Quality Evaluation and Testing Method of Industrial Application Based on ISO/IEC 25000 SQuaRE Standard

Xiuming Yu, Wenpeng Li*, Yangyang Zhang, Zengzhi Liu
Software engineering and Appraisal Center
China Electronic Standardization Institute
Beijing, China
yuxiuming @cesi.cn, liwp@cesi.cn*, zhangyy@cesi.cn, liuzz@cesi.cn

CEUR-WS.org/Vol-2800/paper-03.pdf

***Abstract*—More attention is paid to the quality of industrial application software with the growing importance of testing in industrial development. The paper gives a discussion about 12 characteristics of quality model regarding industrial product attributes and software product attributes at the same time. A practical software testing case is present to provide for the effectiveness of methodology. The contribution of the paper is to identify quality model of industrial application software for software testing.**

Keywords*—*software quality; quality models; quality standards; ISO; industrial;

I. Introduction

Industrial application is a kind of software carrying industrial knowledge, experience or best experience, which addresses specific business requirements in industrial domain, such as research and development, manufacturing, operation maintenance, operating management. With the development of industrial internet and cloud computing, more and more industrial applications adopt advanced software engineering techniques and methods, such as DevOps and agile, to speed up the release of software productions, but software quality still needs to be considered at the forefront of any software development.

According to ISO/IEC 25000 [1], software quality is defined by "capability of software product to satisfy stated and implied needs under specified conditions". And from IEEE 730-2014 Standard for Software Quality Assurance Processes [2], "quality depends on the degree to which those established requirements accurately represent stakeholder needs, wants and expectations". Stakeholder value is expressed in requirements. The definition of software quality is related with requirements. In a short way, software quality measures whether software meets its requirements.

Software quality model is useful to considering an overall understanding of software quality. In [3], we have discussed the quality model of industrial application software. Some quality characteristics are given based on ISO/IEC 25010:2011[4]. We think that industrial application software carries industrial product attributes and software product attributes simultaneously. Therefore, we classify those characteristics into three groups: industrial properties, software product quality properties and software quality in use properties. In this paper, we would like to present a case study to explain software product quality and software quality in use in detail.

II. Quality Evaluation and test process

According to the ISO/IEC 25040:2011[5] SQuaRE evaluation process and ISO/IEC/IEEE 29119-2:2015 [6] test process standard, we give a quality evaluation and test process diagram to explain the procedure of evaluation quality of an industrial software in figure 1.

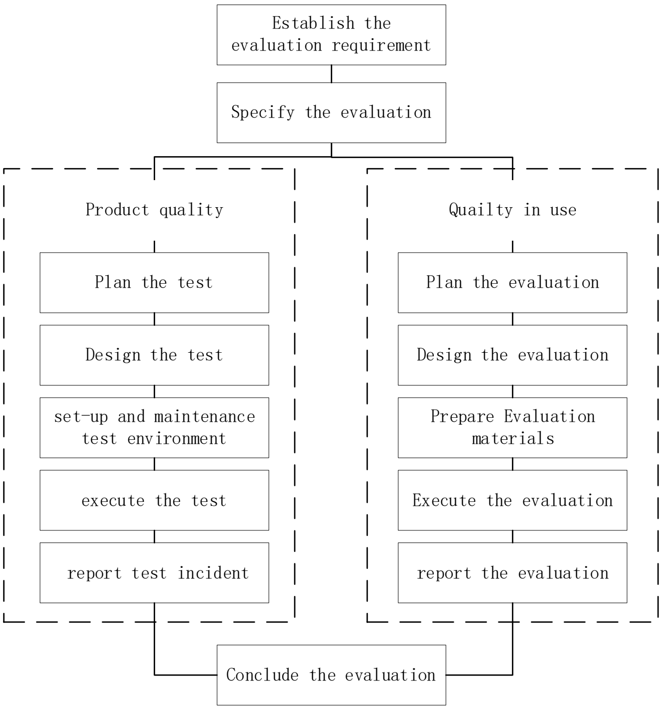

Fig. 1. Quality evaluation and test process diagram

The purpose of establishing the evaluation requirements is to identify the purpose and scope of industrial application in the evaluation, and confirm the stringency of the evaluation.

The purpose of specifying the evaluation is to confirm the quality elements and decision criteria.

Product quality is measured by software testing, which contains test planning, test design, test environment set-up and maintenance, test execution, test incident reporting. While quality in use is measured by manual review and evaluation.

The purpose of conclude the evaluation is to review the results and create report, and provide report and feedback to the delegate.

Copyright © 2020 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

III. DISSCUSSION OF QUALITY CHARACTERISTICS

In this part, we will discuss 12 characteristics: product quality, including functionality, performance efficiency, compatibility, usability, reliability, security, maintainability, portability; and quality in use, including effectiveness, use efficiency, satisfaction, and risk resistance.

Most industrial applications need to be customized to meet the requirements of different manufacturing scenarios, and the functions of the same industrial application vary greatly in different environments. So context coverage of quality in use in ISO/IEC 25010:2010 is tailored, while functionality appropriateness needs to be considered comprehensively.

A. Functionality

The extent to which the product's feature set covers the specified tasks and user goals. For example, the product description can describe the sufficiency, completeness, and coverage of the function realization of the software function. In functional correctness, industrial application has particularly high requirements for safety, especially in manufacturing area. Different from common software, most industrial applications need to be customized according to application industries, application scenarios and industrial equipment, so functional appropriateness may be different on the same application. For example, the optimum temperature of oil production line is different between production processes and plants. Functional compliance is an important consideration according to the Law, regulation and standard of safe and production requirements.

B. Performance efficiency

The requirements for efficiency are similar to common software. The statement of efficiency should include operating environment system configuration elements (usually including software, hardware, network, data environment, etc.) and performance index elements (such as business response time, resource occupancy, throughput, start-up time and loading time, disk I/O utilization, business processing response time under a certain number of concurrent users, etc.).

C. Compatibility

The product description needs to substitute another designation for the same purpose for the industrial application in the same environment. The capabilities of industrial application are explained. For example, name the software compatible with the software and the incompatible software. The degree to which two or more systems, products, or components are able to exchange information and use the information that has been exchanged. For example, whether a program can communicate with each other in two or more different systems. The biggest challenge in industrial application compatibility testing is the diversity of file types and industry protocols, including standardization protocols and privatization protocols.

D. Usability

The degree to which users can identify whether a product or system is appropriate for their needs. For example, software operation process, help information, documentation, website home information, especially in scenario with domain knowledge.

Product description shall describe the ability of industrial application users to learn their applications. For example, software provides measures to help users learn, including help documents, online counseling and so on.

Product description shall describe the ability of industrial application users to operate and control it. For example, whether it is easy for users to operate and control the software, and measures to guide users to operate. Whether the human-machine interface is friendly, and whether the interface design is scientific and reasonable, easy to operate, etc.

The degree to which the system prevents users from making mistakes. For example, mutex button is to set, preventing the user from wrong operation of the prompt, etc.

The degree to which a product or system can be used by individuals with universal characteristics and capabilities within a given usage environment. For example, individual understanding of the product, specific production safety protection.

E. Reliability

The product description shall state the ability of the industrial application to provide corresponding services to users when it is used under the software and hardware environment or other special conditions (such as certain load pressure) that meet its requirements. For example, software fault density, defect severity, integrity level, etc.

The degree to which a system, product, or component can operate and be accessed when it needs to be used is a combination of maturity, fault tolerance, and resiliency.

The product description shall state whether the software can provide users with the corresponding service in case of incorrect operation and parameter passing due to illegal data, illegal operation, misoperation and other reasons.

In the event of a software failure, the product description should state what steps to take to rebuild the ability to provide the appropriate service to the user and to recover the directly affected data.

Software failure can be manifested in the following situations:

- Crash: no output of software.
- Speed mismatch: the speed of data (input) or output with the requirements of the system is not operating.
- Insufficient calculation accuracy: due to insufficient data collection or algorithm problems. The calculation accu-racy of the output parameter value does not meet the requirements.
- Defective output: missing some necessary output values.

The output items are redundant: the software outputs data/commands that the system does not expect.

Measures to avoid software failure can be:

- Restart software;
- Restore the backed-up data;
- Restore data with one click;
- Incorrect operation prompt;
- Contact the service provider.

F. security

Products need to ensure that users are authorized to access the data. Protect information in its original state so that it remains authentic. For example, information encryption, high coupling low cohesion and so on. In the network environment, the two parties of information exchange cannot deny their behavior of sending or receiving information during the exchange. According to the user's activities in the system, it can be traced back to the user, that is, to provide a basis for tracing security problems that occur. The system can determine whether the information source is true or false. Security compliance is important according architecture and operating environment.

G. Maintainability

A system or computer program composed of multiple independent components, in which the change of one component has little or no impact on other components. Information can be applied to multiple systems or used in other constructions. The product description shall describe the defect or failure reason in the industrial application diagnosis software or the ability to identify the part to be modified. Whether the software supports failure diagnosis function, status monitoring function, etc. The product description needs to explain the industrial application ability to enable the specified modification to be realized. For example, is it easy to upgrade the software? The product description shall explain the ability of the industrial application to enable the modified software to be confirmed. Some code-level test, such as cyclomatic complexity, fan-in and fan-out, number of parameters, can be considered.

H. Portability

Product descriptions need to describe the industrial application's ability to adapt to different specified environments without using additional activities or means. For example, data structure, adaptability of software and hardware environment, etc. The product description needs to explain the ability of the industrial application to be installed in the specified environment. For example, the software installation method is custom or quick installation, software reinstallation, etc. The product description shall explain the situation where the industrial application replaces another designated industrial application of the same purpose under the same environment. For example, the new version of the software replaces the old version.

I. Effectiveness

Software products enable users to obtain the ability to meet the specified goals of accuracy and completeness in a specific use environment. Check the product description if is based on the user＇s error frequency and task completeness requirements.

J. Use efficiency

The software product's ability to use an appropriate amount of resources related to the obtained efficiency in a specific use environment of the software product. Related resources include intelligence, physical strength, time, materials, and financial resources. Verify the time consumption, resource utilization and capacity requirements of the software computer system in the product description. For example, the product description expects the target to complete a specified task in 5 seconds. Then the accuracy can be compared or Verify the time characteristics for measurement.

K. Satisfaction

The degree to which the user＇s requirements are met in the specified use environment of the product or system. Satisfaction is a kind of psychological state. The sense of pleasure when the needs are met is the relative relationship between the customer＇s prior expectations of the product or service and the actual feelings they get after actually using the product or service.

L. Risk resistance

The risk originates from many uncertain factors. Once it occurs, it will cause positive or negative effects. Therefore, certain measures should be taken to prevent the occurrence of the risk. The extent to which a product or system mitigated a potential risk in terms of economic status, human life, health or the environment. This includes economic risk mitigation, health and safety risk mitigation, and environmental risk mitigation.

IV. EXAMPLE

Defect prediction and health management industrial application is a software based on B/S and microserver architecture, developed by JAVA language, running on industrial internet platform, and using technical methods such as data transmission, big data storage, equipment modeling, diagnostic analysis and so on to establish equipment state evaluation models. Comprehensively evaluate the equipment status from five dimensions of communication (data quality), energy efficiency level, operating condition, health state and operation and maintenance quality, to identify fault components, fault mode, fault cause, fault level and improvement measures.

The independent evaluator needs to give a quality evaluation according to a group standard [7], which descripted in [3] detailly. The purpose of evaluation is to Determine the level of defect prediction and health management industrial application ranging from 1 to 5.

In the measurement of product quality, the testing requirements of performance efficiency and portability are lower because this application adopts SaaS service model; while testing requirements of functionality and security are higher because this application need to maintain the accuracy of forecasts, security of data, and conformance with relevant standards in the power industry, which this software mainly serves.

As the software involves a great deal of industry domain knowledge, the evaluation of quality in use adopts the way of expert reviews. Representatives of experts attending the conference rate the quality of use based on materials and experience.

According to the result of evaluation, this application meets the requirements of product documentation, and its product quality and quality in use meet the requirements of level 4 of T/CESA 1046-2019.

REFERENCES

[1] International Organization for Standardization. Systems and Software Engineering: Systems and Software Quality Requirements and Evaluation (SQuaRE): Guide to SQuaRE[M]. ISO/IEC, 2014.

[2] Heimann D. IEEE Standard 730-2014 Software Quality Assurance Processes[J]. IEEE Computer Society, New York, NY, USA, IEEE Std, 2014, 730: 2014.

[3] Zhang Y, Liu X, Sun X. Research on Quality Model of Industrial Application Software. Proceedings of the 1st International Workshop on Experience with SQuaRE Series and Its Future Direction co-located with 26th Asia-Pacific Software Engineering Conference (APSEC 2019)Malaysia: CEUR workshop Proceedings, Vol-2545:22-24.

[4] ISO/IEC 25010:2011, Systems and software engineering: systems and software quality requirements and evaluation (SQuaRE) - system and software quality models, 2011.

[5] IEEE. ISO/IEC 25040:2011, Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Evaluation process, 2011

[6] ISO/IEC/IEEE 29119-2:2013, Software and systems engineering — Software testing — Part 2: Test processes, 2013

[7] T/CESA 1046-2019, Classification, grading and evaluation of industrial application , 2019