International Journal of Electrical and Computer Engineering (IJECE)
Vol. 6, No. 3, June 2016, pp. 1038 ~ 1047
ISSN: 2088-8708, DOI: 10.11591/ijece.v6i3.9836

Academic Cloud ERP Quality Assessment Model

Kridanto Surendro, Olivia
Information Systems & Technology, Institute of Technology Bandung, Indonesia

Article Info

Article history:

Received Dec 30, 2015
Revised Mar 16, 2016
Accepted Apr 1, 2016

Keyword:

Assessment model
Cloud computing
Enterprise resource planning
Software quality

ABSTRACT

In the past few decades, educational institutions have been using conventional academic ERP system to integrate and optimize their business process. In this delivery model, each educational institutions are responsible of their own data, installation, and also maintenance. For some institutions, it might cause not only waste of resources, but also problems in management and financial aspects. Cloud-based Academic ERP, a SaaS-based ERP system, begin to come as a solution with is virtualization technology. It allows institutions to use only the needed ERP resources, without any specific installation, integration, or maintenance needs. As the implementation of Cloud ERP increases, problems arise on how to evaluate this system. Current evaluation approaches are either only evaluating the cloud computing aspects or only evaluating the software quality aspects. This paper proposes an assessment model for Cloud ERP system, considering both software quality characteristics and cloud computing attributes to help strategic decision makers evaluate academic Cloud ERP system.

Corresponding Author:

Kridanto Surendro,
Information Systems & Technology,
Institute of Technology Bandung,
Jalan Ganesa 10, Bandung 40132, Indonesia.
Email: endro@informatika.org

1. INTRODUCTION

Academic enterprise resource planning (ERP) systems are meant to integrate the separate business process and functions within an educational institution to help streamlining the process and to provide real time, on demand information needs [1]. However, as the business continues to grow, more and more resources need to be managed in the system. In educational institutions, there are increasing numbers of students, departments, and other academic data each year. This continuous growth requires continuous scaling and improvement of the academic ERP system. Cloud computing technology can meet several of the advance and futuristic requirements of ERP implementations in higher educational institutions and can cater to increasing volume and range of services [2]. Therefore, in order to adapt to this continuous growth, academic ERP system begins to be constructed based on cloud computing platform.

Cloud computing, through its Software-as-a-Service (SaaS) service model, enables consumer to use applications remotely, without any specific installation or integration needed. The applications are being run on a cloud provider's infrastructure. These applications can be used by consumers from various client devices, such as web browser, or a program interface. The consumer doesn't need to manage or control the underlying infrastructure by themselves. SaaS applications usually allows only limited user-specific application configuration settings [3]. Mone [4] proposed a multy-user security policies to access data, while Radha [5] proposed a service level agreements of cloud computing.

ERP system that is implemented as a software-as-a-service has been mentioned in various names, from service-based ERP, ERP III, to cloud ERP [6]. In this paper, it is referred as cloud ERP. Cloud ERP is a flexible, yet powerful information system incorporated web-based SOA and cloud computing version, which enables virtual enterprises to offer increasing degrees of flexibility, agility and dynamic amorphousness [7].

This ERP system applies cloud computing characteristics. Its on-demand-service characteristic allows ERP implementation to be more flexible, adaptable, and scalable. Its resource pooling and measured service characteristic allows ERP implementation to be more efficient and affordable [8].

Many organizations are shifting or starting to shift ERP deployment, from on-premise to on-cloud. Research shows that the main reasons are because ERP implementation is becoming more complex and more challenging to manage, and organizations only have limited resources for the implementation. They don't want to spend much on buying the servers and hire the IT resources. They are looking for easier and more economical ERP implementation [9]. Cloud ERP is the answer of his problem. It is cost-saving, and easy to use. But, as the implementation of cloud ERP increases, including the ones for educational institutions, problem arises on how to evaluate the system.

Recent research [10],[11] shows that there are a few challenges on cloud ERP implementation. One of the primary challenge is on monitoring, analysis, and building trust [10]. ERP is a very large and critical application, and its outsourcing requires constant evaluation, monitoring, and reporting. It is therefore, a model to assess and evaluate academic cloud ERP system is needed.

This paper proposes a model to assess and evaluate academic cloud ERP system. Initial model is based on previous research [12],[13] on ERP quantitative assessment model. The main difference is in this research paper, the model is more specialized for software quality aspects of academic ERP system and cloud computing characteristics, especially for SaaS. Based on studies on these areas, the assessment model is developed and proposed.

The main objective of this paper is to provide a method for strategic decision makers (with no technical background) to evaluate an academic cloud ERP system. Hopefully, this paper could help educational institutions to measure the quality of their academic cloud ERP system. With proper evaluation and measurement, educational institutions can take actions to improve their system and their business.

The paper is organized as follows, in section 2 the academic cloud ERP system is defined and explored, Software-as-a-Service quality assessment criterias are discussed in section 3. The combination and proposed academic cloud ERP assessment model is described in section 4, model implementation is shown in section 5, and finally the result is concluded in section 6.

2. ACADEMIC CLOUD ERP

2.1. Academic Cloud ERP Architecture

To be able to deliver its functionalities, academic cloud ERP system is supported by a client/server architecture. The commonly used model for cloud ERP is the three-tier client/server architecture. This architecture consists of 3 layers: application/presentation layer, processing layer, and execution/storage layer. The first layer is located on the device interface on the client's side, whereas the rest two layers are located in the cloud server [9],[10].

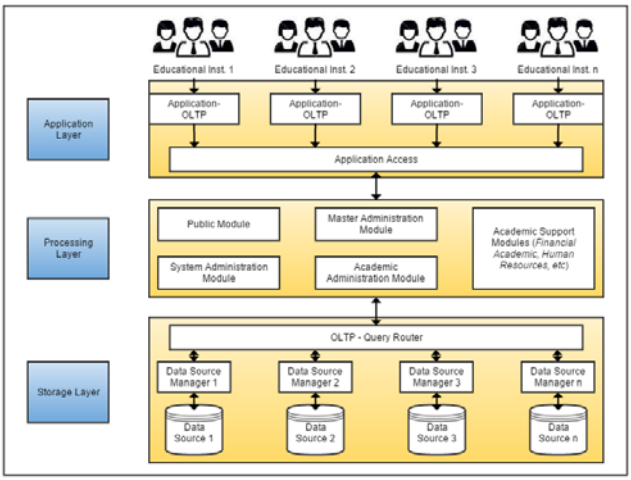

Figure 1. Academic Cloud ERP Architecture

Application layer manages user interface, so the users can conduct various transactions with the system. Components of this layer is used to set data format and interface for the users and handle user inputs

and response from the processing layer. This layer also manages user access to ensure security within the academic cloud ERP system [10].

Processing layer manages the functional modules of academic cloud ERP system. Inputs from application layer are processed using the related data from storage layer. Components of this layer act as the brain of the system, controlling data and functions to be used by related users. Processing layer also includes functions for policy and security, authentication, and virtualization control [10].

Storage layer consists of components handling orders for OLTP data input and storage in physical storage devices. Data storage is managed using a relational database which is used to store master and transactional data. To maximize the process efficiency of the academic cloud ERP system, usually similar database model is used for every module in the system.

2.2. Academic Cloud ERP Requirements

Similar with any other system, academic cloud ERP system has its own requirements. These requirements have to be met in order for the system to work. Academic cloud ERP system has to comply with both academic ERP requirements and cloud ERP attributes.

Sabau, et al [14] defined requirements for academic ERP system. An integrated university information system must provide:

- **Integration**. Academic ERP system must provide data that is integrated enterprise-wide. This integration is needed to provide complete coverage of all needs and availability of information and avoid data inconsistency so the processes in the institutions could be done well.
- **Flexibility**. Operations in academic enterprises nowadays is very complex. Changes are done regularly and frequently, varies from the academic sites (curriculum, courses, etc.) or regulatory sites (academic policy). Therefore, the ERP system should be flexible for the changes to be applied in the system so it can match the evolution of organization.
- **Support in decision making**. The academic ERP system must provide accurate and good materials to support the strategic analysis and decision making in institutional governance process. Data validity, reliability and availability is the core requirements to support this governance process.
- **Service evolution**. One of the main purpose for implementing ERP in the academic institutions is to provide better service for the institution's academic community. Therefore, the service in the ERP should be able to be improved for time to time, e.g. could be accessed from different devices.

Shao [15] defined attributes for multi-tenant architecture, which is used for the academic cloud ERP system. It has to comply with 4 attributes:

- **Resource isolation**. The resource separation between tenants is done in a fair manner so each tenant get the fair infrastructure and software services. It prevents improper service usage from one tenant that can interfere with the service for other tenants.
- **Customization**. Cloud applications usually need customization to fit with customer’s needs. However, in cloud ERP, complex customization is not recommended as it can lead to slow processing. Additional computation will be needed at runtime, and adds complexity to the database as individual customization needs to be stored in addition to various data.
- **Security**. In cloud ERP, tenants share their data storage and other code processing. This sharing schema might rise some significant security risks. Data access protection is one of the most vital component in cloud ERP security.
- **Scalability**. Application development on cloud ERP might be more limited than the on-premise ERP. Cloud service providers have to have multiple copies of the same software that can be dynamically created to provide services, in order to maintain the system scalability.

The aspects above must be fulfilled so the academic cloud ERP system can provide the expected benefits and advantages for the academic institutions.

3. SOFTWARE-AS-A-SERVICE QUALITY ASSESSMENT

The quality of a Software-as-a-Service (SaaS) application is related to its product quality and service quality. Product quality is related to the functionalities of the application itself. On the other hand, service quality is related to the rate of compliance and services provided by the cloud computing providers. These two types of quality need to be met by academic cloud ERP system, as this system requires not only academic ERP functional capabilities, but also benefits of cloud computing services.

3.1. Software Quality Measurement

Software product quality could be measured with several frameworks [16]-[18]. One of the most simple, yet one of the most powerful one is the software quality measurement framework from Wagner. Wagner [19] proposed the steps for developing software quality model, evaluation, and measurement.

1. Software Quality Planning: These are the steps required to develop a software quality model.

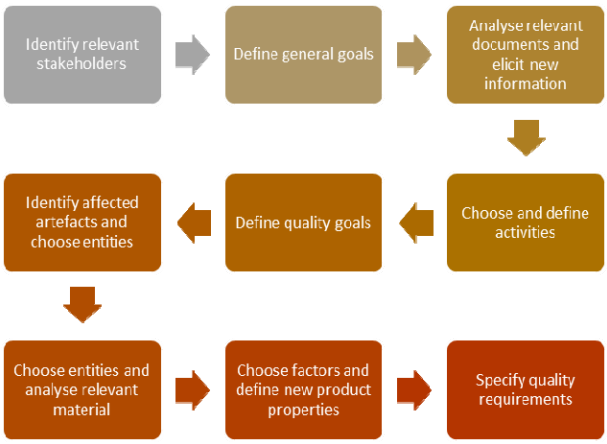

Figure 2. Software Quality Planning Steps [19]

The first step is to identify and list all stakeholders related to the system/software. Then, define general goal for each stakeholder. The goals defined can be refined by analyzing relevant documents to get more information. After that, list activities between stakeholder and the system that is needed to reach the goals. Quality goals can be defined by analyzing activities, prioritizing, and deciding on how deep is each activity going to be supported by the system. The next step is to identify artefacts related to each quality goals, the impact from the system, and choose the entities to be evaluated from each artefacts. Material analysis can be used to get product factor from each entities. Defined product factors are checked again and added or deducted if needed. Finally, quality requirements can be specified [19].

2. Software Quality Evaluation and Measurement

After the quality requirements are defined, the next step to do is to evaluate and measure system's fulfillment of each software quality requirement. The required steps are shown in Figure 3.

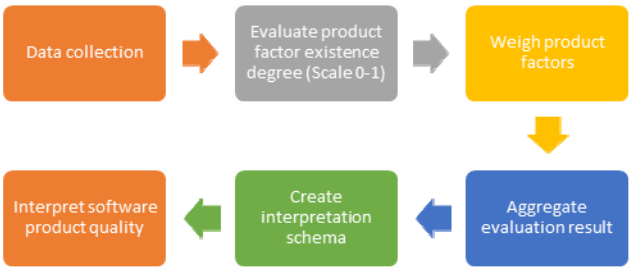

Figure 3. Software Quality Evaluation and Measurement [19]

Data is collected for each product factor from quality planning. Then, the degree of existence for each product factor is evaluated from the data collected. Each product factor is given a weight number, based on its impact on the product quality. Then, the evaluation result is aggregated with the weight, so we can get the quality measurement. After that, we need to develop an interpretation schema, in order to be able to interpret the measurement result, is it a good or bad number [19].

3.2. Software-as-a-Service Evaluation

Following the increasing use of SaaS, the quality evaluation methods are also arising. Wen, et al [20] proposed a model to evaluate SaaS quality. The proposed model is based on SaaS Maturity Model from Forrester and SaaS quality model, which covers security measures, Quality-of-Service (QoS) measures, and software quality measures.

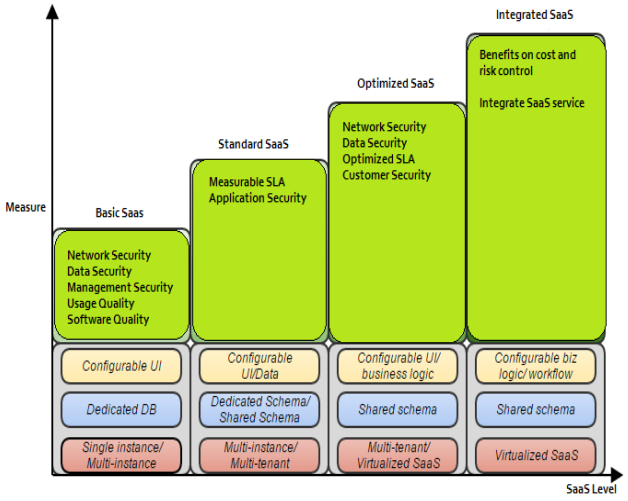

Figure 4. SaaS Quality Evaluation [20]

There are four levels on the evaluating model. These four levels are Basic SaaS, Standard SaaS, Optimized SaaS, and Integrated SaaS. SaaS that is categorized as Basic SaaS fulfills software quality criterias, SaaS basic features (multi-tenant), and configurable UI, dedicated database, single/multi instance technical architecture. Standard SaaS fulfills all Basic SaaS criterias, with measured Service Level Agreement (SLA) and standardized application security. Optimized SaaS fulfills all Standard SaaS criterias, added with standardized network, user, and data security, also optimized SLA. Integrated SaaS fulfills all Optimized SaaS criterias with Quality of Service (QoS) on cost-saving and risk control [20].

Bauer, et al [21] gathered key quality indicators (KQI), subset of key performance indicators (KPI) across the customer facing service boundary characterize key aspects of the customer's experience and perception of quality. For cloud-based application, KQIs can cover the characteristics below.

- Service Availability: The service is online and available to users.
- Service Latency: The service promptly responds to user requests.
- Service Reliability: The service correctly responds to user requests.
- Service Accessibility: The probability that an individual user can promptly access the service or resource that they desire.
- Service Retainability: The probability that a service session will continuously be rendered with good service quality until normal termination of that session.
- Service Throughput: Meeting service throughput commitments to customers.
- Service Timestamp Accuracy: Meeting billing or regulatory compliance accuracy requirements.

KQIs usually cover high-level business considerations, including service qualities that impact user satisfaction and churn [21]. The KQIs can be adjusted based on the application's environment and customer's focus.

4. ASSESSMENT MODEL

The academic cloud ERP assessment model is developed based on software quality measurement framework and SaaS quality evaluation model, in accordance with academic cloud ERP requirements. Software quality measurement covers the product aspects of academic cloud ERP system, whereas SaaS quality evaluation covers the service aspects of the system. If the academic cloud ERP system scores well in both criterias, we can highly expect the system is running very well.

4.1. Academic Cloud ERP Application Quality Assessment

Application quality assessment can be done based on software product factors. Product factors can be defined from stakeholder goals and activities. From stakeholders' activities to achieve goals, we can find the related artefacts to be examined. Each artefact has its own aspect that can be assessed, which called product factor. The table below provides product factors for academic cloud ERP application quality assessment.

The academic cloud ERP application quality model consists of 5 major criterias to be evaluated. Those criterias are functionalities, user-friendliness, security, business support, and stability. In total, they make up 15 product factors to be evaluated. These factors are derived from stakeholders' goals, activities, and artefacts. Each factor has one or more measure to be checked in the assessment of each factor. The weighing of those measures is based on how crucial those aspects are in affecting the performance of academic cloud ERP system.

Table 1. Application Quality

Product Factor	Measure	Weight (1-10)
1. Functionalities		
Accessibility	Existence	10
Completeness	Functional implementation completeness	10
	Functional implementation coverage	10
Correctness	Computational Accuracy	10
	Precision	10
	Accuracy to expectation	10
Appropriateness	Functional adequacy	10
	Functional specification stability (volatility)	5
2. **User-friendliness**		
Time-behavior	Response time	10
	Throughput	7
	Turnaround time	5
UI aestheticness	Message Clarity	10
	Interface element clarity	10
	Attractive interaction	6
	Porting user friendliness	5
3. Security		
Concealment	Data encryption	10
Limitedness	Session control	10
Authentication	Access auditability	7
	Access controllability	10
Guardiness	Data corruption prevention	10
4. Business support		
Dependency	Data-coupled module ratio	10
	Autonomy	10
Cohesion	Unified data reference	10
Analyzability	Activity recording	10
	Diagnostic function support	8
5. Stability		
Operability	Operational consistency	10
	Error correction	8
	Operational error recoverability in use	5
	Customizability	5
Reliability	Fault detection	10
	Test adequacy	8
	Incorrect operation avoidance	8
	Failure density against test cases	10

Functionalities, business support, and security has most high weighing. These three criterias have the most impact on system performance and quality. Functionalities are related to the functional modules of the system. Without proper functionalities, system cannot deliver the expected outcomes. Business support includes modularity and analyzability which contributes to system complexity and troubleshooting activities. Security also plays a big role in academic ERP performance, as this system is used by a big number of users and stores a large amount of enterprise data. A single security flaw could lead to massive performance delay of the academic cloud ERP system.

These factors cover the academic ERP requirements. Functionalities and stability criteria covers the integration requirements. Business support criteria covers flexibility and support in decision making requirements. User-friendliness and security covers the service evolution requirements.

4.2. Academic Cloud ERP Service Quality Assessment

Service quality assessment can be done based on cloud computing service aspects. These aspects includes SaaS features and architecture. The table below provides service aspects for academic cloud ERP service quality assessment.

There are three main aspects need to be considered on academic cloud ERP service quality assessment model. Those criterias are technical architecture, quality of service, and security. There are several measures that can be used to evaluate the aspects. Technical architecture covers the multi-tenancy degree, data isolation level (dedicated/shared), application isolation level (single instance/multi instance), and configurability (on UI, business logic, workflow). Quality of Service covers the scalability, availability, reliability (process usability, effectiveness and efficiency), Service Level Agreement (existed/managed/optimized), and interoperability (multi-device-use) of the system. Security covers aspects ensuring security on customer (SLA and risk management), application (authentication and access control, prevention on common security threat, private data encryption), network (ensure successful data transmission), data (confidentiality, integrity, accessibility, reliability), and management (security management system).

Table 2. Service Quality

Service Aspects	Measure	Weight (1-10)
Technical architecture	Multi-tenancy	10
	Data isolation level	7
	Application isolation level	7
	Configurability	5
Quality of Service	Scalability	10
	Service availability	10
	Service reliability	10
	Service Level Agreement	7
	Interoperability	7
Security	Customer security	7
	Application security	10
	Network security	10
	Data security	10
	Management security	5

Quality of Service and security has higher weighing amount. These two criterias are impacting directly on how application service is perceived by the customers. Customers using cloud computing services are expecting to get all the benefits of its technology, which mainly are scalability, interoperability, easiness to use, and cost-saving (which are stated in Quality of Service). Without these benefits, academic cloud ERP system has no additional point from conventional academic ERP system. Security is also very important in academic cloud ERP system, as the system is used by multiple tenants at the same time, and the management of the system is outsourced with the cloud service providers.

Technical architecture is also important, although it is not directly affecting the service. Technical architecture of the system affects scalability and usability of the system. The system's technical architecture have to fulfill the cloud ERP architecture attributes, which are resource isolation, customization, security, and scalability. In this model, resource isolation is covered by data and application isolation measures. Customization is covered by configurability measures. Security is covered on security service aspects, and scalability is covered on Quality of Service aspects.

4.3. Assessment Value

Each measures on every aspects need to be evaluated on the system. The evaluation can be done by performing system testing. From the test results, one can define the degree of existence for each measure from the scale 0-1, with 0 the lowest degree (non-existent) and 1 the highest degree (fully performing). Every degree measurement is then aggregated using the weighing. It is multiplied with the associated weight, and then averaged to get the quality assessment value.

Provided below is the mathematic equation for the proposed model. Let z be the score value of a product factor, n be the number of measures of a product factor (for application quality) or service aspects (for product quality), w be the weight of the corresponding measure, and x be the score value of the system related to the corresponding measure.

$$z = \frac{\sum_{i=1}^{n} wi.xi}{\sum_{i=1}^{n} wi}$$

The final score value for the application quality / service quality can be determined by finding the average of all z values.

$$Final\ application\ quality\ score\ value = \frac{\sum_{i=1}^{15} zi}{15}$$

$$Final\ service\ quality\ score\ value = \frac{\sum_{i=1}^{3} zi}{3}$$

The academic cloud ERP quality assessment value should be a number in 0-1 range, both for application quality and service quality.

5. RESULTS AND ANALYSIS

In this research, the proposed model was implemented to evaluate the quality of an Indonesian academic cloud ERP system, which is Cloud SisfoKampus [13]. The results of the quality assessment using this model are as follows.

5.1. Application Quality Assessment

Following the proposed model, application quality assessment was done by examining quality requirements of the system in relation to its product factors and measures. These are a few examples of the product quality requirements, the related assessment criterias, and the measurement value results.

Table 3. Application quality assessment Example

Quality Requirement	Product Factor	Measure	Value
There is a HTML page for each module of the system	accessibility	existence	1
Organization data is stored based on needs to operate the system	appropriateness	Functional adequacy	1
		Functional specification stability (volatility)	0.5
Organization data is stored correctly based on user input	Correctness	Computational Accuracy	1
		Precision	1
		Accuracy to expectation	1
Private user data is stored accordingly	Concealment	Data encryption	1

Assessment values are determined by testing. For example for the first quality requirement, to measure existence, there are sets of test cases to check HTML page existence for all 12 modules of the system. The system received value score 1 due to 48 HTML pages existence for all 12 modules of the system. Functional adequacy values can be determined by testing the number of functions that are suitable for the system from all related functions. Functional specification stability can be scored by testing the number of functions that are unchanged from the requirements to implementation and maintenance. This system received score 0.5 due to 20 out of 40 data functions changed on the way to implementation.

All the defined score values are then aggregated based on the weighing defined in the proposed model. They are first aggregated per quality requirements, then averaged for all quality requirements. Following this assessment model, the academic cloud ERP system received the score of 0.978 out of maximum score 1.

5.2. Service Quality Assessment

Similar to the product quality assessment, service quality assessment values can be determined by performing test to the specified service aspects and measures. These are a few examples of the service assessment criterias, and the measurement value results for Cloud SisfoKampus [13].

Table 4. Service quality assessment Example

Service Aspects	Measure	Value
Quality of Service	Scalability	1
	Service availability	0.814
	Service reliability	0.915
	Service Level Agreement	0
	Interoperability	0.25
Security	Customer security	0
	Application security	0
	Network security	1
	Data security	1
	Management security	1

Values for technical architecture and security service aspects can be done using system test. On the other hand, score values for quality of service aspects need to be defined using user acceptance test. For example, the availability of the system is tested by checking up time from access by several users. The interoperability is tested by user accessing the system from various devices (desktop and mobile). The system is not running very well on mobile device, hence the interoperability score is low. The customer and application security aspects haven't been taken care properly because the system is running on external platform (not managed by the system).

After all the score values defined, these values are aggregated using the designated weighing from the model, and then averaged. The service quality score for Cloud SisfoKampus is 0.752 out of maximum score 1.

6. CONCLUSION

The use of academic ERP system in educational institution is very vital. However, some educational institutions cannot implement the system because of lack in budget or resources, and other educational institutions need more scalable academic ERP system to adapt with their evolving business. Academic cloud ERP system enables these features and quickly become a solution. And as the implementation of Cloud ERP increases, problems arise on how to evaluate this system. Therefore, this paper provides the assessment model for academic cloud ERP quality evaluation, to support project managers in making decision on implementing cloud-based academic ERP system.

The assessment model is developed based on academic cloud ERP requirements, software product quality measurement, and Software-as-a-Service evaluation. The assessment aspects are focused on the characteristics of academic cloud ERP system as a product and as a service. The weighing for these aspects are determined on how crucial these aspects are in affecting ERP performance.

The three most important aspects to be considered on product quality sides are functionalities, business support, and security. Other aspects from product quality measurement includes user-friendliness and stability. On the service quality sides, the main aspects to be considered are Quality of Service and also security. The other aspect need to be considered is the technical architecture of the system. Evaluation on these aspects can be done by performing system testing and user acceptance testing. From the evaluation and weighing aggregation, quality assessment value can be determined. The final results will be 2 assessment values, one for software quality, and another for service quality. These assessment value ranges from 0-1, with 0 means the academic cloud ERP system is not performing, and 1 means the academic cloud ERP system has the best performance.

ACKNOWLEDGEMENTS

This research was supported by the Research Grants Institute of Technology Bandung, Indonesia (No 0265d/I1.C07/PL/2015).

REFERENCES

[1] C. Murphy, "ERP: The Once and Future King of Campus Computing," in *Campus Technology, Syllabus Media Group*, 2004. http://campustechnology.com/articles/2004/01/erp-the-once-and-future-king-of-campus-computing.aspx

[2] S. Goel, *et al.*, "Impact of Cloud Computing on ERP implementations in Higher Education," *International Journal of Advanced Computer Science and Applications*, vol/issue: 2(6), pp. 146-148, 2011.

[3] National Institute of Standards and Technology (NIST), "NIST Definition of Cloud Computing: Recommendations of the National Institute of Standards and Technology," U.S. Department of Commerce, 2011.

[4] S. P. Mone and S. S. Dhotre, "Enforcing Multi-user Security Policies in Cloud Computing," *International Journal of Electrical and Computer Engineering (IJECE)*, vol/issue: 3(4), pp. 504~508, 2013.
[5] K. Radha, *et al.*, "Service Level Agreements in Cloud Computing and Big Data," *International Journal of Electrical and Computer Engineering (IJECE)*, vol/issue: 5(1), pp. 158~165, 2015.
[6] I. Saeed, *et al.*, "Cloud Enterprise Resource Planning Adoption: Motives & Barriers," *CONFENIS*, pp. 99-122, 2011.
[7] Y. Wan, and B. Clegg, "Enterprise Management and ERP Development: Case Study of Zoomlion Using the Dynamic Enterprise Reference Grid," *Communications in Computer and Information Science*, vol/issue: 109(3), 191-198, 2010.
[8] G. Raihana and F. Haseen, "Cloud ERP - A Solution Model," *International Journal of Computer Science and Information Technology & Security*, vol/issue: 2(1), pp. 76-79, 2012.
[9] G. N. Purohit, *et al.*, "Challenges Involved in Implementation of ERP on Demand Solution: Cloud Computing," *International Journal of Computer Science*, vol/issue: 9(4), pp. 481-489, 2012.
[10] A. Chaudhary, "Database Architecture of OLTP in the SaaS-based Multi-tenant Educational Enterprise Resource Planning (Java Enterprise Edition (JEE) Based Engineering approach)," *Proceedings of 2012 International Conference on Education and e-Learning Innovations*, 2012. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber= 6360636.
[11] S. Roy, *et al.*, "A cognitive Approach for Evaluating the Usability of Storage as a Service in Cloud Computing Environment," *International Journal of Electrical and Computer Engineering (IJECE)*, vol/issue: 6(2), pp. 759~769, 2016.
[12] Olivia and K. Surendro, "Generic Quantitative Assessment Model for Enterprise Resource Planning (ERP) System," *ICSIIT - Intelligence in the Era of Big Data. Bali*, vol. 4, pp. 563-571, 2015.
[13] Olivia, "Cloud Computing Implementation for Academic Enterprise Resource Planning (ERP) System in Higher Educational Institutions in Indonesia," *Undergraduate thesis (in Bahasa). Bandung: Institute of Technology Bandung*, 2015.
[14] G. Sabau, *et al.*, "An Evaluation Framework for Higher Education ERP System," *WSEAS TRANSACTIONS on COMPUTERS*, vol/issue: 8(11), pp. 1790-1799, 2009.
[15] Q. Shao, "Towards effective and intelligent multi-tenancy SaaS," *PHD thesis*, Arizona State University, 2011.
[16] G. O'Regan, "A Practical Approach to Software Quality," *Springer*, 2002.
[17] W. Suryn, "Software Quality Engineering: A Practitioner's Approach," *Wiley*, 2014.
[18] C. R. Pandian, "Software Metrics: A Guide to Planning, Analysis, and Application," *CRC Press LCC*, 2004.
[19] S. Wagner, "Software Product Quality Control," *Springer*, 2013.
[20] P. X. Wen and L. Dong, "Quality Model for Evaluating SaaS Service," *Fourth International Conference on Emerging Intelligent Data and Web Technologies. Fukuoka*, 2013.
[21] E. Bauer and R. Adams, "Service Quality of Cloud-Based Applications," *IEEE Press and Wiley*, 2014.

BIOGRAPHIES OF AUTHORS

Kridanto Surendro is a lecture in the School of Electrical Engineering and Informatics – Institute of Technology Bandung, Indonesia. . He graduated from doctoral degree in Computer Science at Keio University, Japan. His research area includes Cloud Computing, Information Systems Quality, Information Governance, and IT Service Management.

Olivia graduated from bachelor degree in Information Systems & Technology, School of Electrical Engineering & Informatics - Institute of Technology Bandung, Indonesia in 2015. Her research area includes: Cloud Quality Assessment, Information Systems, and Software Engineering.