Updated DeLone and McLean IS Success Model and Commercial Open Source Software (COSS) Company Success

Shimels Diriba Garomssa, Rathimala Kannan and Ian Chai
Multimedia University, Malaysia, {1181402786@student.mmu.edu.my,rathimala.kannan@mmu.edu.my, ianchai@mmu.edu.my}

ABSTRACT

Commercial Open Source Software (COSS) is a promising business model as it represents the middle ground between expensive proprietary software and free software. The unique nature of COSS companies has captured researchers' attention; hence several studies have been conducted to assess the success of a few prominent COSS companies. However, comprehensive empirical study consisting of various COSS companies of different size, type, and prominence is lacking. Hence, the aim of this study is to evaluate the success of diverse COSS companies by adapting the DeLone and McLean Updated Information Systems (IS) Success Model. The result indicates that COSS companies' success is significantly influenced by user satisfaction while the impact of software use on COSS company success is insignificant. Moreover, both software quality and product property positively impact user satisfaction as well as software use.

Keywords: Commercial Open Source Software, company success, DeLone & McLean Model.

I INTRODUCTION

Software can generally be classified as proprietary software, open source software (OSS), and free software (Anand, Tiwari, Krishna, & Sharma, 2018). OSS in turn is divided into two categories of software: community open source software and commercial open source software (COSS) (Shahrivar, Elahi, Hassanzadeh, & Montazer, 2018; Riehle, 2009). Community OSS is a cost-free software offered by foundations or individuals within the OSS-community. In contrast, COSS companies use two distinct strategies. The first strategy is a dual-licensing strategy where the same product is licensed under open source license and commercial license. The second strategy is the open core model which entails open sourcing the core, but closing off the extensions or additional features (Riehle, 2020).

The COSS industry consists of several big and small companies. The top tier alone consists of 42 companies with an estimated market value of 150 billion USD in 2019 (Riehle, 2019). This has attracted multibillion-dollar proprietary software companies such as Facebook, Microsoft and IBM, whose heavy investment in OSS is highly threatening the survival of COSS companies (Daniel, Midha, Bhattacherhjee, & Singh, 2018).

An in-depth review of literature revealed that a COSS companies' success model is absent (Kamal, Mahoto, & Memon, 2018; Amrollahi, Khansari, & Manian, 2014) indicating the existing knowledge gap. A lack of standard measurements to assess the success of COSS companies is also evident (Mäntylä, Jørgensen, Ralph, & Erdogmus, 2017). Finally, a simple search on google scholar showed that there are around 4 million publications on and around Red Hat, MySQL, and Linux. However, there appear to be no comprehensive studies involving groups of companies. Hence, these research gaps motivated this study to investigate the critical factors that contribute to the success of COSS companies.

The current study adapts the widely applied and validated DeLone and McLean Updated Information Systems (IS) Success Model to the context of COSS companies. An expert-validated tool for assessment of COSS companies' success is also used. Finally, data is collected from 49 COSS companies of varying sizes, services and products. Thus, the purpose of the study is to evaluate the success of COSS companies. The rest of the paper is organized as follows: related literature, research design, discussion, implication, and conclusion.

II RELATED LITERATURE

A. COSS Companies

Commercial open source software (COSS) is controlled by a single company, who is responsible for the development of some or all of the source code for the software which it licenses under an open source and commercial version. Additionally, COSS companies generate revenue through complementary products or services (Riehle, 2020).

The societal benefits of OSS development methodology are widely popular among commercial companies, developers, sponsors, and OSS communities. Some of the advantages of OSS development methodology include the ability to utilize the support of the OSS community in providing technical support, test products before release, promotion, and distribution (Riehle, 2009; Deodhar, Saxena, Gupta, & Ruohonen, 2012). These advantages, in turn, translate into low development cost, lower distribution cost, higher quality, lower price offer for the customer, and social welfare (August, Shin, & Tunca, 2017). Thus, COSS companies can offer software products at an affordable price to individuals, SMEs, civic societies, and educational institutions.

In recognition of these benefits, several COSS companies have been able to hybridize closed source alongside open source development strategies. They have been able to successfully develop complex systems, On the other hand, many others seem to suffer from lack of clarity and pay the ultimate price (Lokhman, Mikkonen, Hammouda, Kazman, & Chen, 2013). The enormous success enjoyed by pioneers such as Linux, Mozilla, and MySQL have attracted researchers' attention (Lokhman, Mikkonen, Hammouda, Kazman, & Chen, 2013). Between 2002 and 2017 alone there have been more than 474 published articles in relation to success of OSS and COSS (Gezici et al., 2019), yet the majority of COSS companies face a variety challenges to succeed (Ehls, 2017; Silic & Back, 2015; Ghapanchi and Aurum, 2012; Singh, Tan, and Mookerjee, 2011; Stewart & Maruping, 2006).

COSS company challenges can be broadly categorized into technical and non-technical (project and business management decisions related) (Verner, Sampson, & Cerpa, 2008). Among the major causes of technical problems affecting the COSS companies is vulnerability. The increased adoption of OSS by commercial companies has raised reported cases of vulnerability from 14,000 in 2017 to 16,000 in 2018 (De Villiers, 2019). Closely related to vulnerability is the issue of bugs that forces software programs to deviate from the expected operation (Chen, Shi, Shoga, Wang, & Boehm, 2018).

Bugs claim the lion share accounting for an alarming 81.1-86.7% of all software related problems (Tong, Ying, Hongyan, & Zhonghai, 2016). This is notwithstanding developers committing 45-80% of their valuable development time to bug evaluation and fixing. Often, bug inflicted failures take the form of system failure, malfunction or vulnerability (Akbarinasaji, Caglayan, and Bener, 2018; Zineddine, Alaoui, and Saidou, 2017; Tong, Ying, Hongyan, and Zhonghai, 2016).

Another important technical issue faced by COSS companies is software testing. Insufficient planning and testing of software both during the development and acceptance test phases may have dire consequences (Kaur & Sengupta, 2011). Additionally, maintainability and obsolescence are other issues rendering a software product useless. Maintainability problems may arise from an architectural design decision during initial development, while obsolescence is a result of features, applications or technologies no longer desired by the customer or client. Besides technical problems, COSS companies are affected by non-technical matters such as project abandonment, lack of sustainable participation, documentation problems, and failure to meet the demands of the OSS community (Daniel, Maruping, Cataldo, and Herbsleb, 2018; The Linux Foundation, 2017; Stol & Babar, 2010; Fang & Neufeld, 2009). Another non-technical issue affecting COSS companies is licensing. In some cases, the combination of different licenses is illegal. For instance, the use of GPL v2 in combination with Apache v2 is impossible as the condition of GPL v2 prohibit the mixing of licenses that exhibit stronger conditions (Lokhman, Mikkonen, Hammouda, Kazman, & Chen, 2013).

In addition, COSS companies are affected by quality uncertainty. Due to the availability of a wide variety of similar software products in the market customers are often unable to determine the quality of the product they intend to buy (Stol K. and Babar M.A., 2010; Zaidan et al., 2015). Moreover, software development efforts by COSS company can be influenced by predicaments such as inability to accurately budget, schedule, meet expectations of clients, lack of skills, goal ambiguity and communication problems (Peter as cited in Zahid, Haider, Farooq, Abid, and Ali, 2018). Hence, these problems are worrisome as they may have a detrimental effect on COSS company's success.

Consequently, technical and non-technical challenges can sometimes be financially devastating. For instance, software development failure has cost US business 30 billion USD in 2010 alone (Kaur & Sengupta, 2011). In 2014 an estimated 77.75 billion dollars were lost to software development failures (The Standish Group Report, 2014). Hence, in order to overcome these challenges associated with COSS company success, evaluating and understanding the critical factors behind successful COSS companies is important.

B. Updated DeLone and McLean IS Success Model

The first DeLone and McLean IS success model was developed in 1992 which was updated a decade later (DeLone & McLean, 2003). Since its development, the DeLone and McLean model (D & M model) has been applied in diverse disciplines and contexts for the past 18 years. For instance, it has been deployed to evaluate the success of expert systems, knowledge management systems, enterprise systems, online shopping, e-commerce, mandated use and vulnerary use of technology, etc. (Jeyaraj, 2020).

However, these are not the only settings D & M model has been used in. Romi (2013), has extended and validated D & M among financial institutions in

Palestine. Similarly, Sharma and Sharma (2019) have applied D & M to evaluate the growing use of mobile banking in Oman. The purpose of their study was to find out customers' intention to use mobile banking and the associated customer satisfaction (Sharma & Sharma, 2019). Likewise, Rahi and Abd.Ghani (2019) assessed the continuance intention of internet banking users.

Another area of study that D & M has found wide application is the education sector. For instance, Alzahrani, Mahmud, Ramayah, Alfarraj, and Alalwan (2017) have adopted the D & M model for the study of digital library systems among four universities in Malaysia. The goal of their study was to identify the causes or determinants of lower usage of digital libraries in Malaysia. Similarly, Kurt (2019) studied the Italian e-learning system using D & M model. More specifically, his investigation was aimed at evaluating students' perception of e-learning systems. An earlier study by Freeze, Alshare, Lane, and Wen (2010) has investigated the same theme.

In the telecom sector, Adroni and Sitorus (2017) have studied the success in implementation of Bardix information system in Indonesia by a telecom company called "Telkomsel". Closely related to the telecom sector, Wang, Tseng, Wang, Shih, and Chan (2019) have examined the success of mobile applications via observing customer loyalty and intention to reuse.

Although not specifically in COSS company settings, D & M model has been used in studies that have evaluated the success of OSS (Lee, Kim, & Gupta, 2009; Lee and Lee, 2012; Gezici, Tarhan, & Chouseinoglou, 2019). Noticeably these studies have adopted the dimensions of the D & M model to OSS context. The dimensions of the updated DeLone and McLean IS success model includes six dimensions: information quality, system quality, service quality, use, user satisfaction, and net benefits developed to assess information systems success (DeLone & McLean, 2003).

Since the D & M model was developed for information systems success, operationalizing the six dimensions to software production and marketing is essential. Consequently, information quality has been adjusted to product property. This is because information quality in the D & M model assumes processing or production of information (Gezici, Tarhan, & Chouseinoglou, 2019). In contrast product property is concerned with launching new software products, developing a complex piece of software, and the ability to add more lines of code (Gezici et al., 2019).

The second dimension, system quality, is also operationalized as software quality, since COSS companies generate revenue through sales, software products, and complementary services (Riehle, 2020). Furthermore, the service quality dimension is conceptualized to include community service (i.e., technical support by the OSS-community) (Lee, Kim, & Gupta, 2009; Gezici, Tarhan, & Chouseinoglou, 2019) and support from the COSS company (Riehle, 2012,2020)

In addition, whereas user satisfaction is observed in terms of ratings, reviews, and customer satisfaction, software use is measured using popularity and number of downloads (Crowston, Annabi, & Howison, 2003; Gezici,Tarhan, & Chouseinoglou, 2019). Finally, the net benefits dimension has been adapted as COSS company success.

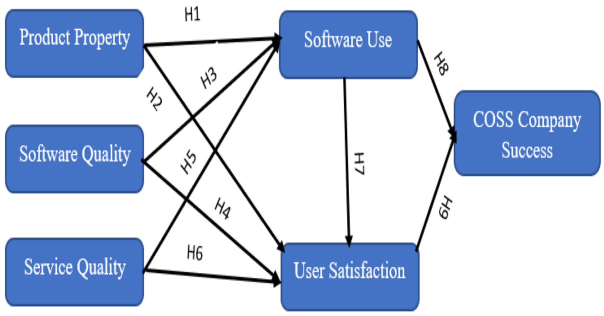

Figure 1. Research Model

Based on the above relationships, the following hypotheses have been developed and the research model is illustrated in Figure 1.

H 1: There is a significant positive relationship between product property and software use.

H 2: There is a significant relationship between product property and user satisfaction.

H 3: There is a significant positive relationship between software quality and software use.

H 4: There is a significant positive relationship between software quality and user satisfaction.

H 5: There is a significant positive relationship between service quality and software use.

H 6: There is a significant positive relationship between service quality and user satisfaction.

H 7: There is a significant positive relationship between software use and user satisfaction.

H 8: There is a significant positive relationship between software use and COSS company success.

H 9: There is a significant positive relationship between user satisfaction and COSS company success.

III RESEARCH DESIGN

The study adopts a quantitative survey method to assess the success of COSS companies adapting the DeLone and McLean updated IS success model.

A. Sample Population and Data Collection

The research model was tested by means of data gathered through survey sent to 3,750 randomly selected software development experts and users working in 49 COSS companies. Furthermore, the required sample size was determined by G*Power analysis. Accordingly, a minimum sample size of 176 respondents is required setting statistical power at 95%, effect size at 10%, error probability at 5%, and using 3 predictors (i.e., product property, software quality, and service quality). Thus, a total of 194 valid questionnaires were used for statistical analysis.

Out of the total of 194 respondents 65% have 6 or more years of experience, 32% have 1-5 years of experience, while the remaining 3% have less than 1 year of experience in software development. With respect to roles assumed while working in the present and past jobs 80% have participated in coding, 68% have taken part in software design, 49% have taken part in software requirements, 41% are engaged in software improvement process as well as software testing and integration, 34% have been involved in version management, yet another 24% have taken part in software quality assurance and users.

Table 1. Mardia's Multivariate Skewness and Kurtosis

	b	z	p-value
Skewness	11.4533	370.32344	0
Kurtosis	75.2073	19.33841	0

Finally, the skewness and kurtosis are not close to zero meaning data is not normally distributed. Thus, a nonparametric test is recommended. Among nonparametric tests, partial least square structural equation modeling (PLS SEM) is the most robust statistical tool. Consequently, the study utilizes PLS SEM multivariate analysis tool (Hair, Hult, Ringle, & Sarstedt, 2017). More specifically, a two-stage approach of structural equation modelling using SmartPLS 3.0 is adopted. The two-stage approach as the name implies involves two phases. In the initial stage the measurement model will be assessed. Followed by the evaluation of structural model in the second stage (Sarstedt, Hair, Cheah, Becker, & Ringle, 2019).

IV RESULTS AND DISCUSSIONS

The purpose of the study was to evaluate the success of COSS companies by adapting the updated DeLone and McLean IS Success Model. The results presented in Table 2 indicates that software use has an insignificant effect on company success (H5) as well as user satisfaction (H6). In addition, service quality seems to have no influence on user satisfaction (H8). This result is consistent with the result obtained by Lee and Lee (2012). Accordingly, they make a distinction between voluntary use of the software product as compared to mandatory use of information systems (Lee & Lee, 2012).

Secondly, the insignificant relationship between service quality and user satisfaction is also consistent with the findings of Wang, Tseng, Wang, Shih, and Chan (2019). Perhaps, this is associated with the unique nature of COSS companies. COSS companies provide professional support only for paying customers and nonpaying users often utilize the support of OSS-Community or help one another. Thus, service quality is not the same as in a traditional firm (Riehle, 2012, 2020).

Table 2. Path Coefficients

Hypotheses	Beta Value	t-Value ($t>1.96$)	p-Value ($p<0.05$)	Decision
H1	0.155	2.381	0.02	Supported
H2	0.267	3.723	0.00	Supported
H3	0.247	2.813	0.01	Supported
H4	0.398	5.245	0.00	Supported
H5	0.078	**0.874**	**0.38**	Not Supported
H6	0.128	**1.472**	**0.14**	Not Supported
H7	0.357	5.316	0.00	Supported
H8	0.069	**0.908**	**0.36**	Not Supported
H9	0.554	7.301	0.00	Supported

In contrast, H1, H2, H3, H4, H7, and H9 are all supported and the results are consistent with previous studies (e.g., Lee & Lee, 2012; Romi, 2013; Nugroho & Prasetyo, 2018). In addition, the research model explains 35% of the variability in COSS company success, 32.9% in software use, and 44.3% in user satisfaction. This can be considered a good result as compared to Lee, Kim, and Gupta's (2009) 28% for OSS use and 54% for user satisfaction. Lastly, model relevance assessment revealed $Q^2=0.21$ and PLS-predict 71% is interpreted as having medium predictive power (Shmueli et al., 2019). Hence, the adaption of DeLone and McLean Model in the current context was justified.

V IMPLICATIONS

Observing the total effect for the target variable, company success, would allow the examination of the result in a more sensible manner (Hair, Hult, Ringle, & Sarstedt, 2017). Accordingly, COSS company success is strongly influenced by user satisfaction (0.55), followed by software quality (0.26), product property (0.17), software use (0.15), and service quality (0.09). Therefore, COSS companies may need to give priority to the enhancement of user satisfaction by paying more attention to customer reviews and ratings, customer needs for features' enhancement, and software bug reports.

Secondly, internal quality assurance alone may not impact company success. Consequently, higher quality value propositions to customers should be supported with an effective and efficient marketing strategy such as word-of-mouth (WOM) marketing (Riehle, 2020). Thirdly, frequently launching new complex software products excites customers. Hence, COSS companies may opt to not only use rapid release cycles but also swiftly handle issues and fix bugs that may cause customer dissatisfaction (Costa, McIntosh, Kulesza, & Hassan, 2016). This also has implications to the improvement and promotion of software use. Finally, service quality can be enhanced through the provision of suitable support tools such as forums, wikis, documentation and self-help materials (Riehle, 2020).

VI CONCLUSION

The main aim of this study was to evaluate the success of COSS companies by adapting the updated DeLone and McLean Information Systems (IS) Success model. A review of previous studies revealed that a success model for assessing COSS companies is not yet available. Moreover, past studies have focused on the success of very few prominent COSS companies individually, instead of a more robust empirical study of a group of companies. Hence, a comprehensive empirical study incorporating 49 COSS companies of variable sizes, types, and prominence were analyzed. The result showed that product property, software quality, and user satisfaction significantly affect COSS company success, while service quality seems to have a partial influence. Additionally, the adaption of the updated DeLone and McLean Information Systems (IS) Success model is encouraging since all Q^2 value, Q^2 predict, and RMSE values demonstrate medium predictive relevance and predictive power. Finally, the findings advocate enhancement of user satisfaction, improvement of software quality efforts, rapid release, and prompt handling of issues and bugs.

VII LIMITATION AND FUTURE RESEARCH

Although the study contributes towards the application of the updated DeLone and McLean Information Systems (IS) Success model to a unique set of COSS companies that use a hybrid business model, it has some shortcomings that could be useful inputs for future research. First, the current study uses a quantitative survey; therefore, future studies could use a qualitative study to gain a deeper understanding of the diversity of COSS company success. Second, using a cross-sectional survey may not capture causality very well, hence a longitudinal study would further validate the use of the updated DeLone and McLean Information Systems (IS) Success model in the context of COSS companies.

REFERENCES

Adroni, N., & Sitorus, P. M. (2017). Evaluation of Badix Information System Implementation Success in Telkomsel Region of Sumbagsel using DeLone & McLean Model. International Journal of Business and Economic Affairs, 2(3), 211-219.

Akbarinasaji S., Caglayan B., & Bener A. (2018). Predicting bug-fixing time: A replication study using an open source software project . The Journal of Systems and Software , 136, 173–186.

Alzahrani, A. I., Mahmud, I., Ramayah, T., Alfarraj, O., & Alalwan, N. (2017). Modelling digital library success using the DeLone and McLean information system success model. Journal of Librarianship and Information Science, 1–6.

Amrollahi, A., Khansari, M., & Manian, A. (2014). How Open Source Software Succeeds? A Review of Research on Success of Open Source Software. International Journal of Information and Communication Technology Research, 6(2), 67-77.

Anand, A., Tiwari, R., Krishna, A., & Sharma, R. (2018). Comparative Analysis between Proprietary Software VS. Open-Source Software VS. Free Software. 5th IEEE International Conference on Parallel, Distributed and Grid Computing (PDGC-2018), 144-147. Solan: IEEE.

August T., Shin H., & Tunca T.I. (2017). Generating Value Through Open Source: Software Service Market Regulation and Licensing Policy. Information Systems Research, 1-20.

Chen C., Shi L., Shoga M., Wang Q., & Boehm B. (2018). How Do Defects Hurt Qualities? An Empirical Study on Characterizing A Software Maintainability Ontology in Open Source Software. IEEE, 226-237.

Costa, D. A., McIntosh, S., Kulesza, U., & Hassan, A. E. (2016). The Impact of Switching to a Rapid Release Cycle on the:An Empirical Study of the Mozilla Firefox Project. 2016 IEEE/ACM 13th Working Conference on Mining Software Repositories, 374-385. Austin: IEEE/ACM.

Crowston, K., Annabi, H., & Howison, J. (2003). Defining Open Source Software Project Success. ICIS 2003 Proceedings, 327-340. Association for Information Systems.

Daniel S. L., Maruping L.M., Cataldo M., & Herbsleb J. (2018). The Impact of Ideology Misfit on Open Source Software Communities and Companies. MIS Quarterly , 42 (4), 1069-1096.

Daniel, S., Midha, V., Bhattacherhjee, A., & Singh, S. P. (2018). Sourcing knowledge in open source software projects: The impacts of internal and external social capital on project success. Journal of Strategic Information Systems, 27 (2018), 237–256.

De Villiers M. (2019, 03 11). https://www.itweb.co.za/content/mYZRXv9PzeOqOgA8. Retrieved 09 06, 2019, from https://www.itweb.co.za/: https://www.itweb.co.za/content/mYZRXv9PzeOqOgA8

DeLone, W. H., & McLean, E. R. (2003). The DeLone and McLean Model of Information Systems Success: A Ten-Year Update. Journal of Management Information Systems, 19(4), 9–30.

Deodhar, S. J., Saxena, K., Gupta, R. K., & Ruohonen, M. (2012). Strategies for software-based hybrid business models. Journal of Strategic Information Systems, 21 (2012), 274–294.

Ehls, D. (2017). Open Source Project Collapse – Sources and Patterns of Failure. Proceedings of the 50th Hawaii International Conference on System Sciences, 5327-5336. Hawaii: Attribution-NonCommercial-NoDerivatives 4.0 International.

Fang, Y., & Neufeld, D. (2009). Understanding Sustained Participation in Open Source Software Projects. Journal of Management Information Systems, 25(4), 9-50.

Freeze, R. D., Alshare, K. A., Lane, P. L., & Wen, H. J. (2010). Freeze, Ronald D.; Alshare, Khaled A.; Lane, Peggy L.; and Wen, H. Joseph (2010) "IS Success Model in E-Learning Context Based on Students' Perceptions. Journal of Information Systems Education, 21(2), 173-184.

Gezici, B., Özdemir, N., Yılmaz, N., Coşkun, E., Tarhan, A., & Chouseinoglou, O. (2019). Quality and Success in Open Source Software: A Systematic Mapping. 2019 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), 363-370). Thessaloniki: IEEE.

Gezici, B., Tarhan, A., & Chouseinoglou, O. (2019). Internal and external quality in the evolution of mobile software: An exploratory study in open-source market. Information and Software Technology, 112 (2019), 178–200.

Ghapanchi, A. H., & Aurum, A. (2012). Competency rallying in electronic markets: implications for open source project success. Electron Markets, 22,117–127.

Hair, J. F., Hult, G. T., Ringle, C. M., & Sarstedt, M. (2017). A Primer on Partial Least Squares Structural Equation Modeling (PLS-SEM), 2nd Eds. London: SAGE Publications, Inc.

Jeyaraj, A. (2020). DeLone & McLean models of information system success: Critical meta-review and research directions. International Journal of Information Management , 54 (2020), 102139.

Kamal, Q.-u.-N., Mahoto, N. A., & Memon, I. (2018). Discovering Success Factors of OSS Projects. International Journal of Advanaced Studies in Computer Science and Engineering, 7(5), 4-11.

Kaur, R., & Sengupta, J. (2011). Software Process Models and Analysis on Failure of Software Development Projects. International Journal of Scientific & Engineering Research , 2(2), 1-4.

Kurt, Ö. E. (2019). Examining an e-learning system through the lens of the information systems success model: Empirical evidence from Italy. Education and Information Technologies, 24, 1173–1184.

Lee, S. M., & Lee, S.-H. (2012). Success factors of open-source enterprise information systems development. Industrial Management & Data Systems, 112(7), 1065-1084. .

Lee, S.-Y. T., Kim, H.-W., & Gupta, S. (2009). Measuring open source software success. Omega, 37 (2009), 426 – 438.

Lokhman, A., Mikkonen, T., Hammouda, I., Kazman, R., & Chen, H.-M. (2013). A Core-Periphery-Legality Architectural Style for Open Source System Development. 2013 46th Hawaii International Conference on System Sciences, 3140-3157. Hawaii: IEEE.

Mäntylä, M. V., Jørgensen, M., Ralph, P., & Erdogmus, H. (2017). Guest editorial for special section on success and failure in software engineering. Empirical Software Engineering Journal, 22,2281–2297.

Nugroho, Y., & Prasetyo, A. (2018). Assessing Information System Success: A Respecification of the Delone and McLean Model to Integrating the Perceived Quality. Problems and Persepectives in Management, 16(1), 348-360.

Rahi, S., & Abd.Ghani, M. (2019). Integration of DeLone and McLean and self-determination theory in internet banking continuance intention context. International Journal of Accounting & Information Management, 27 (3), 512-528.

Riehle, D. (2009). The Commercial Open Source Business Model: In: Nelson M.L., Shaw M.J., Strader T.J. (eds) Value Creation in E-Business Management. AMCIS 2009. Lecture Notes in Business Information Processing, 36,18–30. Berlin: Springer.

Riehle, D. (2012). The single-vendor commercial open course business model. Information Systems and E-Business Management, 10, 5-17.

Riehle, D. (2019). Single-Vendor Open Source at the Crossroads. Unpublished manuscript, 1-7.

Riehle, D. (2020). Open Source Expanded. Single-Vendor Open Source Firms [Editorial]. IEEE, 68-72.

Romi, I. M. (2013). Testing Delone and Mclean's Model in Financial Institutions. American Academic & Scholarly Research Journal, 5(3), 121-129.

Sarstedt, M., Hair, J. F., Cheah, J.-H., Becker, J.-M., & Ringle, C. M. (2019). How to specify, estimate, and validate higher-order constructs in PLS-SEM. Australasian Marketing Journal (AMJ), 27,197–211.

Shahrivar, S., Elahi, S., Hassanzadeh, A., & Montazer, G. (2018). A business model for commercial open source software: A systematic literature review. Information and Software Technology, 103 (2018), 202–214.

Sharma, S. K., & Sharma, M. (2019). Examining the role of trust and quality dimensions in the actual usage of mobile banking services: An empirical investigation. International Journal of Information Management , 44 (2019). 65–75.

Shmueli, G., Sarstedt, M., Hair, J. F., Cheah, J.-H., Ting, H., Vaithilingam, S., & Ringle, C. M. (2019). Predictive model assessment in PLS-SEM: guidelines for using PLSpredict. European Journal of Marketing, 53(11), 2322-2347.

Silic M., & Back A. (2015). Technological Risks of Open Source Software Adoption in the Organizational Context – Linux in Munich (LiMux) Case. Gallen: https://ssrn.com/abstract=2633005 or http://dx.doi.org/10.2139/ssrn.2633005.

Singh P.V., Tan Y., & Mookerjee V. (2011). Network Effects: the Influence of Structural Capital on Open Source Project Success. MIS Quarterly, 35(4), 813-829.

Stewart K. J., Ammeter A. P., & Maruping, L. (2006). Impacts of License Choice and Organizational Sponsorship on User Interest and Development Activity in Open Source Software Projects. Information Systems Research, 17 (2), 126–144.

Stol K., & Babar M.A. (2010). Challenges in Using Open Source Software in Product Development: A Review of the Literature. FLOSS-3, May 8, 2010,Cape Town , South Africa. 17-22. Cape Town: ACM.

The Linux Foundation. (March 15, 2017). 6-operational-challenges-to-using-open-source-software. Retrieved 07 05, 2019, from https://www.linuxfoundation.org/blog/2017/03/6-operational-challenges-to-using-open-source-software/

The Standish Group Report. (2014). Chaos Report. United States: The Standish Group International, Inc.

Tong J., Ying L., Hongyan T., & Zhonghai W. (2016). An Approach to Pinpointing Bug-induced Failure in Logs for Open Cloud Platforms. IEEE, 294-302.

Verner, J., Sampson, J., & Cerpa, N. (2008). What factors lead to software project failure? Second International Conference on Research Challenges in Information Science 71-80. Marrakech: IEEE.

Wang, Y.-S., Tseng, T. H., Wang, W.-T., Shih, Y.-W., & Chan, P.-Y. (2019). Developing and validating a mobile catering app success model. International Journal of Hospitality Management, 77 (2019) 19–30.

Zahid, A. H., Haider, M. W., Farooq, M. S., Abid, A., & Ali, A. (2018). A Critical Analysis of Software Failure Causes from Project Management Perspective. VFAST Transactions on Software Engineering, 13(3), 113-119.

Zaidan, A., Zaidan, B., Al-Haiqi, A., Kiah, M., Hussain, M., & Abdulnabi, M. (2015). Evaluation and selection of open-source EMR software packages based on integrated AHP and TOPSIS. Journal of Biomedical Informatics, 53: 390–404.

Zineddine M., Alaoui C., & Saidou N. (2017). Commercial software companies and open source community reaction to disclosed vulnerabilities: Case of Windows Server 2008 and Linux patching. 2017 International Conference on Wireless Technologies, Embedded and Intelligent Systems (WITS), 1-4. Fez: IEEE.