.
- Based on the authors' experience in RE and Digital Design, we believe that we have included the relevant characteristics, so that R2-AC2 is satisfied.
- From our teaching experience, we do believe that the model is applicable as a guideline and framework both in RE and Digital Design. So we believe that our model satisfies R3-AC1 and R3-AC2.
- Our model has two levels and does not go into any detail about how to measure the stated characteristics and subcharacteristics. This satisfies R4-AC1 and R4-AC2.

B. Coverage of the New Model

On the top level, our model fully covers the quality characteristics of the product quality models of both ISO/IEC 25010:2011 and ISO/IEC DIS 25010:2022. With respect to the quality-in-use model of ISO/IEC DIS 25019:2022, we believe that *beneficialness* and *acceptability* are emergent properties that cannot be reasonably specified as a part of the quality

model. *Freedom from risk* is partially covered (see Sect. III-A). Whether or not economic risk should be part of a quality model is subject to future work. Currently, we believe that *freedom from economical risk* should not be part of our quality model.

C. Tailoring of the New Model

It makes sense to tailor a quality model to the context where it is used. According to the principle of "Open for extension, closed for modification", tailoring should be done by adding those subcharacteristics which are missing in the given context of use. Characteristics which are not needed should *not* be deleted, but marked as irrelevant in the given context.

D. Quality-in-use Revisited

We have argued that a separation into a product quality model and a quality-in-use model does not make sense (see Sect. III-A). That said, when *assessing* qualities (by measuring, reviewing, testing, etc.), it does make sense to distinguish between *static*, *at-runtime* and *in-use* assessment techniques. However, as there is an m:n relationship between quality characteristics and assessment techniques, this distinction cannot be carried over to the quality characteristics.

E. Limitations

There is no empirical validation of our new quality model, neither of the model as such, nor of its actual applicability as a guideline and framework for eliciting and documenting requirements and for designing digital systems and solutions. So far, this work is built solely on the expertise of the authors in RE, Digital Design and software quality. As this paper is a position paper, we believe that it is acceptable to present a not yet validated new concept for discussion in the community.

Satisfying requirement R4 is both an asset and a limitation of our model. It is an asset, because it makes the model simple and easy to use. On the other hand, it is a limitation, because for improving its value in practical application, the model needs to be evolved into a full quality framework, including advice about how to measure the quality characteristics and/or how to state acceptance criteria. Also, for some quality subcharacteristics, it might be necessary to add a third level, subdividing them into sub-subcharacteristics.

VII. Conclusions and Future Work

We believe in the importance of using standards and are concerned about the low adoption of RE standards in industry [20]. However, with respect to quality frameworks for RE and Digital Design, it is time to create a better framework than what the current standards and templates provide.

In this paper, we have sketched a new quality model that we believe to contribute a first step into this direction.

Obviously, further work is needed. We are still in the process of discussing new general considerations for quality models, such as the idea to have a stronger focus on system runtime. Furthermore, we need to polish the model and make it more self-contained by adding definitions for all terms used. After that, the next major step will be to undertake an empirical study for validating the model and its applicability.

References

[1] M. Glinz, *A Glossary of Requirements Engineering Terminology, Version 2.0.1*. International Requirements Engineering Board, IREB, 2022, https://www.ireb.org/downloads/#cpre-glossary (Accessed 2023-06-05).

[2] B. W. Boehm, J. R. Brown, and M. Lipow, "Quantitative Evaluation of Software Quality," in *2nd International Conference on Software Engineering, San Francisco*, 1976, pp. 592–605.

[3] J. A. McCall and M. T. Matsumoto, "Software Quality Measurement Manual, Vol. II," Rome Air Development Center, Tech. Rep. RADC-TR-80-109-Vol-2, 1980.

[4] "ISO/IEC 25010: Systems and Software Engineering — Systems and Software Quality Requirements and Evaluation (SQuaRE) — System and Software Quality Models," International Standard, 2011.

[5] M. Glinz, "A Risk-based, Value-oriented Approach to Quality Requirements," *IEEE Software*, vol. 25, no. 2, pp. 34–41, 2008.

[6] S. Robertson and J. Robertson, "Volere Requirements Specification Template," https://www.volere.org/templates/volere-requirements-specification-template/ (Accessed: 2023-06-05).

[7] M. Glinz, "On Non-Functional Requirements," in *15th IEEE International Requirements Engineering Conference (RE '07), Delhi, India*, 2007, pp. 21–26.

[8] T. Olsson, S. Sentilles, and E. Papatheocharous, "A Systematic Literature Review of Empirical Research on Quality Requirements," *Requirements Engineering*, vol. 27, no. 2, p. 249–271, 2022.

[9] S. Kopczyńska and J. Nawrocki, "Using Non-Functional Requirements Templates for Elicitation: A Case Study," in *4th IEEE International Workshop on Requirements Patterns (RePa), Karlskrona, Sweden*, 2014, pp. 47–54.

[10] M. Glinz, H. van Loenhoud, S. Staal, and S. Bühne, *Handbook for the CPRE Foundation Level according to the IREB Standard. Version 1.1*. International Requirements Engineering Board, IREB, 2022, https://www.ireb.org/downloads/#cpre-foundation-level-handbook (Accessed 2023-06-05).

[11] "ISO/IEC 9126: Software Engineering — Product Quality," Withdrawn International Standard, 1991.

[12] "ISO/IEC 9126-1: Software Engineering — Product Quality — Part 1: Quality Model," Withdrawn International Standard, 2001.

[13] "ISO/IEC DIS 25010:2022 (E): Systems and Software Engineering — Systems and Software Quality Requirements and Evaluation (SQuaRE) — Product Quality Model — Part: Product Quality Model," Draft International Standard, 2022.

[14] "ISO/IEC DIS 25019:2022(E): Systems and Software Engineering — Systems and Software Quality Requirements and Evaluation (SQuaRE) — Product Quality Model — Quality-in-Use Model," Draft International Standard, 2022.

[15] N. Bevan, "Quality in Use: Meeting User Needs for Quality," *Journal of Systems and Software*, vol. 49, no. 1, pp. 89–96, 1999.

[16] L. B. R. Oliveira, M. Guessi, D. Feitosa, C. Manteuffel, M. Galster, F. Oquendo, and E. Y. Nakagawa, "An Investigation on Quality Models and Quality Attributes for Embedded Systems," in *8th International Conference on Software Engineering Advances (ICSEA)*, 2013, pp. 523–528.

[17] C. Becker, R. Chitchyan, L. Duboc, S. Easterbrook, B. Penzenstadler, N. Seyff, and C. C. Venters, "Sustainability Design and Software: The Karlskrona Manifesto," in *37th International Conference on Software Engineering, Florence, Italy*, 2015, pp. 467–476.

[18] K. Lauenroth, "Digital als Material mit Form, Funktion und Qualität begreifen. [Understanding digital as a material with form, function and quality (in German)]," in *Digital Design@Bauhaus*. Bitkom, 2019, https://www.bitkom.org/sites/main/files/2019-09/20190910_sammelband-digital-designbauhaus.pdf (Accessed: 2023-06-05).

[19] K. Lauenroth, D. Gilbert, M. Kemper, K. Lehn, N. Seyff, M. Stade, and M. Trapp, *Digital Design Professional – Education and Training Handbook for the Digital Design Professional at Foundation Level (DDP FL). Version 1.1*. International Requirements Engineering Board, IREB, 2022, https://www.digitaldesign.org/media/pages/downloads/af8bf5f99a-1662713832/ddp_foundationlevel_handbook_en_v1.1.0.pdf (Accessed 2023-06-05).

[20] X. Franch, M. Glinz, D. Mendez, and N. Seyff, "A Study About the Knowledge and Use of Requirements Engineering Standards in Industry," *IEEE Transactions on Software Engineering*, vol. 48, no. 9, p. 3310–3325, 2022.