Measuring software quality with usability, efficiency, and portability characteristics

Sholiq¹, R A Audá, A P Subriadi³, A Tjahyanto⁴, and A D Wulandari⁵

¹,²,³,⁴,⁵ Department of Information Systems, Institut Teknologi Sepuluh Nopember
Kampus ITS Sukolilo-Surabaya 60111, Surabaya, Indonesia
Email: sholiq@is.its.ac.id¹

Abstract. The paper aims to measure the quality of the “Application of software cost estimation (AoSCE)” produced by the author in the previous study. The quality characteristics measured include usability, efficiency, and portability using the ISO 9126-1 standard. The measurement results indicate that the three characteristics are good, but the usability characteristics still need improvement. The interesting of this paper is that testing procedures and worksheets can be applied to measure the quality of other software by adjusting the conditions of the test object.

1. Introduction
One factor that must be considered for software project development is software quality factor. The quality of the software is a significant factor that guarantees that the software produced can meet client requirement [1]. Several software qualities models have been created to measure the software quality, which the most popular models include Boehm, McCall, IEEE, and ISO 9126-1 [2] [3]. One of the popular models is the model of ISO 9126-1 which has six characteristics, and then the model has been revised with the ISO 25010 model [4] [5].

In the quality of software using the ISO 9126-1 standard (or ISO 25010 standard), there are six characteristics, namely: functionality, reliability, maintainability, usability, efficiency, and portability [6]. In the previous paper [7], we have assessed the quality of the AoSCE, which is the product of the previous study. The quality assessment carried out includes three characteristics, (maintainability, reliability, and functionality). To get a comprehensive quality dimension is necessary to measure quality using other characteristics, namely usability, efficiency, and portability.

According to Rehman and Majumdar [8], those usability characteristics are the key characteristics of all other characteristics, which evaluates usability characteristics needed to improve software quality. The efficiency characteristic is a data envelopment analysis (DEA) model that includes customer perceptions about the quality of services delivered through the application. Therefore, project managers involved in software projects should consider carrying out efficiency analyzes, such as meeting following guiding principles [9]. At the same time, portability characteristics state that portability testing is done by accessing applications using several different devices or platforms [10].

Besides the importance of three characteristics of software quality, namely usability, efficiency, and portability, another reason to measure software quality using these characteristics is to complement the measurement of quality characteristics that have been done in previous papers [7].

The results of software quality measurements in this paper have a good practical contribution for practitioners of software development and theory for other researchers. Practically, the measurement results can be used as an evaluation to improve the software of the case study before the software product goes online. Theoretically, procedures and worksheets papers for measuring software quality with three
characteristics, namely: usability, efficiency, and portability can be implemented to complement the procedures and worksheets in previous papers.

This paper is organized as follows: (i) Chapter 1 contains an introduction as we are discussing. (ii) chapter 2: literature review containing work related to previous research-related and a brief description of the quality of the software, (iii) chapter 3 contains the method used, (iv) Chapter 4 discusses the research results. (v) chapter 5 is the conclusion of the study.

2. Literature Review
2.1. Software Quality
The Institute of Electrical and Electronics Engineers defines software quality as the level of a system, component, or process of software that can meet the specifications of its requirements or the needs/wants/expectations of users [11]. Another similar definition states that the level of features or characters of software that has the ability to meet user needs or expectations [12].

Software quality is the existence of the characteristics of a product that are described in terms of its requirements, meaning that it must first be seen what characteristics are related or not to the requirements desired by the user. Therefore, to quantitatively determine the quality of the software, a quality model is needed, which contains the characteristics of quality. The quality model represents an interaction between a group of characteristics and sub-characteristics that serves as the basis for the specification of quality requirements for quality assessments [13]. Quality models are needed to characterize the quality of software products.

ISO 9126-1 is an internationally standardized quality model, which defines the evaluation of software quality [14]. ISO and IEC worked together to produce ISO / IEC 9126-1, then in 2001, it was released which was a refinement from ISO / IEC 9126, which was released in 1991[15]. This standard is currently widely used to access quality software [16].

ISO 9126 identifies six main characteristics of software quality, namely [15] [17]: (i) functionality: the capabilities to cover the functionality of a software product that satisfies user needs, (ii) reliability: the capabilities of the software to maintain the performance level, (iii) usability: capabilities related to the use of software, (iv) efficiency: the capabilities related to the physical resources used when the software is running, (v) maintainability: the capabilities needed to make software changes, and (vi) portability: capabilities related to the ability of software to be delivered to different environments.

3. Related Work
ISO/IEC 9126 has been used popularly to measure software quality. In the previous study has been conducted by measuring the quality of the e-learning application system [18] whose results state that the ISO 9126 standard able be implemented to measure and assess e-learning systems by providing detailed results regarding the strengths and weaknesses of the overall system [18].

The combination of ISO 9126 and the Bayesian network was used by Stefani and Xenos [19] to assist in combining user needs. It could provide reasons for the trouble that happened, and the model was applied to simplify the probability calculation and was useful in making decisions based on errors that appear. Abran et al. [20] proposed a normative model to evaluate the usability of software which was an improvement to the usability character in ISO standards. Survey of various quality models was used to measure the quality of device components [21]. This study used the Analytical Hierarchy Process (AHP) to give weight to the characteristics and sub-characteristics, then the weights were used to assess the overall quality of the components. Alrawashdeh et al. [22] applied ISO 9126 for measuring an ERP software quality. Six quality characteristics in ISO 9126, namely functionality, reliability, usability, efficiency, maintenance, and portability were suggested to be the minimum requirements for an ERP system quality model.

In this study, we apply the ISO 9126 for measuring the AoSCE quality, which it was the author previous research results. This study complements the previous study [7], which tested the characteristics of maintainability, reliability, and functionality while in this study, it assessed the characteristics of usability, efficiency, and portability.

4. Methodology
The research methodology is what stages are carried out in the research so that the research steps become systematic. In this study, the steps taken follow the previous paper [7].

4.1. Collecting data
At this stage, data collection is needed to help to expedite the research. Collecting documents regarding AoSCE is the first stage. The document can be a Software requirement specification, User manual book, and application/software of AoSCE. Then proceed to make the interview protocol. Interviews are conducted with application developers if there are things that have not been stated in these documents, or we need further explanation.

4.2. Planning a test
In this stage, a software test plan (STP) be created basing on the collected documents, and the interviews result with developers in the previous step. The results in this step are used as guidelines for making descriptive testing documents at a later stage.

4.3. Making test descriptions
We create a software test description (STD), in this stage, basing the STP document. At this stage, we must ensure that the questions in the STD are sufficient for all information that be required for the process of software measurement.

4.4. Implementing test
We apply the tests based on the test cases generated in the previous stage. The testing products are documented in the software test report (STR), which is used as a guideline for conducting quality assessments at the next stage. Testing is built upon three characteristics, namely usability, efficiency, and portability.

4.5. Assessing quality
In this stage, we conduct quality assessments of software applications based on data obtained from previous processes. Sometimes the process is also carried out to quantify qualitative data (from interviews). The results of the data quantification are included in the metrics data, then metrics of the sub-characteristics is weighted with applying pairwise comparison.

Likewise, the sub-characteristics of the characteristics are calculated by multiplying the sub-characteristic values by the weights that have been obtained using pairwise comparison. The above steps are repeated for all characteristics, so we get a value for each characteristic. Thus, we get the value of each characteristic as the value of the quality of the software as an object of measurement.

4.6. Generating conclusions and recommendations
In this step, we generate conclusions based on the results of assessing quality. After the assessment, we know how far the software quality for the characteristics studied. Recommendations are made based on the value of each characteristic.

5. Result and Discussion
5.1. The Collected Data
The collected data is recapitulated in a table for every characteristic. Information is required to calculate the metric value of sub-characteristics. Table 1 has contents information which was obtained from data extraction for usability characteristic, while efficiency and portability characteristics are not shown due to the limited space of paper. The sub-characteristics which have metrics are displayed in this table, and for example, understandability has three metrics: completeness of description, demonstration accessibility, and access auditability.
Table 1. The collected data for the characteristic of usability

| Sub-characteristic | Metric                  | Collected Data                                                                 |
|--------------------|-------------------------|-------------------------------------------------------------------------------|
| Understandability  | Completeness of Description | The number of functions explained correctly.                                |
|                    | Demonstration Access       | The total number of functions are available.                                  |
|                    | ability Audit              | The number of successful tutorials accessed.                                 |
|                    |                          | The total number of tutorials are available.                                |
|                    |                          | The number of access types is that have been stored.                        |
|                    |                          | The number of access types needed.                                           |
| Learnability        | Effectiveness of the help system in use | The number of functions can be applied correctly after using the help system. |
|                    |                          | The number of functions available.                                           |
| Operability         | Audit Ability              | Amount of data that can be audited.                                          |
|                    | Uniformity                | The number of all data.                                                      |
|                    |                          | The number of parts comply with the standard.                               |
|                    |                          | The total number of products.                                                |
| Customizability     | Globalization             | The number of functions is capable of obtaining globalization and proven by evaluation. |
|                    |                          | The total number of functions.                                               |

5.2. Sub-characteristic and Metric Weighting Results

After all the data have been obtained, it is continued with weighting on the sub-characteristics. Usability has five sub-characteristics, namely: understandability (under), learnability (learn), operability (opera), attractiveness (attract), and customizability (custom). For weighting, because the sub-characteristics of the customizability with the globalization metric are not relevant to the application software accessed, this sub-characteristic is not included in the weighting.

Weighting for sub-characteristics is by giving a low, medium, and high grades. The three values are changed into a qualitative form on a scale of one to three, namely one for low, two for medium, and three for high. Weighting of the sub-characteristics is structured in a matrix of pairwise comparison by comparing the two sub-characteristics like in this paper [7]. Table 2 shows a matrix of pairwise comparisons for usability characteristics.

Table 2. Pairwise comparison matrix for usability characteristic

| Sub-characteristic | Under | Learn | Opera | Attract |
|--------------------|-------|-------|-------|---------|
| Under              | 1.000 | 1.000 | 2.000 | 3.000   |
| Learn              | 1.000 | 1.000 | 1.000 | 3.000   |
| Opera              | 0.500 | 1.000 | 1.000 | 2.000   |
| Attract            | 0.333 | 0.333 | 0.500 | 1.000   |
| Total              | 2.833 | 3.333 | 4.500 | 9.000   |

Then, normalization was carried out for the matrix in Table 2 by dividing every cell by the total in each column. Furthermore, the horizontal average value is used to obtain the weight of the sub characteristic. Table 3 displays the output of the normalized matrix and average horizontal (line) values for usability characteristics. So, we obtain weights for the usability sub-characteristics, namely: under = 0.358, learn = 0.302, opera = 0.230, and attract = 0.110.
Table 3. Normalized matrix and weighting for usability characteristics

| Sub-characteristic | Under | Learn | Opera | Attract | Avg |
|-------------------|-------|-------|-------|---------|-----|
| Under             | 0.353 | 0.300 | 0.444 | 0.333   | 0.358 |
| Learn             | 0.353 | 0.300 | 0.222 | 0.333   | 0.302 |
| Opera             | 0.176 | 0.300 | 0.222 | 0.222   | 0.230 |
| Attract           | 0.118 | 0.100 | 0.111 | 0.111   | 0.110 |
| Total             | 1.000 | 1.000 | 1.000 | 1.000   | 1.000 |

Using the same way for the metrics on the understandability sub-characteristics consisting of metrics, namely completeness of description, Demonstration accessibility, and Access audit ability, we create the pairwise comparison matrix as shown in Table 4. In the same way, the normalized matrix and weighting for understandability sub-characteristics are generated and presented in Table 5. Meanwhile, other sub-characteristics cannot be shown due to the limited space of this paper.

Table 4. Matrix of pairwise comparison for understandability sub-characteristic

| Metric                      | Completeness of Description | Demonstration Accessibility | Access Audit Ability |
|-----------------------------|-------------------------------|----------------------------|----------------------|
| Completeness of Description | 1.000                         | 1.000                      | 2.000                |
| Demonstration Accessibility | 1.000                         | 1.000                      | 2.000                |
| Access Audit Ability        | 0.500                         | 0.500                      | 1.000                |
| Total                       | 2.500                         | 2.500                      | 5.000                |

Table 5. Normalized matrix and weighting for understandability sub-characteristic

| Metric                      | Completeness of Description | Demonstration Accessibility | Access Audit Ability | Avg |
|-----------------------------|-------------------------------|----------------------------|----------------------|-----|
| Completeness of Description | 0.400                         | 0.400                      | 0.400                | 0.400 |
| Demonstration Accessibility | 0.400                         | 0.400                      | 0.400                | 0.400 |
| Access Audit Ability        | 0.200                         | 0.200                      | 0.200                | 0.200 |
| Total                       | 1.000                         | 1.000                      | 1.000                | 1.000 |

Finally, we get weights for the three characteristics we are discussing, as well as weights for the sub-characteristics for the three characteristics of software quality, the information is presented in Table 6.

Table 6. The weight for three characteristics and sub-characteristics in this study

| Sub-characteristic | Weight | Metric                          | Weight |
|--------------------|--------|--------------------------------|--------|
| Understandability | 0.358  | Completeness of Description     | 0.400  |
|                    |        | Demonstration Accessibility      | 0.400  |
|                    |        | Access Audit Ability            | 0.200  |
| Learnability       | 0.302  | Effectiveness of the help system in use | 0.500  |
| Operability        | 0.230  | Audit Ability                   | 0.500  |
|                    |        | Uniformity                      | 0.500  |
5.3. Quality Metric Assessment Results
Here, we describe the assessment of 21 metrics from three quality characteristics, namely: Usability, which has four sub-characteristics, efficiency, which has two sub-characteristics, and portability which has four sub-characteristics. Table 7 displays the calculations for the metrics of the understandability sub-characteristic as part of the usability characteristics. The table is only an example for calculating metric values due to the limited space in this paper, so we cannot display everything. As shown in Table 6, the understandability sub-characteristics have four metrics, namely: completeness of description, demonstration accessibility, access, and audibility. Table 7 shows how to get the value of each metric and interpretation for the three metrics of the understandability sub-characteristic. Whereas other metrics, they are not displayed due to the limited capacity of the presentation space. However, they are presented in Table 8 at the same time with the results of the sub-characteristics and overall characteristics.

5.4. Software Quality Assessment Results
In the section, we can get the sub-characteristic values if all of the metric values in those sub-characteristics have found in the previous section. Using equation (1) a sub-characteristic value is obtained by the sum of all metric values multiplied by the weight.

\[
SubCharacteristicValue_i = \sum_{j=1}^{n} MetricValue_j \times Weight_j
\]  

Furthermore, characteristic values are obtained using the equation (2) which is the sum of sub-characteristic values multiplied by the weight of the characteristic.

| Metric | Formula | Calculation | Description | Interpretation |
|--------|---------|-------------|-------------|----------------|
| Completeness of Description | \( x = \frac{A}{B} \) | \( x = \frac{54}{54} \) | Based on the data collected from the test results, all functions were described in the product description. | \( 0 \leq x \leq 1 \) |
| | | | The closer to 1 the better |

Table 7. Calculation of metric values for understandability sub-characteristics
Based on the data collected, all tutorials of the functions of the application can be accessed successfully.

The closer to 1 the better

By SRS documents, there are two types of login on the application, i.e. user administrator and regular user. Both have been implemented well.

\[
\text{CharacteristicValue}_i = \sum_{j=1}^{n} \text{SubCharacteristicValue}_j \times \text{Weight}_j
\]

Table 8 shows the outputs of calculations for 21 metrics, 11 sub-characteristics, and three characteristics. We see that the sub-characteristic value is the sum of metrics multiplied by their weight, and the characteristic value is the sum of the sub-characteristic values multiplied by their weight.

Table 8. Calculation results of metric values, sub-characteristics, and characteristics in this study

| No | Metric Name                        | Value | Weight | Value * Weight | Name          | Value | Weight | Value * Weight | Characteristic | Value |
|----|-----------------------------------|-------|--------|----------------|---------------|-------|--------|----------------|----------------|-------|
| 1  | Completeness of Description       | 1     | 0.400  | 0.400          | Understandability | 1.000 | 0.318  | 0.318          | Usability       | 0.876 |
| 2  | Demonstration Accessibility        | 1     | 0.400  | 0.400          | Access Audit Ability | 1.000 | 0.278  | 0.278          |                           |       |
| 3  | Access Audit Ability              | 1     | 0.200  | 0.200          |                           |       |        |                |                |       |
| 4  | Effectiveness of the help system in use | 1 | 0.500  | 0.500          | Learnability            | 1.000 | 0.278  | 0.278          |                |       |
| 5  | Effectiveness of the user documentation | 1 | 0.500  | 0.500          |                            |       |        |                |                |       |
| 6  | Audit Ability                      | 1     | 0.500  | 0.500          | Operability              | 1.000 | 0.206  | 0.206          |                |       |
| 7  | Uniformity                         | 1     | 0.500  | 0.500          |                            |       |        |                |                |       |
| 8  | Simplicity                         | 0.75  | 1.000  | 0.750          | Attractiveness           | 0.750 | 0.099  | 0.074          |                |       |
| 9  | Globalization                      | 0     | 1.000  | 0.000          | Customizability          | 0.000 | 0.099  | 0.000          |                |       |
| 10 | Response Time                      | 1     | 0.333  | 0.333          | Time Behavior            | 1.000 | 0.500  | 0.500          | Efficiency      | 1.000 |
| 11 | Throughput                         | 1     | 0.333  | 0.333          |                            |       |        |                |                |       |
| 12 | Turnaround Time                    | 1     | 0.333  | 0.333          |                            |       |        |                |                |       |
| 13 | I/O Devices Utilization            | 1     | 0.500  | 0.500          | Resource Utilization     | 1.000 | 0.500  | 0.500          |                |       |
| 14 | Memory Utilization                 | 1     | 0.500  | 0.500          |                            |       |        |                |                |       |
| 15 | Hardware Independence              | 1     | 0.351  | 0.351          | Adaptability             | 1.000 | 0.200  | 0.200          | Portability     | 1.000 |
| 16 | Software Independence              | 1     | 0.351  | 0.351          |                            |       |        |                |                |       |
Figure 1 shows the final results of the assessment of the three characteristics, namely, usability, efficiency, and portability. These three characteristics have respective values of 87.6%, 100%, and 100%, which are categorized as well. If we detail the sub-characteristic values, Figure 2 shows the 11 characteristic values of the three characteristics. There are two sub-characteristics whose value is less than 100%, namely Attractiveness (75%) and Customizability (0%).

The attractiveness sub-characteristic has only one metric, namely simplicity. The simplicity metric states the level of product simplicity obtained from the number of simple parts/ functions divided by the total number of functions. From observations on the measured application objects, several parts need to be made simpler, including the menu entering the values of Technical complexity factor (TCF) and Environmental complexity factor (ECF). We suggest to the development team to make the two menus given default values to make it easier for users to enter TCF and ECF values.

The customizability sub-characteristic also only has one metric, namely globalization. The globalization metric states the degree to which a function implementation has the ability to receive the required results in various countries. So far, AoSCE is still intended to be accessed within the scope of Indonesia and has not been designed to be accessed globally; therefore, it needs to be planned to be accessible globally.
6. Conclusion

We have measured the quality of AoSCE application software for three characteristics, namely Usability, Efficiency, and Portability. There are eleven sub-characteristics and twenty-one metrics of the three characteristics. Before providing the assessment, we weighted the sub-characteristics of each characteristic using a pairwise comparison matrix from the Analytical Hierarchy Process (AHP). Weighting using AHP is also carried out for the metrics in each sub-characteristic.

The measurement results of the AoSCE application show that the Usability characteristic has a value of 87.6%, efficiency = 100%, and, portability = 100%. For the Usability characteristics, if detailed into the sub-characteristics, then the Attractiveness and Customizability sub-characteristics have a value of 75% and 0% respectively. We have given recommendations for improvement to make the AoSCE application simpler and can be planned to be globally accessible.

The interesting in this paper is that the procedures and worksheets used to measure and assess the quality of the AoSCE application can be adapted to measure and assess the quality of other software by making adjustments as needed.

7. References

[1] L. Bass, P. Clements, and R. Kazman, *Software Architecture in Practice*. Addison-Wesley Professional, 2003.

[2] A. B. AL-Badareen, M. H. Selamat, M. A. Jabar, J. Din, and S. Turaev, “Software Quality Models: A Comparative Study,” in *Software Engineering and Computer Systems*, vol. 179, J. Mohamad Zain, W. M. bt Wan Mohd, and E. El-Qawasme, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 46–55.

[3] J. Ghayathri and E. M. Priya, “Software Quality Models: A Comparative Study,” vol. 2, no. 1, p. 10, 2013.

[4] J. P. Miguel, D. Mauricio, and G. Rodriguez, “A Review of Software Quality Models for the Evaluation of Software Products,” *Int. J. Softw. Eng. & Appl.*, vol. 5, no. 6, pp. 31–53, Nov. 2014, doi: 10.5121/ijsea.2014.5603.

[5] S. Fahmy, N. Haslinda, W. Roslina, and Z. Fariha, “Evaluating the Quality of Software in e-Book Using the ISO 9126 Model,” *Int. J. Control Autom.*, vol. 5, no. 2, p. 8, 2012.

[6] ISO, “ISO/IEC 25010:2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models,” 2011. [Online]. Available: https://www.iso.org/standard/35733.html.

[7] Sholiq, D. A. Dewangga, A. P. Subriadi, and F. A. Muqtadiroh, “Quality measurement of software based on characteristics of functionality, reliability, and maintainability,” 2018 Int. Semin. Res. Inf. Technol. Intell. Syst. ISRITI 2018, pp. 192–197, 2018, doi: 10.1109/ISRITI.2018.8864449.

[8] M. U. Rehman, S. I. Majumdar, E. Vocabulary, and D. Base, “An Analytical and Comparative Study of Software Usability Quality Factors,” no. 1978, pp. 800–803, 2016.

[9] P. Sudhaman and C. Thangavel, “Efficiency analysis of ERP projects-software quality perspective,” *Int. J. Proj. Manag.*, vol. 33, no. 4, pp. 961–970, 2015, doi: 10.1016/j.ijproman.2014.10.011.

[10] D. V. Silakov and A. V. Khoroshilov, “Ensuring portability of software,” *Program. Comput. Softw.*, vol. 37, no. 1, pp. 41–47, 2011, doi: 10.1134/S0361768811010051.

[11] I. S. C. Committee and others, “IEEE Standard Glossary of Software Engineering Terminology (IEEE Std 610.12-1990). Los Alamitos,” *CA IEEE Comput. Soc.*, vol. 169, 1990.

[12] R. Fitzpatrick, “Software quality: definitions and strategic issues,” *Softw. Qual.*, 1996.

[13] K. Khosravi, “On Issues with Software Quality Models,” in *Proceedings of 9th ECOOP workshop on Quantitative Approaches in Object-Oriented Software Engineering*, 2005, pp. 70–83.

[14] H. Al-Kilidar, K. Cox, and B. Kitchenham, “The use and usefulness of the ISO/IEC 9126 quality standard,” in *2005 International Symposium on Empirical Software Engineering*, 2005., 2005, pp. 7--pp.

[15] F. Coallier, “Software engineering-Product quality-Part 1: Quality model,” *Int. Organ. Stand. Geneva, Switz.*, 2001.

[16] M. F. Bertoa and A. Vallecillo, “Quality Attributes for Software Metamodels,” p. 14, 2010.
[17] B. Zeiss, D. Vega, I. Schieferdecker, H. Neukirchen, and J. Grabowski, “Applying the ISO 9126 quality model to test specifications--exemplified for TTCN-3 test specifications,” Softw. Eng. 2007--Fachtagung des GI-Fachbereichs Softwarentwicklung, 2007.

[18] B. B. Chua and L. E. Dyson, “Applying the ISO 9126 model to the evaluation of an e-learning system,” 2004, pp. 184–190.

[19] A. Stefani and M. Xenos, “E-commerce system quality assessment using a model based on ISO 9126 and Belief Networks,” Softw. Qual. J., vol. 16, no. 1, pp. 107–129, Mar. 2008, doi: 10.1007/s11219-007-9032-5.

[20] A. Abran, A. Khelifi, and W. Suryn, “Usability Meanings and Interpretations in ISO Standards,” p. 14, 2003.

[21] A. Sharma, R. Kumar, and P. S. Grover, “Estimation of quality for software components: an empirical approach,” ACM SIGSOFT Softw. Eng. Notes, vol. 33, no. 6, p. 1, Oct. 2008, doi: 10.1145/1449603.1449613.

[22] T. A. Alrawashdeh, M. Muhairat, and A. Althunibat, “Evaluating the Quality of Software in ERP Systems Using the ISO 9126 Model,” Int. J. Ambient Syst. Appl., vol. 1, no. 1, pp. 1–9, 2013.

Acknowledgement
We are pleased to thank the Direktorat Riset dan Pengabdian Masyarakat (DRPM) of Institut Teknologi Sepuluh Nopember and DRPM of Dirjen Pendidikan Tinggi to fund this study.