EST Journal of Educational Science and Technology
Volume 4 Number 3 December 2018 page 211-220
p-ISSN:2460-1497 and e-ISSN: 2477-3840
DOI: http://dx.doi.org/10.26858/est.v1i1.7110

A Preliminary Study Strategy on Development of Personalized Education Tool Using Joint Requirement Planning Technique

Fadhla Junus[1], Nur Andula[2]
[1] Information Technology Education, State Islamic University (UIN) Ar-Raniry Banda Aceh, Indonesia
Email: fadhlayn@ar-raniry.ac.id
[2] Information Technology Education, State Islamic University (UIN) Ar-Raniry Banda Aceh, Indonesia
Email: nurandula2@gmail.com

(***Received***: Oktober-2018; ***Reviewed:*** November-2018; ***Accepted:*** December-2018; ***Published:*** December-2018)

ABSTRACT

Involving a number of the prospective system user and applying various techniques to discover user requirements are fundamental in software engineering. This paper briefly reports upon the inceptive phase of software development life cycle (SDLC) that aimed to assemble potential user's recommendation in developing educational purposed application for the mobile system. The conducted study implemented Joint Requirement Planning (JRP) technique to substitute individual interview which spends a huge effort in time and cost. This small-scale research entangled 45 students of Vocational Information Technology at State Islamic University Ar-Raniry Banda Aceh Indonesia, whose acquaintance with both software engineering and pedagogical knowledge. They were entreated in observing eight selected applications that exist on the market. The software packages were utilized as a role model for developing a future personalized education tool (PET) system. The students' experience during JRP session then were qualitatively analyzed using QDA Miner to draw a recommendation for creating a better personal learning software. This study found that the explored applications were insufficient to fulfill students' needs, therefore the main outcome of this research expounds user requirements that will be used to design a better tool in supporting formal learning i.e. variety learning activities, incomplex navigation, and adaptable system.

Keywords*: system study; personal learning; requirement discovery; software engineering education; JRP technique.*

INTRODUCTION

Managing a preliminary study is fundamental in system study phase of software development life cycle (SDLC), because it describes fact-finding of proposed system. Bentley et al. (2007) stated 10 underlying principles for systems development of which three related to preliminary study strategy: (1) get the system users involved, (2) use a problem-solving approach, (3) establish phases and activities.

The conducted research was supposed to discover user requirements that will be used for building a personalized education tool (PET). PET is a mobile-based application which serves a personal learning environment (PLE) to its users, thus they are able to learn without regard to place and time. According to (Luksha & Peskov, 2014) the development of PET will expand students' opportunities in organizing their educational purposes. Therefore, PET system must have mechanisms to assist its user's achievements (Dagger, Wade, & Conlan, 2004).

Currently there are many sorts of PET that exist on market. Some of them are not only developed by formal institutions, but also by independent developers. This was triggered by Open Educational Resources (OER) movement which used Web 2.0 to serve openness and collaboration on the internet (Mikroyannidis, Okada, Little, & Connolly, 2011). Moreover, since massive open online course (MOOC) was introduced, it became a role model for developing PET applications.

Many literatures in SE reported on conducted research that involved students as subjects. Some examples of earlier work was arranged at IBM which embraced students in its longitudinal project (Runeson, Host, Rainer, & Regnell, 2012). Other related works were summarized in (Carver, Jaccheri, Morasca, & Shull, 2004): (1) Runeson who examined the distinctness between undergraduate and graduate students' viewpoints, (2) Höst et al.(2012) who concluded that the reseach goals must be corresponded to the teaching goals. Another supporting study also described on a small scale of software experiments using students as respondents. However, for larger scale of SE research usually would rather use practitioners whose industrial background than students. The reason of excluding students in participating the research because they are less experienced compared to professional workers (Keele, 2007).

Choosing students as subjects in a SE project was possible to organize when the students have knowledge in SE activities. Students who learned software development process may participate in the study. Moreover, students are part of the stakeholders of the developing system who will be impacted later on by the software (Bentley, Lonnie, Whitten, & Jeffrey, 2007; Carver, Jaccheri, Morasca, & Shull, 2004). Therefore, involving them is obligatory in a system study.

Designing a preliminary study must appropriate to the objective of the research. Kitchenham et al. (2002) mentioned that SE has unclear criteria in defining the form of contextual data which should be measured, acquired, and reported in the study design. Nevertheless, the following four recommendations must be taken into account in planning the study: (1) select subjects and objects of the study, (2) specify the selection process of population, (3) describe applicable procedures, (4) define study limits according to literature. Other study (Peffers, Tuunanen, Rothenberger, & Chatterjee, 2007) combined system development with research process that deduced these points: (1) review prior paper works, (2) elaborate systems, (3) conduct investigations, (4) observe findings.

Although there were no exact rules in system study, normally, software development process starts with exploring system user requirement. This process is intended to extract information that related to system problem and solution from the user group. As cited in Guidelines for conducting and reporting case study research in software engineering, Lethbridge et al. classified three ways in requirement discovery to direct, indirect and independent. The classification is similarly to Bentley, Lonnie, Whitten, and Jeffrey (2007) which covered interview system user, existing system inspection and sampling system documentations. Both categorizations could be practiced by collaborating university research with project of independent software developers or formal SE industries.

To the best of author's knowledge, there is unavailable standard which regulates on how a PET application should be. Furthermore, no academic researchs that specifically evaluate the PET software in system user's point of view. Hence, this paper briefly reveals the fact-finding techniques that implement theoretical approaches in software engineering (SE) to gain valuable information from students as the system user of PET.

METHOD

The method described here consists of theoritical frameworks that commonly applied in SE preliminary studies. According to Runeson & Höst (2009), a preliminary study in SE is identical with case study, consequently it can not be measured by statistical approach.

The conducted study was designed to acquire information about system users' expectation of PET applications. The collected information then was used to answer the research question whether PET system is ideal for their personal learning environment (PLE).

As reported by Kitchenham, Budgen, and Brereton (2011) that conforming to Verpoorten et al. (2009) who cited Glahn, Specht, and Koper (2007), an ideal PLE should meet two aspects i.e. learners' perspective in current learning context and evaluation system that allows learners to trace their learning activities. In addition to that, Valtonen et al. (2012) deduced a remarkable resemblance between the goal of PLE and personalized learning. Both had to supply a number of methods and applications that could be chosen based on students' ideas and needs. Furthermore, Jia and Zhang (2018) introduced six ground rules for the mobile MOOC design comprising clear navigation, simple layout, linear display, harmonious coloring, smooth video, and full interaction.

To obtain the above objective, the subject and object of study must be selected in advance. As mentioned in the related literatures, selecting subject in software development shoud involved system users who will be directly impacted later on by the software, resultantly, students are the most felicitous system user of PET.

This study comprised 45 students who comprehend the basic level of SE field such as designing and programming software as well as pedagogical knowledge. While object of this research was eight mobile applications that exist on application store (Junus, 2017).

The selected applications were intended to be role model for developing future PET. They were chosen based on user rating, number of downloaded users, and variety of provided programming courses. These three prerequisites must fulfill minimum point of four stars user rating, downloaded by 5,000 users, and containing five programming languages. Table 1 shows the selected softwares that met the defined requirements.

Table 1. Object of Study

PET Name	Defined Requirements		
	User Rating	Number of Downloaded User	Number of Programming Language
Coursera	4	79,775	Many
Udacity	4	19,029	Many
EdX	4	13,199	Many
Udemy	4.1	58,109	7
Programming Hub	4	46,770	17
SoloLearn	5	33,155	11
Learn Programming	4	12,130	30
Enki	4.5	5,656	5

This preliminary study used Joint Requirement Planning (JRP) for fact-finding techniques. As referred to Bentley, Lonnie, Whitten, & Jeffrey, (2007), JRP is an alternative method to interview in SE research. When collecting data through a normal interview process, several times of questioning and evaluation were carried on. By applying JRP, the spending time on requirement discovery can be reduced, as most JRP sessions normally last three days up to two weeks.

The JRP participants consist of a researcher whose role as facilitator, the students as part of user community, and two scribes who also functioned as information technology (IT) staff. To achieve the study objective effectively, the students were separated into nine discussion groups in which was occupied by five members. The conducted JRP session in this preliminary study took four working days, where each day is equal to five working hours. There were eight sessions, which correspond to the number of the examined applications, with time allocated for each period was 150 minutes. The agenda of JRP during days of observation show in table 2.

During the meeting hours, the students were required to explore and probe the eight PET applications with different interested topics of programming learning. Their experiences for each application must be able to answer the questions, as shown in table 3, which were given in digital format that could be accessed online. All answered queries of each person that were stored in the cloud then were deliberated within the group.

Throughout the time, the facilitator and the scribes visited the discussing groups to communicate about any technical difficulties which found in the time of discussion. Besides that, the visitation also fostered creative ideas of the students about an ideal future PET system from pedagogical context.

At the end of each session, a representative of each group generally described about the group's opinions to all JRP participants. Meanwhile, the facilitator and the scribes recorded the given descriptions.

The final mean of data that had been collected during the JRP session were both in text and voice format. The text data containing the answers of every participant were downloaded from the cloud, whereas recorded voice data comprising the summaries in every session were scripted into text format to ease the analysis process. Afterward, all of the data were analyzed using QDA Miner.

Analysis procedure in QDA Miner treated each imported text document as a variable which was affiliated to a case and was named according to the name of each text file. Each case related to a student's opinion.

Next substansial step was preparing to code each script which was also followed by picking code color. This stage was started with code name definition. The code name is a descriptive keyword of the given questions. Therefore, the analysis involved three codes that represented each question. Each code then was spanned to several subcodes according to code frequency that recurred in the scripts. Those subcodes were defined while the analyst was scanning the whole cases manually. To do this, the analyst had to read every single word in the sentence carefully. If a word, either explicitly or implicitly, was found and it was appropriate to the code names, it must be added as a subcode under the most related codes. This step then was followed by assigning the word to the defined subcodes.

Table 2. JRP Agenda

Day	Agenda	
	Activities	**Duration**
1	Opening: Research objective explanation JRP sessions description and regulation	10 minutes
	Session 1: PET exploration and discussion 1	150 minutes
	Session 2: PET exploration and discussion 2	150 minutes
2	Session 3: PET exploration and discussion 3	150 minutes
	Session 4: PET exploration and discussion 4	150 minutes
3	Session 5: PET exploration and discussion 5	150 minutes
	Session 6: PET exploration and discussion 6	150 minutes
4	Session 7: PET exploration and discussion 7	150 minutes
	Session 8: PET exploration and discussion 8	150 minutes
	Closing: All sessions conclusion	30 minutes

The last step in analysis phase was analyzing the code frequency. Before doing this, text and code retrieval were worth to perform in order to ensure the validity. It was done by using Retrieve feature, the analyst only needed to insert a keyword and specified in which searching unit the data was located. Finally, Coding Frequency feature was used for drawing the results of coded cases.

A preliminary study in SE must be able to ensure whether the taken methods is reliable.

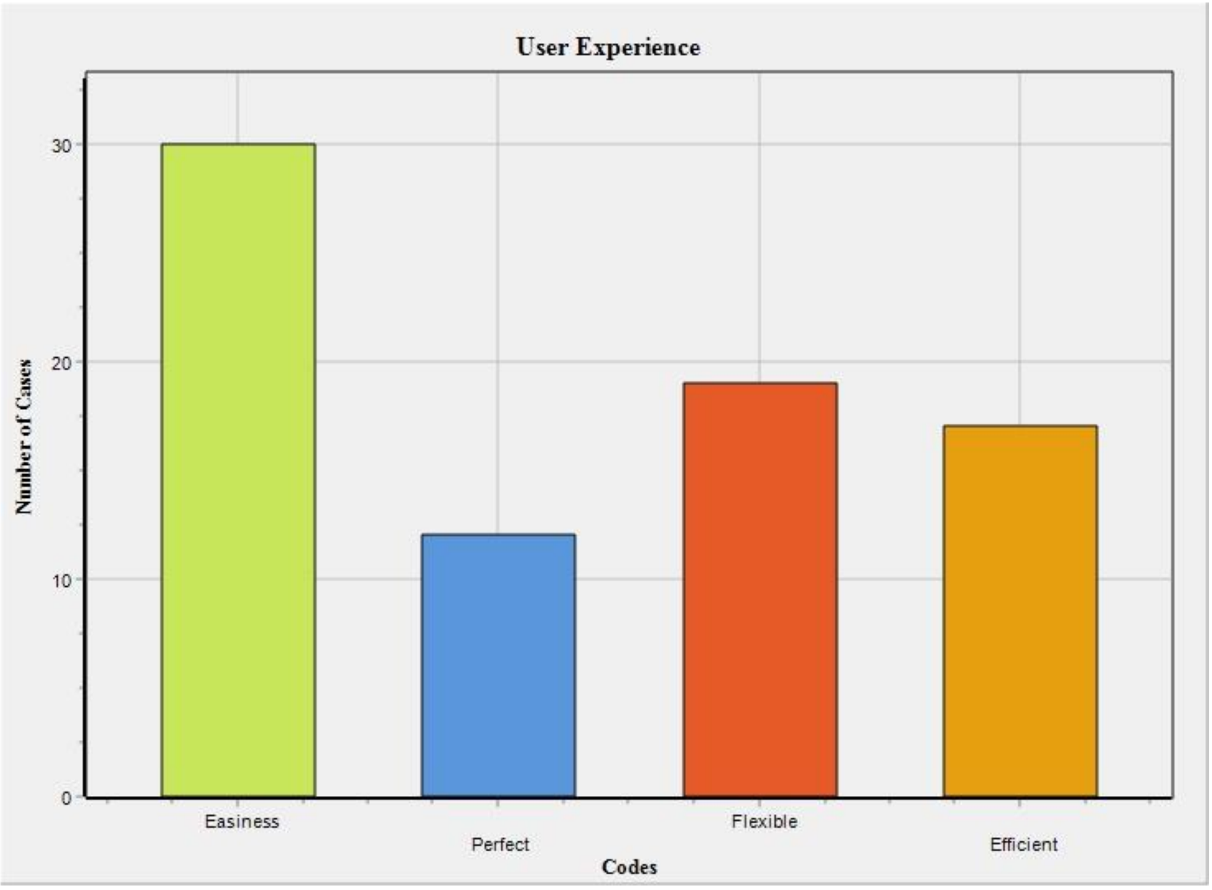
Any possible threats must be anticipated on the whole activities during software development because it affects the quality of a study. Validity procedure was prepared to address internal and external validity that threatened the conducted experiments in which its subject of study are students (Sjøberg et al., 2005; Runeson & Höst, 2009).

Two threats infected the internal validity of this study, they might arise due to a clausal relationship between factors that took part in the research. The conditions of the students during JRP sessions possibly caused their experiences on the observed software. Besides that, the second factor would be the system interface of PETs which could impress either good or bad mood of the students.

Table 3. List of JRP Questions

Number	Questions
1	Describe your learning experience about the observed applications!
2	What are the primary drawbacks that you found in these applications?
3	If you were a developer, what kind of features that you will provide to achieve an ideal PET application?

Figure 1. User Experience



While a risk for external validity could occure in data analysis phase. Incorrect script coding could generate a missed interpretation that would lead to fallibility in drawing conclusion.

The following procedures were taken to minimize the potential risks above: (1) visiting method which were done in the JRP sessions could keep the motivation of students in observing the PETs. The present of researcher to look over the problems and discuss any issues would reduce the threats to internal validity. Thus, the collected data could be more trustworthy. (2) doing text and code retrieval during data analysis could also address the threat to external validity, since the implicit phrases could only be interpreted by human.

RESULTS AND DISCUSSION

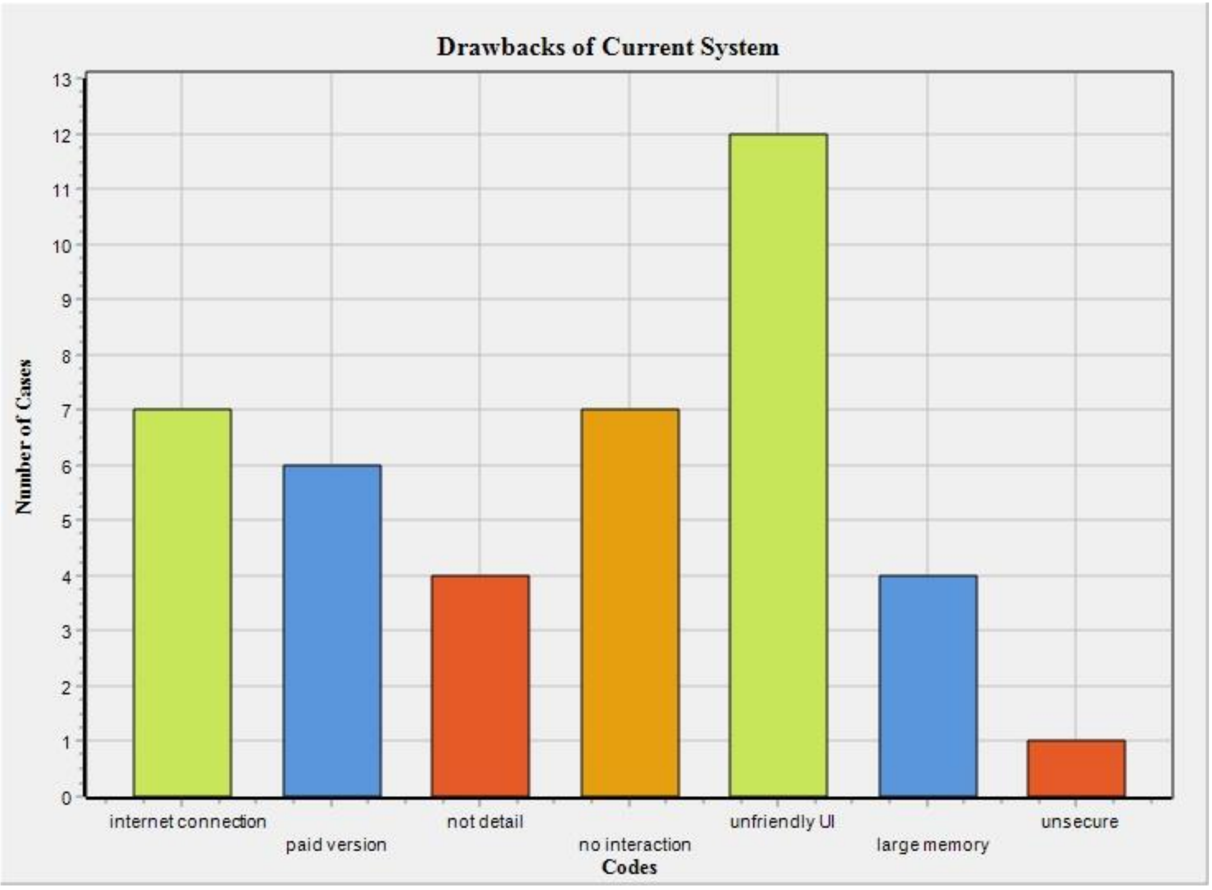
Result

Overall, the results presented below show the number of codes that appeared in data transcript of 45 cases during JRP session as stated in Table 2. The three pictures below represented the students' responses to the questions as explicated in Table 3.

Figure 1 above illustrates students experience with the selected PET applications as shown in Table 1. It is clearly seen that Easiness was the most frequent code as it appeared in 30 cases. Easiness explicitly iterated in 18 cases, while the rest of it was periphrastic. Most students opined that the existing applications helped them to learn easier on their interested topics without regarding to place and time. Moreover, the learning materials which was also possible to access in offline mode benefitted them in completing assignments, hence, they could learn more flexible and efficient.

Additionally, the first graph also depicts the less significance opinion of 12 respondents who were perfectly impressed on those applications. Interestingly, the number of impressed students were equal to whom commented on unfriendly user interface (UI) (Figure 2). This main drawback was criticized over following issues e.g. using English as UI language, sequential process in completing each learning chapter, and unused menu features such as social media sharing button.

Figure 2. Drawback off Current System

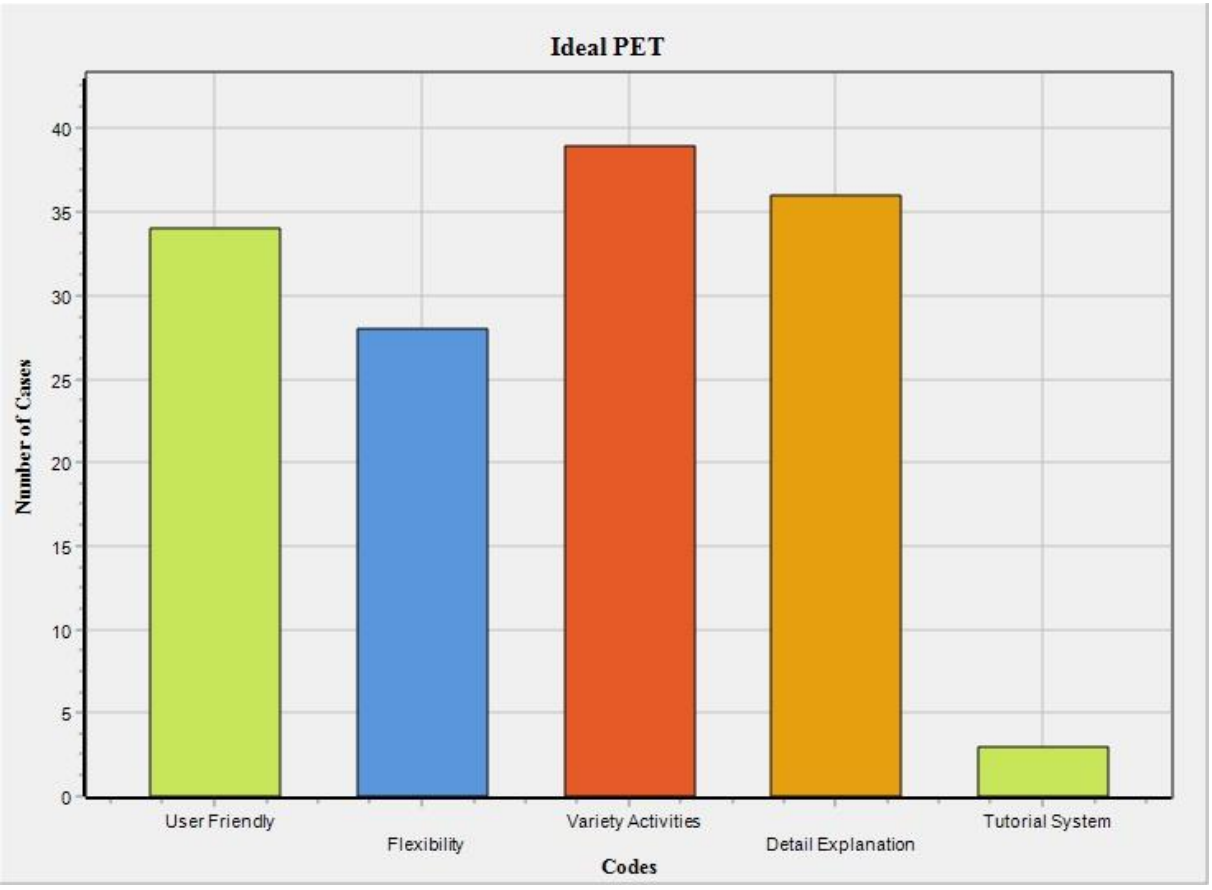


These critical issues were deteriorated by unsupervised learning activity (coded as no interaction) that might lead students to unsuccessful learning. Another encountered problem was also occured while accessing to particular learning materials, because they were available only in online and paid version. While the rest of the hindrances in existing softwares were not significant, the three listed upshots (coded as not detail, large memory, and unsecure) could be helpful to considering about things which need to be improved for future system.

As mentioned in the introduction and the method section, this preliminary study was directed to have students experience for a better PET system. Thus, the final analysis step construed the idea of students according to what they have acquaintance with the current systems. Fig. 3 reveals recommendations for an ideal PET system where more than 80% of students proposed two primary things that must be included in future system. Variation of learning activity was on the top then followed by a more extensive explanation.

Surprisingly, figure 3 also sketches a few number of students who suggested to include a prerecorded tutorial. They assumed the tutorial system would supervise them while learning. Even though the number of responded students was not significant at this point, however, it could also be a worthy consideration to develop in an upcoming PET application.

Figure 3. Ideal PET



Discussion

The main goal of this work was to reveal the students' views of the PET applications they had experienced with. The findings indicated two most advantageous things of the examined PET software were a facile learning method and adjustable access to learning material. A number of various topics in programming language assisted 66% students to learn easily anytime and anywhere. The results correspond to Valtonen et al. (2012) who clarified one of personal learning attributes was that should allow students to select the way for learning.

In contrary, more than 25% JRP participants expressed dissatisfaction about user interface involving the use of english as the primary language to communicate with the system, and the unfavorable navigation which allowed access to learning sources only in consecutive mode. Both disadvantages had caused discouragement of students whose either lower english proficiency or higher level of knowledge in computer programming. These findings disclosed the development of the explored PET applications which contradicted to the guildelines for mobile MOOC design (Jia & Zhang, 2018).

As stated in the method section, the obtained information would be used to design a mobile PET system that will be suitable to students' need. In spite of nearly 30% students showed direct expression about perfectness of PET for their PLE, a double number of them believed that PET would be more ideal if it involved more interesting features such as interactive quiz in a game format, live chat with experts, and console for testing the code. Besides that, a broader annotation of learning resources should also be ameliorated by succeeding system. As it would address the difficulties with compressed learning materials which were found in existing systems. The findings also outlined the future PET should have prerecorded video tutorial so learners could obtain a better understanding.

Conclusively, based on the review of eight PET applications during JRP session, the ideal PET that accomodated students' PLE was a composition of EdX and SoloLearn. Both applications have complied with PLE design principles which were stated by Kitchenham, Budgen, and Brereton (2011) based on the preceding study of Verpoorten et al. (2009) who cited Glahn, Specht, and Koper (2007).

Using PET in a formal education is still uncommon, thus the conducted study endeavored to evaluate the existing software that will be used in developing a blueprint for further research. The results reported in this paper only asserted within those eight PET applications which supplied information in programming language topics in perspective of a batch of students at State Islamic University Ar-Raniry. Consequently, any limitation during the work might be occured primarily in interpreting data, as it was one of the toughest phase at some point in this study.

ACKNOWLEDGMENT

Author would like to thank all the students of Vocational Information Technology at State Islamic University Ar-Raniry Banda Aceh who contributed information and experience to this study, which formed the basis for the conclusions made in this report.

CONCLUSIONS

This preliminary study was conducted in order to investigate what a prospective user absolutely needs from a PET system. Students are the most appropriate user of PET system, moreover, when the students have adequate knowledge base in sofware engineering as well as pedagogic.

JRP was used for explicating the students requisites in effective and efficient way. Hence, it is worth to demonstrate in a small-scale study as well as in a large-scale research in software engineering area.

According to the analysis of students perspective with several PET models, only one quarter of the potential users found that the current systems have fulfilled personal learning environment. Whereas majority of them indicated some weaknesses that must be improved. This work concludes the following features that must be taken into account for future PET system: (a) Variation of learning activities, In order to prevent tedious learning, the suggested system has to include more fascinating items e.g. interactive quiz and game on each level of learning chapter, real-time discussion with expert communities, and console for testing the code; (b) Simple user interface, The current system interface only used English for communicating with users. It could raise a problem for non-native English users. To deal with this issue, the students recommend to provide auto detected language based on user location. Furthermore, visibility of learning chapters could be retrieved directly in any level without having to complete each chapter sequentially. Other than that, the upcoming PET must be equipped with an evaluation feature which shows the learning pace of students from time to time; (c) Versatile system, An ideal PET system should offer many learning services. It must cover extensive knowledge area. A few students argued that a flawless PET model is the combination of EdX and SoloLearn. Both applications supplied detail and free learning materials which also be accessible while disconnecting from internet. Besides that, the given materials provided many examples, so the users do not need to look for comprehensive explanation from other learning sources, particularly, when the students need to perceive the meaning of few lines of code. In addition, PET would be easier for students when its system is embedded with video tutorial, because it could assist them in understanding the learning topic.

Although the above recommendations were deduced from a small-sized study, however in any case, these could be used by the developers of the observed applications to improve their existing systems. Besides that, a further study will be conducted by including more participants from another organizations in order to validate these preliminary results.

REFERENCES

Bentley, Lonnie D and Whitten, Jeffrey L. (2007). Systems Analysis and Design for the. Global Enterprise, 7 th. Edition, International Edition.

Carver, J., Jaccheri, L., Morasca, S., & Shull, F. (2004, September). Issues in using students in empirical studies in software engineering education. In Software Metrics Symposium, 2003. Proceedings. Ninth International (pp. 239-249). IEEE.

Dagger, D., Wade, V., and Conlan, O. 2004. Developing Active Learning Experiences for Adaptive Personalised eLearning. Paper presented at the 2nd International Workshop on Authoring Adaptive and Adaptable Educational Hypermedia.

Glahn, C., Specht, M., Koper, R., 2007, Smart indicators on learning interactions. In: E. Duval, R. Klamma, M. Wolpers (eds.) Creating new learning experiences on a global scale. LNCS, vol. 4753, (Berlin, Heidelberg: Springer), pp. 56-70. Cited in Verpoorten, D., Glahn, C., Kravcik, M.,Ternier, S., & Specht, M., 2009, Personalisation of Learning in Virtual Learning Environments. In U. Cress, V. Dimitrova & M. Specht (Eds.), Lecture Notes in Computer Sciences, Vol. 5794. (Berlin, Germany, Springer-Verlag), pp. 52-66.

Jia, J., & Zhang, B. (2018). *Design Guidelines for Mobile MOOC Learning — An Empirical Study*. Springer International Publishing. http://doi.org/10.1007/978-3-319-94505-7

Junus, F. (2017). The Role of Personalized Education Tools in Computer Programming Learning. In: 3rd International Conference on Education. [online] Kuala Lumpur: TIIKM, pp.92-98. Available at: https://doi.org/10.17501/icedu.2017.3110

Keele, S. (2007). Guidelines for performing systematic literature reviews in software engineering. In Technical report, Ver. 2.3 EBSE Technical Report. EBSE. sn.

Kitchenham, B. A., Pfleeger, S. L., Pickard, L. M., Jones, P. W., Hoaglin, D. C., El Emam, K., & Rosenberg, J. (2002). Preliminary guidelines for empirical research in software engineering. IEEE Transactions on software engineering, 28(8), 721-734.

Kitchenham, B. A., Budgen, D., & Brereton, O. P. (2011). Using mapping studies as the basis for further research–a participant-observer case study. Information and Software Technology, 53(6), 638-651.

Luksha, P., and Peskov, D., 2014, Future Agendas for Global Education. Executive Summary. Date of access: 16/10/2016. http://edu2035.org/pdf/GEF.Agenda_eng.pdf.

Mikroyannidis, A., Okada, A., Little, S. and Connolly, T., 2011, Supporting the collaborative adaptation of Open Educational Resources: The OpenScout Tool Library. In: ED-MEDIA 2011: World Conference on Educational Multimedia, Hypermedia & Telecommunications, Lisbon, Portugal, 27 June - 1 July, pp.3734-3739

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. Journal of management information systems, 24(3), 45-77.

Phases of System Development Life Cycle, http://oer.nios.ac.in/wiki/Phases_of_System_ Development_Life_Cycle, 2017.

Runeson, P., & Höst, M. (2009). Guidelines for conducting and reporting case study research in software engineering. Empirical software engineering, 14(2), 131.

Runeson, P., Host, M., Rainer, A., & Regnell, B. (2012). Case study research in software engineering: Guidelines and examples. John Wiley & Sons.

Sjøberg, D. I., Hannay, J. E., Hansen, O., Kampenes, V. B., Karahasanovic, A., Liborg, N. K., & Rekdal, A. C. (2005). A survey of controlled experiments in software engineering. IEEE transactions on software engineering, 31(9), 733-753.

Valtonen, T., Hacklin, S., Dillon, P., Vesisenaho, M., Kukkonen, J., & Hietanen, A. (2012). Computers & Education Perspectives on personal learning environments held by vocational students, *58*, 732–739. http://doi.org/10.1016/j.compedu.2011.09.025