

# Towards Resolving Software Quality-in-Use Measurement Challenges

<sup>1</sup>Issa Atoum, <sup>2</sup>Chih How Bong, <sup>3</sup>Narayanan Kulathuramaiyer

<sup>1,2,3</sup> Faculty of Computer Science and Information Technology, University Malaysia Sarawak, 94300 Kota Samarahan, Sarawak, Malaysia

<sup>1</sup>[atoum@siswa.unimas.my](mailto:atoum@siswa.unimas.my), <sup>2</sup>[chbong@fitunimas.my](mailto:chbong@fitunimas.my), <sup>3</sup>[nara@fitunimas.my](mailto:nara@fitunimas.my)

## ABSTRACT

Software quality-in-use comprehends the quality from user's perspectives. It has gained its importance in e-learning applications, mobile service based applications and project management tools. User's decisions on software acquisitions are often ad hoc or based on preference due to difficulty in quantitatively measure software quality-in-use. However, why quality-in-use measurement is difficult? Although there are many software quality models to our knowledge, no works surveys the challenges related to software quality-in-use measurement. This paper has two main contributions; 1) presents major issues and challenges in measuring software quality-in-use in the context of the ISO SQuaRE series and related software quality models, 2) Presents a novel framework that can be used to predict software quality-in-use, and 3) presents preliminary results of quality-in-use topic prediction. Concisely, the issues are related to the complexity of the current standard models and the limitations and incompleteness of the customized software quality models. The proposed framework employs sentiment analysis techniques to predict software quality-in-use.

**Keywords:** *Software Quality-in-use; ISO 25010; SQUARE series; Sentiment analysis*

## 1. INTRODUCTION

With thousands of software published online, it is essential for users to find the software that matches their stated or implied needs. Users often seek better software quality. Garvin [1] identified five views/approaches of quality. The nearest definition in this paper is the user based approach definition "meeting customer needs". If the customer is satisfied, then product or service has good quality. It has been implemented in mobile-based applications[2]–[4] and Web applications[5]–[7].

Software quality can be conceptualized from three dimensions; the quality characteristics, the quality model, and software quality requirements. A Quality characteristic is "category of software quality attributes that bears on software quality" [8, p. 9]. Quality requirements are what the user needs in the software such as performance, user interface or security requirements. The quality model is how quality characteristics are related to each other and to the final product quality. Measuring the software quality will check if user requirements are met and decide the degree of quality.

The ISO/IEC 25010:2010 standard (ISO 25010 hereafter), a part of a series known as the Software Quality Requirements and Evaluation (SQuaRE), defines systems' quality as "the degree to which the system satisfies the stated and implied needs of its various stakeholders, and thus provides value" [9, p. 8]. The ISO 25010 has two major dimensions: Quality-in-use (QinU) and Product Quality. The former specifies characteristics related to the human interaction with the system and the latter specifies characteristics intrinsic to the product. QinU is defined as "capability of a software product to influence users' effectiveness, productivity, safety and satisfaction to satisfy their actual needs when using the software product to achieve their goals in a specified context of use" [8, p. 17]. The QinU model consists of five characteristics: effectiveness, efficiency, satisfaction, freedom from risk and

context coverage. Table 1 illustrates the definition of these characteristics.

**Table 1:** definitions of quality-in-use characteristics as defined by the ISO 25010 standard

Characteristic	Definition
<b>Effectiveness</b>	Accuracy and completeness with which users achieve specified goals (ISO 9241-11).
<b>Efficiency</b>	Resources expended in relation to the accuracy and completeness with which users achieve goals (ISO 9241-11).
<b>Freedom From Risk</b>	Degree to which a product or system mitigates the potential risk to economic status, human life, health, or the environment.
<b>Satisfaction</b>	Degree to which user needs are satisfied when a product or system is used in a specified context of use.
<b>Context Coverage</b>	Degree to which a product or system can be used with effectiveness, efficiency, freedom from risk and satisfaction in both specified contexts of use and in contexts beyond those initially explicitly identified.

### 1.1 Problem Statement

This paper investigates these problems 1) there are many challenges that need to be tackled in order to measure QinU systematically. However, current literature reviews on software QinU does not identify or explain them. To the best of our knowledge this is the first work that specifically identifies and explains the problems towards measuring QinU. 2) Insufficient research on other possible research directions to tackle the first problem. To our knowledge, little work target to resolve QinU problem [10].

**1.2 Research Contributions**

- This paper identifies and explains several problems while measuring software QinU using the standard and customized quality models. This paper is the first that surveys several quality models and explains various challenges to measure QinU. In brief, most of the challenges in ISO standard models are related to the complication and incompleteness of the documents. On the other hand, customized quality models are subject to incomplete models that are designed for their own specific needs.
- Proposes a novel framework to predict software QinU from software reviews. Given the issues related to measuring QinU a framework is presented to resolve these issues. The framework is based on sentiment analysis, an emerging branch of Natural Language Processing. Sentiment analysis or opinion mining targets to analyze textual user judgments about products or services[11], [12]

First major software quality-in-use related models are illustrated. Then, the quality-in-use measurement challenges are explained. Next, a proposed approach is presented and finally, the paper is concluded.

**2. SOFTWARE QUALITY-IN-USE MODELS**

There have been many works in software quality models but to our knowledge, no research has been conducted to summarize the main problems in measuring quality-in-use. Measuring software quality-in-use can be divided in two main frameworks; the standard and customized model frameworks.

**2.1 Standard Frameworks**

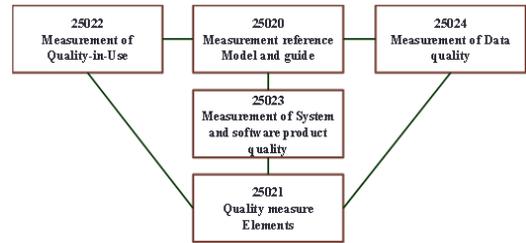
There have been many standards that can support software quality, but many of them are rather check list guide. For example, the ISO 9000 family has been criticized in literature not to be used for software [13]. The ANSI/IEEE 730-2002[14] support quality assurance plans. ISO/IEC 15504[15] or as it is known Software Process Improvement and Capability Determination (SPICE), is a set of technical standards documents for the computer software development process and related business management functions. These standards are not designed to address quality-in-use nor specific characteristics of software product quality.

Recently, the Software Product Quality Requirements and Evaluation (SQuaRE) ISO standard series are a result of blending the ISO/IEC 9126 and ISO/IEC 14598 series of standards. The purpose of the SQuaRE series of standards is to assist developing and acquiring software products with the specification of quality

requirements and evaluation. From the viewpoint of the stakeholders the quality requirements are specified, the quality of the product is evaluated based on this specification utilizing chosen quality model, quality measurement and quality management process.

2503n Quality Requirement Division	2501n Quality Model Division	2504n Quality Evaluation Division
	2500n Quality Management Division	
	2502n Quality Measurement Division	
ISO/IEC 25050 – 25099 SQuaRE Extension Division		

**Fig 1:** Organization of Square series of International Standards



**Fig 2:** Structure of the Quality Measurement division

To measure QinU effectively, five divisions of the SQuaRE series have to be considered the ISO 2502n to ISO 25024 and in line with the ISO 25010 model as shown in Fig. 1. Technically and, more precisely, the QinU Measurement Standard ISO 25022 has to be considered in the context of four other standards: the Measurement Reference Model and Guide ISO 25020; the Measurement of Data Quality 25024, the Measurement of System and Software Product Quality ISO 25023, and Quality Measure Elements Standards ISO 25021. Fig. 2 depicts the relationship between the ISO/IEC 25022 and other ISO/IEC 2502n division of standards.

While these standards provide the freedom of customization, they need careful quality assurance to provide apparent integration between related standards. They also suffer to detail how the customization need to be carried out.

**2.2 Customized Software Quality Models**

Below are some of related models grouped in logical groups.