Software Quality-in-Use: A Systematic Literature Review

Yvette D. Hastings
Gianforte School of Computing
Montana State University
Bozeman, MT, USA
Email: yvettehastings@montana.edu

Ann Marie Reinhold
Gianforte School of Computing
Pacific Northwest National Laboratory
Montana State University
Bozeman, MT, USA
Email: reinhold@montana.edu

***Abstract*—Quality-in-use (QIU) describes how well software allows users to achieve goals within a specific context of use. Despite established standards, such as ISO/IEC 25010 and ISO/IEC 25019, the extent and consistency of QIU evaluations in software engineering research remain unclear. This paper presents a systematic literature review on QIU evaluations from 38 primary studies published between 2014 and 2024. We examine publication trends, software application domains, QIU frameworks, assessment methods, evaluated QIU aspects, and reported QIU issues. Results show that QIU evaluations are sporadic through time and concentrated in user-facing domains, such as healthcare, education, and e-commerce. Meanwhile, other domains are underrepresented. ISO/IEC 25010 was the most frequently used framework. However, frameworks varied in how QIU was operationalized and measured. Evaluations commonly focus on usability-related QIU aspects, while broader QIU aspects received less attention. Many studies also failed to explicitly report QIU issues. These findings highlight fragmentation in how QIU evaluations are conducted, emphasizing the need for more consistent, comprehensive, and domain-diverse evaluation approaches and reporting.**

***Index Terms*—software engineering, software product quality evaluations, quality-in-use, stakeholder perspective, context of use**

I. Introduction

Software applications are integral to modern life, supporting success in both personal and professional settings. With the importance of software impacting almost every aspect of our lives, it is imperative to assess software quality throughout the development lifecycle [1].

Software quality is evaluated at multiple stages of the development lifecycle. It is generally categorized into two dimensions: product quality, often associated with functional and technical attributes (e.g., code correctness, performance efficiency, code architecture, etc.; also known as functional requirements) [1], and product quality-in-use (QIU), which assesses how well a product meets user needs in real-world operation (also known as non-functional requirements) [2].

While product quality is essential, QIU is equally critical to measure and to achieve so that software products are developed and maintained to meet user expectations in its given context of use. The ISO/IEC 25019:2023 standard defines QIU as the "extent to which the system or product, when used in a specified context of use, satisfies or exceeds stakeholders' needs to achieve specified beneficial goals or outcomes" [2]. Evaluating QIU therefore extends beyond assessing software functionality alone and instead examines how well a software system supports user goals through QIU aspects such as beneficialness, acceptability, and freedom from risk. Therefore, incorporating QIU considerations throughout the software development lifecycle helps proactively address potential use-related issues such as improved user satisfaction, increased software adoption, and reduced post-release costs associated with technical debt [3], [4].

Despite these benefits, QIU is often overlooked throughout the development lifecycle, as developers often prioritize functional requirements before and after release [4]. This prioritization may be due to the high demand for rapid software development, leading to delayed, insufficient, or missing QIU evaluations. This discrepancy highlights a critical gap in software engineering research and practice. That is, while software quality is extensively studied, QIU remains underexplored, primarily due to the subjectivity and complexity associated with human-centered research [5].

The objectives of this systematic literature review (SLR) are to identify and summarize the existing literature on software QIU evaluations. Specifically, we aim to address the following research questions (RQs):

- **RQ1:** What are the temporal trends of QIU evaluations published between 2014-2024?
- **RQ2:** What primary software application domains have been evaluated for QIU?
- **RQ3:** What QIU standards or frameworks have been used?
- **RQ4:** What methods have been used to assess QIU (automated processes, NLP, sentiment analysis, questionnaires, etc.)?
- **RQ5:** What QIU aspects have been evaluated?
- **RQ6:** What QIU aspects have been identified as lacking (or cause for concern) for each software application domain?

With answers to these research questions, we aim to achieve a better understanding of the current state of QIU evalua-

This work is funded by the Montana State University Software Engineering and Cybersecurity Laboratory.

tions. Specifically, we aim to identify software application domains that are underrepresented in QIU evaluations and gain a better appreciation for the QIU aspects commonly and less commonly explored. From this knowledge, we aim to strengthen the scope and rigor of QIU evaluations (e.g., method consistency and aspect coverage) to better capture user-perceived software quality.

II. Methods

In this section, we present the methods used to conduct our SLR. We reviewed the primary literature associated with QIU software evaluations following the protocol established by Kitchenham and Charters [6]. Additional method details and a full list of included publications are provided in a supplemental document[1].

A. Planning

The authors met over multiple sessions to discuss and plan the purpose of the SLR. Based on our prior work, we identified gaps in QIU evaluations across various software domains. We therefore established RQs 1–6 (presented in Section I) to analyze publication trends in QIU evaluations over time, identify the software domains that have been evaluated, examine the QIU standards and methods employed, assess the scope of evaluated QIU aspects, and identify reported QIU issues across the literature. In this review, the term "QIU aspects" refers to the constructs operationalized in primary studies to evaluate QIU, regardless of the specific framework employed (e.g., ISO/IEC 25010, domain-specific usability models, or custom evaluation criteria).

The purpose of each research question is summarized below:

- RQ1 examines the patterns and trends in the publication of QIU evaluations over time.
- RQ2 identifies software application domains that are commonly evaluated and those that may be underrepresented.
- RQ3 determines the popularity of QIU standards used to guide evaluations.
- RQ4 examines whether evaluations rely on automated, hybrid, or manual methods and whether direct user perspectives are obtained.
- RQ5 assesses whether evaluations capture the full scope and intent of QIU and where they may be limited.
- RQ6 aids in identifying whether similar QIU issues persist across software application domains.

B. Define PICOC and Synonyms

Following planning, the primary author defined the PICOC (Population, Intervention, Comparison, Outcome, and Context) elements. This process refined the review objectives and allowed us to identify keywords and synonyms (Table S1) used to construct the search string. Based on the PICOC elements, the primary author constructed the following search string: ("quality in use" OR "quality-in-use") AND ("user") AND NOT "hardware". This search string focused our review on studies addressing QIU evaluations involving users, while excluding publications related to hardware assessments. Importantly, our search strategy required the explicit use of the term "quality in use." As a result, studies that examine related constructs (e.g., usability or user experience) without explicitly framing them as QIU may have been excluded. This decision was intentional, as the objective of this review is to characterize the extent to which QIU is explicitly operationalized and evaluated in the literature, rather than to capture the broader body of usability or UX research.

C. Inclusion and Exclusion Criteria

On December 19, 2024, we applied our search string to three electronic databases filtered between 2014-2024: IEEE Xplore[2], ACM Digital Library[3], and Engineering Village[4]. The search returned 206 publications related to QIU and users between the three databases (Table I). We deduplicated the list of studies based on the titles and restricted the list of publications to primary studies published in journals or conference proceedings. After applying these restrictions, we were left with 140 unique publications (Table I).

The primary author and a volunteer independently screened the 140 titles and abstracts of the publications to determine their relevance to the SLR and noted them for inclusion or exclusion based on the criteria outlined in Table II. The primary author and volunteer met after the screening process to review the papers marked as "include" or "exclude". Any paper designations that did not match were discussed until a unanimous decision was reached. After the screening process, we selected a total of 38 publications for inclusion in the review (Table I).

Given the exploratory nature of this review, no quality assessment scoring was applied. This decision is consistent with previous SLRs that aim to characterize method diversity over study rigor. The complete list of included studies is provided in Table S2 of the supplemental document, where each paper is assigned a study identifier (Paper ID 1–38).

D. Review of publications

The primary author conducted an in-depth review of each of the 38 selected publications. Information relevant to each research question was documented in an Excel file to support synthesis across studies. Extracted data included publication year, software application domain, QIU standard used, evaluation method, evaluated QIU aspects, and reported QIU issues (refer to Table S3 for a description of extracted data & the Excel file in our Zendo repo[1]). For the classification of software application domains, the authors assigned each study a primary application domain based on the software system evaluated and the context of use described in each study. The extracted data, results, and interpretations were subsequently

[1]https://doi.org/10.5281/zenodo.18791337

[2]https://ieeexplore.ieee.org

[3]https://dl.acm.org/

[4]https://www.engineeringvillage.com/

reviewed by both authors to confirm extracted data and to reduce reporting bias in the SLR.

TABLE I
DIGITAL LIBRARY SEARCH RESULTS BEFORE AND AFTER APPLYING THE INCLUSION AND EXCLUSION CRITERIA

Digital Library	Date Searched	Initial Search Result	Deduplicated Primary Publications	Final Selection
ACM Digital Library	12/19/2024	83	73	8
IEEE Xplore	12/19/2024	23	20	11
Engineering Village	12/19/2024	100	47	19
Total		206	140	38

TABLE II
INCLUSION AND EXCLUSION CRITERIA

Inclusion Criteria	Exclusion Criteria
Publications from 2014-2024	Secondary literature (e.g., reviews, mapping studies)
QIU evaluation	Book chapters
Written in English	Abstract-only publications
Journal and conference articles	Theses and dissertations
Software-focused studies	Proposals or conceptual papers
Primary research literature	Articles not related to software
	Articles not written in English
	Preprint articles

III. RESULTS & DISCUSSION

In this section, we describe the findings and discuss the implications for each of our RQs.

A. RQ1 Findings & Implications

We analyzed the publication years of QIU evaluations and found no clear or consistent temporal trend (Fig. 1). Across the 2014-2024 period, QIU evaluations were sporadic and unevenly distributed. The highest number of publications occurred in 2020, after which the overall number of QIU evaluation studies declined.

One explanation for this decrease is the COVID-19 pandemic, which limited researchers' ability to interact with users and conduct evaluations that rely on user perceptions. For example, Souza-Pereira et al. [7] (Table S2, Paper ID 22) reported that the COVID-19 pandemic made it impossible to work directly with healthcare professionals during their evaluation of healthcare applications. As a result, they remotely distributed a questionnaire and received only 18 responses, a relatively low number given the nature of their study. The observed decline after 2020 may also reflect changes in industry practices, such as rapid software releases [4] and the omission of formal, often costly and time-intensive, QIU evaluations.

Overall, the sporadic and low number of publication patterns suggests that QIU receives limited attention across published software quality studies.

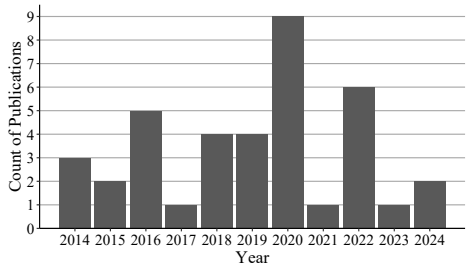

Fig. 1. Count of QIU publications by year from 2014-2024.

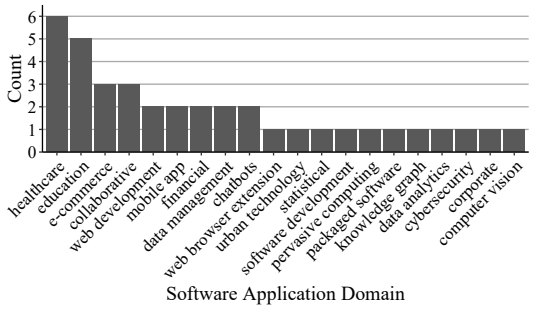

Fig. 2. Count of software application domains evaluated for QIU.

B. RQ2 Findings & Implications

After investigating the temporal trends in publications over time, we identified the software application domains evaluated for QIU. Healthcare, education, e-commerce, and collaborative software applications were the most frequently evaluated domains, collectively accounting for approximately 45% of the reviewed publications (Fig. 2; domains with counts ≥ 3; Table S2, Paper IDs 4, 7-8, 14-16, 18, 21-22, 25-27, & 31-35). This distribution suggests that QIU evaluations are predominantly conducted in user-facing and consumer-oriented domains.

In contrast, several application domains were underrepresented or absent in the literature. These include, but are not limited to, government and public-sector systems, military and defense applications, security-critical systems, and critical infrastructure systems. Extending QIU evaluations to underrepresented application domains is important because software failures in these areas (e.g., critical infrastructure or military and defense applications) can have consequences that extend beyond functional quality issues, including risk to human life, impairments to the environment, or high financial costs due to unmet user needs.

C. RQ3 Findings & Implications

Next, we examined the QIU standards and frameworks used across the literature to identify the most commonly adopted reference frameworks. Of the 19 distinct standards and frameworks used, the majority of studies employed ISO/IEC 25010 (count $= 14$; Fig. 3; Table S2, Paper IDs 4, 8, 11, 13–14, 17, 20–22, 27, 30, 32, & 35).

Several evaluation frameworks, such as the Waseda SQ Framework (WSQF) developed by Tsuda et al. [8] (Table S2,

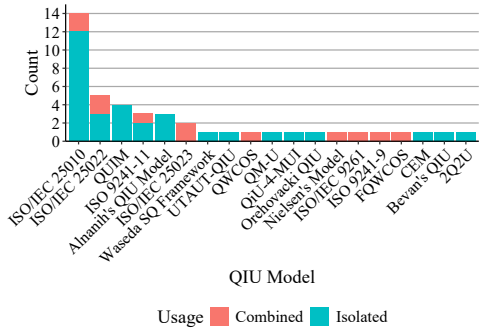

Fig. 3. Count of QIU standards used for evaluations. Red denotes standards that were used in addition to a second standard for a combined evaluation framework; blue denotes standards that were used in isolation, meaning no other standards were used with it.

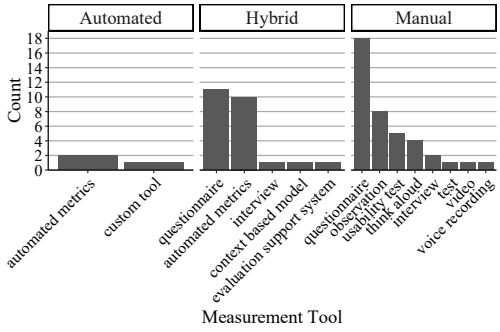

Fig. 4. Distribution of measurement tools across automated, manual, and hybrid QIU evaluation methods.

Paper ID 2), were built upon existing international standards. Tsuda et al. argue that the SQuaRE family of standards and their definitions, such as the ISO/IEC 25010 QIU definitions, are "rather general and abstract" and not readily implementable [8]. Motivated by this perceived lack of operational clarity, they sought to refine and clarify the QIU aspects defined in ISO/IEC 25010 to promote consistent evaluation practices. Similarly, multiple studies adopted Alnanih's QIU model (Table S2, Paper IDs 23–24 & 28), which was also derived from ISO/IEC 25010. These derived frameworks further illustrate the role of ISO/IEC 25010 as a common conceptual foundation for QIU evaluation.

We also observed that several studies used ISO/IEC 25022 in isolation (Fig. 3; 3 of 5 instances; Table S2, Paper IDs 12, 26, & 33). Because ISO/IEC 25022 primarily defines measurement metrics rather than conceptual definitions, applying it without an explicit reference framework may limit how clearly QIU constructs are interpreted. In contrast, the two studies that combined ISO/IEC 25022 with ISO/IEC 25010 or ISO/IEC 25023 (Table S2, Paper IDs 4 & 19) linked metrics to a clearer conceptual grounding. Overall, 32 of the 38 studies relied on a single standard or framework to guide evaluation, indicating limited cross-framework integration.

Although ISO/IEC 25010 is the predominant reference model, the diversity of derived and specialized frameworks suggests substantial variation in how QIU is operationalized and measured. This variation highlights the need for consistent and standardized approaches to QIU evaluations in order to improve comparability across studies and enable cumulative knowledge in QIU research.

D. RQ4 Findings & Implications

We next examined the types of QIU evaluation methods used across the publications. Three studies employed automated approaches (i.e., no direct contact with users), 24 used manual approaches (i.e., involved users or manually mapped metrics to QIU aspects), and 11 used hybrid approaches combining automated and manual techniques (Fig. 4).

To better understand how QIU was operationalized, we analyzed the tools and data sources used within each method category. Among the automated approaches, 2 studies derived metrics from system logs (Table S2, Paper IDs 20 & 27), while 1 developed a custom evaluation tool (Table S2, Paper ID 6). For example, Salomón et al. [9] (Table S2, Paper ID 20) used system logs to characterize user interactions and context of use in a sports betting application without directly collecting user perception data.

Manual methods most frequently relied on direct user input. Nineteen of the 24 manual studies (Table S2, Paper IDs 4–5, 12, 14–18, 21–25, 31–36) employed questionnaires and/or interviews to capture user perceptions. These were often complemented by observational techniques such as think-aloud protocols, video recording, or usability testing sessions.

All 11 hybrid studies (Table S2, Paper IDs 1–3, 7–11, 13, 30, 37) combined user-reported measures with automated data collection or metric computation. Such approaches offer the potential to integrate subjective user perceptions with objective behavioral indicators, providing a broader perspective on QIU.

Overall, manual and hybrid approaches predominantly relied on questionnaires or user observation to directly capture user perceptions. In contrast, automated approaches focused on behavioral or system-derived indicators and did not directly capture subjective user perspectives. Given that QIU is fundamentally concerned with users' perceptions and experiences, its evaluation requires the inclusion of manual or hybrid methods that incorporate direct user input, rather than relying solely on automated measures.

E. RQ5 Findings & Implications

After identifying the evaluation methods, we examined which QIU aspects were assessed across the literature. Across the 38 publications, 37 distinct QIU aspects were reported (Fig. 5). The majority of studies ($> 74\%$) evaluated efficiency, effectiveness, and/or satisfaction (Table S2, Paper IDs 2–18, 20–28, 30–35, 38). Other aspects were considered less frequently, including user error protection ($\approx 24\%$), learnability ($\approx$ 21%), productivity ($\approx$ 18%), and freedom from risk and context coverage (both $\approx 16\%$). All remaining aspects appeared in fewer than 14% of the studies.

The wide range of aspects evaluated across studies indicates substantial variation in how QIU frameworks are operational-

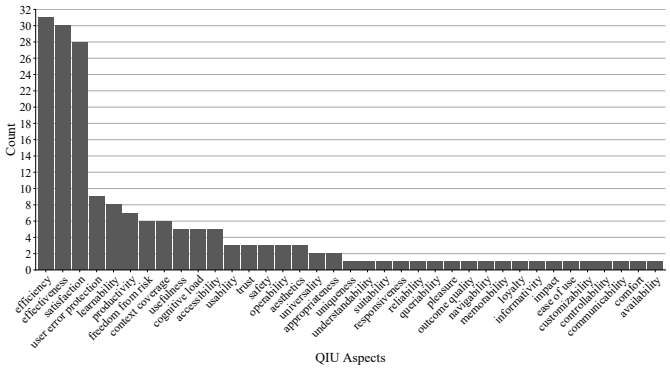

Fig. 5. QIU aspects evaluated across publications.

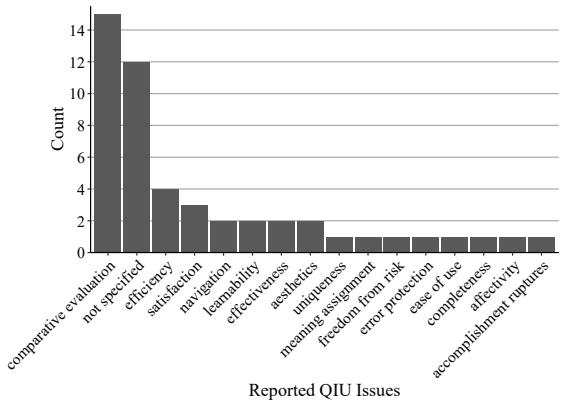

Fig. 6. Reported outcomes of QIU evaluations across the reviewed publications.

ized. This variation complicates comparisons across software domains and makes it difficult to determine which aspects of QIU may be consistently under-addressed. Notably, many evaluations emphasized aspects traditionally associated with usability (i.e., efficiency, effectiveness, and satisfaction) while comparatively less attention was given to other QIU considerations, such as freedom from risk and context coverage described in the ISO/IEC 25010 and 25019 standards [1], [2]. Several authors explicitly framed these usability-related measures as the primary focus of QIU studies [10], stating, for example, that "evaluating the usability of a system is particularly useful when aiming to improve the user interface or to establish the quality in use within a given context" [11], and that "overall customer satisfaction is the most crucial quality-in-use metric" [12] (Table S2, Paper IDs 13, 11, 34, respectively). In these and similar studies, high usability (a measure primarily concerned with user–interface interaction) is effectively treated as synonymous with high QIU.

Focusing predominantly on usability-related aspects may limit the scope of QIU evaluations and leave other important aspects insufficiently examined. For example, historical safety-critical software failures, such as Therac-25 [13] and the Boeing V-22 incidents [14], illustrate the broader consequences that can arise when factors related to risk mitigation or operational context are overlooked. Such failures can result in the loss of human life, create substantial economic harm to organizations, or compromise personal information.

Collectively, these results suggest that current QIU evaluations remain narrowly focused on usability-related issues. Therefore, it is imperative that future research prioritize balanced approaches that capture the full spectrum of QIU aspects.

F. RQ6 Findings & Implications

The final objective of this review was to identify QIU issues reported in the selected publications (Fig. 6). Only 11 of the 38 studies (Table S2, Paper IDs 1–2, 7, 10–12, 14, 33, & 36–38) explicitly described QIU issues (i.e., low efficiency, risk concerns, low operability) identified through their evaluations. The remaining studies either did not specify QIU issues (count = 12; Table S2, Paper IDs 3–6, 8, 17–20, 22, 26, & 30) or used QIU evaluations primarily for comparative purposes, such as selecting between software tools or assessing improvements across versions (count = 15; Table S2, Paper IDs 9, 13, 15–16, 21, 23–25, 27–29, 31–32, & 34–35).

Among the 11 studies that reported QIU issues, more than 25% identified problems related to QIU aspects of efficiency and/or satisfaction (Table S2, Paper IDs 1–2, 7, 10, 14, & 33). This aligns with the earlier observation that many evaluations operationalized QIU primarily through usability-oriented measures (Section III-E). Because operational definitions constrain the range of observable outcomes, this emphasis likely shapes which QIU issues are detected and reported. Consequently, aspects of QIU that extend beyond usability, such as freedom from risk, trustworthiness, and compliance, may be systematically underrepresented in the literature.

Issues beyond the QIU aspects of efficiency, effectiveness, and satisfaction were reported in fewer than 17% of the 11 studies. Although less common, such reports demonstrate that some evaluations adopted a broader perspective on QIU, encompassing both user experience and system performance, enabling more comprehensive recommendations for improvement.

Studies using QIU frameworks for comparative analysis typically did not frame their findings as issues. Rather, they presented results as relative performance and QIU differences. In addition, several studies reported achieving acceptable QIU based on aggregate scores exceeding predefined thresholds [15] (Table S2, Paper ID 5) or presented results in figures and tables without explicitly discussing potential concerns (Table S2, Paper IDs 4, 18–19, & 26). In some cases, qualitative user feedback suggesting possible issues was noted but not explored further because it fell outside the primary study scope [16] (Table S2, Paper ID 3).

Together, these patterns indicate that while QIU is measured, explicit reporting of QIU issues is less common. Therefore, there is a need for a more transparent and systematic reporting of QIU issues to better support cross-domain comparisons, knowledge transfer, and actionable QIU improvement.

IV. Threats to Validity

In this section, we identify and discuss the threats to validity for our study. With regards to internal and construct validity, study screening, data extraction, and classification were performed manually. Although the primary author and a second reviewer independently screened studies and jointly reviewed extracted data, inclusion decisions and the assignment of software domains, QIU frameworks, methods, and aspects required interpretation and may have introduced researcher bias.

Regarding external and conclusion validity, our search was limited to three major digital libraries, English-language publications, and studies from 2014-2024. The search string focused on the term "quality in use," which may have excluded relevant work using alternative terminology such as usability or user experience. In addition, we did not apply a formal quality assessment, meaning studies of varying method rigor were synthesized equally. Because our synthesis depended on the completeness and clarity of reporting in primary studies, and because QIU issues were not always explicitly articulated, some findings required interpretation of graphical results or implicit descriptions. As a result, the trends identified in this review reflect reported practices and documented evidence within the literature, rather than a direct measurement of the underlying state of QIU evaluation research.

V. Conclusion & Future Work

This SLR examined how software QIU has been evaluated across research published between 2014 and 2024. The findings show that QIU evaluations are sporadic over time, concentrated in user-facing software domains, and vary by frameworks, methods, and aspects assessed. Many studies rely heavily on usability-related measures, while broader QIU aspects, such as freedom from risk and context coverage, are less frequently considered. In addition, inconsistent reporting practices and the frequent omission of QIU issues limit opportunities for cross-study comparison and product improvement.

Together, these results indicate that QIU evaluation remains fragmented across software application domains and lacks standardized documentation practices. As future work, we plan to conduct QIU evaluations in underrepresented software application domains. Issues identified in these evaluations will support cross-domain comparisons to determine whether concerns are domain-specific or shared across domains. We also plan to develop a structured quality documentation tool that enables researchers and practitioners to consistently record and share software quality evaluations and findings for both product quality and QIU. By improving transparency, comparability, and accessibility of quality evaluation data, our efforts aim to support more comprehensive and reproducible QIU research and practice.

Acknowledgment

We thank the Montana State University Software Engineering and Cybersecurity Laboratory (SECL) for their support of this work. Special thanks to Dr. D. Reimanis for their continued feedback during the planning portion of this work and for their assistance in the publication screening process. We'd also like to thank Ryan Cummings for their continued support and feedback of this work.

References

[1] ISO/IEC 25010:2023(E), *Systems and software engineering –Systems and software Quality Requirements and Evaluation (SQuaRE) – Product quality model*. Vernier, Geneva, Switzerland: International Organization for Standardization, 2023.

[2] ISO/IEC 25019:2023(E), *Systems and software engineering –Systems and software Quality Requirements and Evaluation (SQuaRE) – Quality-in-use model*. Vernier, Geneva, Switzerland: International Organization for Standardization, 2023.

[3] L. C. d. F. Lage, M. Kalinowski, D. Trevisan, and R. Spinola, "Usability technical debt in software projects: A multi-case study," in *2019 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*, 2019, pp. 1–6.

[4] A. F. F. Costa, A. B. D. S. Marques, I. S. Santos, and R. M. D. C. Andrade, "Towards a process to manage usability technical debts," in *Proceedings of the XXXVI Brazilian Symposium on Software Engineering*, ser. SBES '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 241–246.

[5] S. Abrahão, F. Bourdeleau, B. Cheng, S. Kokaly, R. Paige, H. Stöerrle, and J. Whittle, "User experience for model-driven engineering: Challenges and future directions," in *2017 ACM/IEEE 20th International Conference on Model Driven Engineering Languages and Systems (MODELS)*, 2017, pp. 229–236.

[6] B. Kitchenham and S. M. Charters, "Guidelines for performing systematic literature reviews in software engineering," University of Durham, EBSE Technical Report EBSE-2007-01, July 2007.

[7] L. Souza-Pereira, N. Pombo, and S. Ouhbi, "Software quality: Application of a process model for quality-in-use assessment," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 7, pp. 4626–4634, 2022.

[8] N. Tsuda, H. Washizaki, K. Honda, H. Nakai, Y. Fukazawa, M. Azuma, T. Komiyama, T. Nakano, H. Suzuki, S. Morita, K. Kojima, and A. Hando, "Wsqf: Comprehensive software quality evaluation framework and benchmark based on square," in *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*, 2019, pp. 312–321.

[9] S. Salomón, R. Duque, J. L. Montaña, and L. Tenés, "Towards automatic evaluation of the quality-in-use in context-aware software systems," *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, no. 8, pp. 10 321–10 346, 2023.

[10] E. Ben Ayed, C. Kolski, R. Magdich, and H. Ezzedine, "Towards a context based evaluation support system for quality in use assessment of mobile systems," in *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2016, pp. 004 350–004 355.

[11] M. Smuts, B. Scholtz, and A. Calitz, "Design guidelines for business intelligence tools for novice users," in *Proceedings of the 2015 Annual Research Conference on South African Institute of Computer Scientists and Information Technologists*, ser. SAICSIT '15. New York, NY, USA: Association for Computing Machinery, 2015.

[12] E. U. Okike and S. Mosanako, "Measuring customer satisfaction on software-based products and services: a requirements engineering perspective," in *Fourth International Congress on Information and Communication Technology: ICICT 2019, London, Volume 2*. Springer, 2020, pp. 31–45.

[13] Ethics Unwrapped, University of Texas at Austin, "Therac-25 Case Study," n.d., accessed: Dec. 14, 2025. [Online]. Available: https://ethicsunwrapped.utexas.edu/case-study/therac-25

[14] The Air Current, "V-22 Ospreys Have Serious Unresolved Safety Risks, NAVAIR Review Confirms," n.d., accessed: Dec. 14, 2025. [Online]. Available: https://theaircurrent.com/aviation-safety/v-22-ospreys-safety-risks-navair-review-confirms/

[15] E. Osagie, M. Waqar, S. Adebayo, A. Stasiewicz, L. Porwol, and A. Ojo, "Usability evaluation of an open data platform," in *Proceedings of the 18th Annual International Conference on Digital Government Research*, ser. dg.o '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 495–504.

[16] J. D. Delano, H. K. Jain, and A. P. Sinha, "System design through the exploration of contemporary web services," vol. 9, no. 3. New York, NY, USA: Association for Computing Machinery, Oct. 2018.