IJPTL

International Journal of Practices in Teaching and Learning (IJPTL)

Published: August 2022

Trends in Technology-Enhanced Online Learning for Low Achievers in Introductory Programming: A Systematic Literature Review

Mahfudzah Othman
Universiti Teknologi MARA, Cawangan Perlis
02600 Arau, Malaysia
fudzah@uitm.edu.my

Aznoora Osman
Universiti Teknologi MARA, Cawangan Perlis
02600 Arau, Malaysia
aznoora@uitm.edu.my

Natrah Abdullah
Universiti Teknologi MARA, Cawangan Selangor
40450 Shah Alam, Selangor, Malaysia
natrah@fskm.uitm.edu.my

Siti Zulaiha Ahmad
Universiti Teknologi MARA, Cawangan Perlis
02600 Arau, Malaysia
sitizulaiha@uitm.edu.my

***Abstract*—** Technology-enhanced online learning (TEL) has been widely utilized in facilitating teaching and learning in higher academic institutions. Although exhaustive studies have been done in investigating the usage and effectiveness of TEL in programming education, there are still insufficient studies on the utilization of TEL in introductory programming specifically for low achievers who often faced learning difficulties due to: the lack of motivation and interests, the lack of problem-solving skills and logical thinking and the technical issues in the technologies used. Therefore, this study investigates the trends of TEL in introductory programming in terms of its pedagogical approach in addressing the difficulties faced by the low achievers and the online technologies needed to facilitate their learning through a systematic literature review study from the year 2017 to 2021. By adopting Kitchenham's methodology, a total of 20 papers were selected from various journals and databases. Researchers found that the trends of TEL for low achievers in introductory programming consisted of six main online educational technologies, which are the web-based learning, mobile learning, multimedia learning, gamification, block-based programming and virtual reality or robot simulations. The findings from this review can be used for future research in utilizing TEL in the field of programming education, especially for low achievers.

Keywords—technology-enhanced online learning, introductory programming, teaching and learning, low achievers

I. INTRODUCTION

For the first-year Computer Science students, learning programming is often faced with difficulties and challenges that eventually will lead to high failure rates (Margulieux et al., 2020). A recent study done by Bennedsen and Caspersen (2019) reported an average failure rate of 28% among the CSI students which was slightly less alarming compared to twelve years ago, which was 33%. Nevertheless, this remains as a challenge to the educators in introducing computer programming to the first-year students. This has also challenged the Computer Science community to develop more inclusive and effective learning environments and instructional methods to teach the introductory programming course (Bennedsen & Caspersen, 2019). For many years, the issues and challenges faced by the first-year students of Computer Science studies, especially in the introductory programming course have been widely discussed and researched. For instance, a recent study by Mehmood et al. (2020) stated that one of the reasons for the high failure rates in the introductory programming course is due to the lack of motivation and cognitive abilities among students. Others have addressed the issues of lack of interest as the subject itself is perceived to be difficult (Othman et al., 2019).

Previous studies have proposed many solutions to overcome this matter, and the use of TEL is one of the examples that have been proven effective in improving students' performances in introductory programming (Silva et al., 2020). For this study, researchers will revisit the issues and challenges faced by the low achiever students in the introductory programming course and the integration of TEL towards learning the course. TEL can be described as the use of computer-based technologies that support the learning process either locally or remotely (Sen & Leong, 2020).

Patrick and Doris (2019) also have stated that TEL can be described as an environment that relies on technologies that have been developed and deployed in le arning situations, which involves different activities such as communication and interactivity, knowledge, learning, entertainment, and exploration.

Nevertheless, despite TEL being widely researched and implemented, there is a scarcity of reviews done in the utilization of TEL in learning introductory programming, primarily in helping low achievers strive in this course. Therefore, this paper exhibits the results of a systematic literature review study performed on this subject. This paper is organized and presented as follows; Section 2 discusses the background of the study; Section 3 explains the systematic review method adopted; Section 4 discusses the results of the synthesis and analysis; and finally Section 5 represents the conclusion of this systematic literature review study.

II. Background Of Study

A. *Issues and Challenges in Teaching and Learning of Introductory Programming Course*

It is widely known that high failure and drop-out rates in introductory programming courses have been the issues to many higher academic institutions worldwide (Bennedsen & Caspersen, 2019; Margulieux et al, 2020). To overcome this, previous studies have investigated the issues and challenges in the teaching and learning of introductory programming in higher education. Among the issues and challenges identified were the lack of problem-solving skills, as well as the lack of cognitive and logical thinking skills (Malik et al., 2019; Margulieux et al., 2020). This will eventually lead to students failing to grasp the fundamental concepts of programming as mentioned by Othman et al. (2019). Besides that, the lack of interest in learning this subject will also cause the students to withdraw from this course at the early stage (Facey-Shaw et al., 2020).

Other studies have also mentioned that the lack of motivation and engagement in learning introductory programming will also result in students' early dropouts from the course (Khaleel et al., 2017; Figueiredo & Garcia-Penalvo, 2020). The other reasons mentioned in previous studies are the lack of prior knowledge (Halim & Phon, 2020), the technical nature of the programming languages used (Hidayanto et al., 2017) and issues regarding the methods and tools used in teaching and learning of introductory programming course (Hidayanto et al., 2017; Figueiredo & Garcia-Penalvo, 2020). Further analysis has also revealed that these teaching and learning difficulties were correlated to low performances or achievements in the introductory programming course (Bennedsen & Caspersen, 2019; Mehmood et al., 2020). Some students who were less interested in this course, were regarded as slow learners, would become hesitant and unable to cope with the programming learning environment that is often perceived as strenuous (Mehmood et al., 2020).

B. *Technology-enhanced Online Learning (TEL)*

As mentioned by Sen and Leong (2020), the term TEL is often used interchangeably with other synonyms to describe the usage of technologies that support human learning such as computer assisted instruction, digital technology, e-learning or educational technology. Although there are several other terms that are being used, TEL can be described as computer application or software, consisting of interrelated components that are able to assist and facilitate learners in the learning process whether individually or collaboratively (Patrick & Doris, 2019). Sen and Leong (2020) further explained that TEL is a computer application that is part of the learning system of the teaching and learning process, which aims to help students in their online learning. TEL has evolved positively over the years where it was initially utilized for simple forms and tutorials and has been progressively used for complex activities such as the use of intelligent systems (Patrick & Doris, 2019).

In learning computer programming, the utilization of TEL has been widely implemented, such as a study done by Ivanovic et al. (2017) that used Learning Management System (LMS) to facilitate the online teaching and learning of object-oriented programming (OOP). Another recent study done by Xinogalos et al. (2020) has also investigated the use of LMS, automated assessment tools and tutoring systems in the OOP learning environments. Jamil and Isiaq (2019) has also incorporated simulation-based programming where visualization technology is utilized in enriching the teaching and learning process. Nevertheless, none of those studies have utilized TEL tools to address the learning requirements of low achievers in introductory programming and how to cope with the issues and challenges. Furthermore, there is still a lack of reviews on these subject matters, which led to the construction of this paper that focuses on the trends of TEL tools for low achievers in introductory programming.

III. Methods

Kitchenham's methodology (Kitchenham, 2004) was adopted in conducting the systematic literature review as described in Silva et. al (2020). The following steps were performed: a) constructing the research protocols, b) performing the review, c) performing data synthesis and analysis, and d) reporting the review.

A. *Constructing the Research Protocol*

Developing the research questions: This review intends to investigate the trends of TEL that was utilized to cater to the low achievers in the introductory programming course. To achieve this, three research questions were constructed:

1) What are the issues and challenges addressed in the experiments?
2) How TEL is used to facilitate the teaching and learning of the introductory programming course, especially for the low achievers?
3) What are the TEL tools/technologies used in the experiments?

Keywords and database selection: An exhaustive literature search has been done in several journals and databases such as Google Scholar, IEEE, ACM Digital Library and Scopus to obtain papers related to the usage of TEL in introductory programming for low achievers from year 2017 until 2021. Several keywords such as "low achievers in introductory programming", "technology-enhanced online learning for introductory programming", "novice programmers" and some other related keywords were used during the search. Papers were also filtered and only peer-reviewed papers from conferences or journals as well as Master dissertation and PhD theses were considered.

Inclusion criteria: Additional inclusion criteria were also defined in this study:

1) pedagogical discussions must be related to the introductory programming including algorithm development and fundamental concepts of introductory programming.
2) the experiments were carried out at a higher education academic level.
3) students must interact with TEL tools or technologies explored in the studies.

B. Quality Analysis

To perform the quality assessment to the selected studies, the following criteria adopted from Silva et. al (2020) were proposed, which are: Q1) well-defined research objective, Q2) descriptions of the sample used. Q3) description of the experimental context and design, Q4) use of controlled trials that were randomly selected, Q5) use of pre and post testing and Q6) use of control or experimental group.

C. Review Execution

The review process was performed in two phases as shown in Fig 1. In the first phase, the pre-selection process was conducted, where the articles were analyzed based on keyword, title and abstract to find studies that met the inclusion criteria. Meanwhile, in the second phase, all selected articles were manually read where duplicate studies were excluded and sent for inclusion criteria inspection once again.

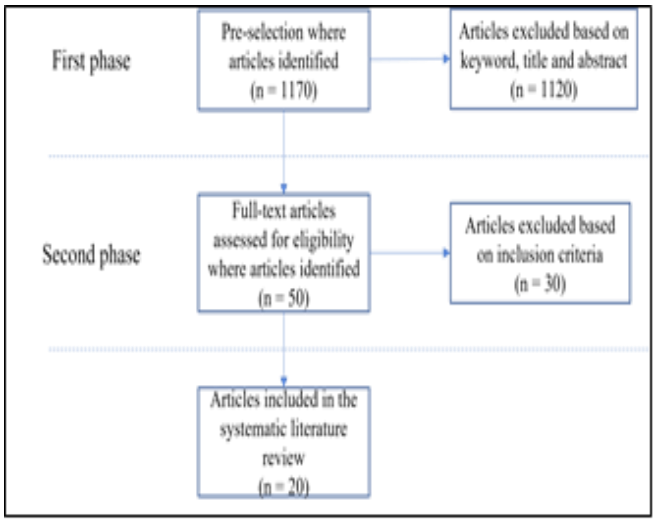

Figure 1: Process of the systematic literature review conducted

IV. RESULTS AND ANALYSIS

In the initial phase, one thousand one hundred and seventy articles were identified, where later, based on the analysis of the inclusion criteria, one thousand and twenty were excluded. Fifty papers that were sent to the second phase were selected for full-reading and went through a rigorous filtration and analysis to ensure the final selection of papers thoroughly met the inclusion criteria. Finally, twenty papers were selected in the final selection for the review.

A. Quality Analysis

For the analysis of the quality assessment, quality indicators (1-Yes, 0-No) that represent the quality criteria as mentioned previously were used. For each 'Y' indicator received by a paper, a total score of 1 will be given, and 0 if it receives 'N'. Table 1 shows the statistics of the quality assessments derived for each paper.

Table 1: Quality Indicators (Y – 1, N – 0)

No	Studies	Q1	Q2	Q3	Q4	Q5	Q6
1	Khaleel et al. (2017)	1	1	1	1	0	1
2	Halim & Phon (2020)	1	0	1	1	1	1
3	Malik et al. (2019)	1	0	1	0	1	1
4	Hidayanto et al. (2017)	1	0	1	1	1	1
5	Figueiredo & Garcia-Penalvo (2020)	1	0	1	1	0	1
6	Lepp et al. (2018)	1	1	1	1	0	1
7	Rogers et al. (2021)	1	0	1	1	0	1
8	Carbonaro (2018)	1	0	1	1	0	0
9	Winanti et al. (2020)	1	1	1	1	0	0
10	Tan & Lim (2019)	1	0	0	1	0	1
11	Facey-Shaw et al. (2020)	1	0	1	1	1	1
12	Chan et al. (2019)	1	1	1	1	0	1
13	Karnalim & Ayub (2017)	1	0	1	1	0	1
14	Kyfonidis et al. (2017)	1	0	1	1	0	1

No	Studies	Q1	Q2	Q3	Q4	Q5	Q6
15	Mutiawani et al. (2018)	1	0	1	1	0	1
16	Noor et al. (2020)	1	0	1	1	0	1
17	Stigall & Sharma (2017)	1	0	1	1	0	1
18	Kurniawan et al. (2018)	1	1	1	1	1	1
19	Oyelere et al. (2019)	1	0	1	1	0	0
20	Yassine et al. (2017)	1	1	1	1	0	1

Later, the mean score was calculated to ensure each paper achieves 50 percent eligibility to be included in the review. Fig 2 shows the chart of quality assessment scores derived for each paper, which explains that all selected papers have achieved 50 percent eligibility. Therefore, the studies selected for this review are considered relevant.

Figure 2: Quality assessment scores for each study selected in this review

B. Paper Analysis

The following are the results derived from the systematic literature review based on the research questions.

What are the issues and challenges addressed in the experiments? The selected studies were mapped to analyze the utilization of TEL in addressing each issue and challenge. From Table 2 below, most of the studies have fairly addressed each issue and challenge.

Table 2: Issues and challenges addressed in each study

No	Issues/Challenges	Studies	Count	Percentage
1	Lack of problem-solving skills	Hidayanto et al. (2017) Rogers et al. (2021) Chan et al. (2019) Malik et al. (2019) Halim & Phon (2020)	5	25%
2	Lack of cognitive skills and logical thinking	Kyfonidis et al. (2017) Carbonaro (2018) Malik et al. (2019) Noor et al. (2020)	4	20%
3	Lack of interest or motivation and engagement	Khaleel et al. (2017) Yassine et al. (2017) Carbonaro (2018) Lepp et al. (2018) Facey-Shaw et al. (2020) Figueiredo & Garcia-Penalvo (2020) Winanti et al. (2020)	7	35%
4	Lack of prior knowledge in programming	Hidayanto et al. (2017) Karnalim & Ayub (2017) Chan et al. (2019) Oyelere et al. (2019) Tan & Lim (2019)	5	25%
5	Technical aspects of the programming language used	Hidayanto et al. (2017) Karnalim & Ayub (2017) Kyfonidis et al. (2017) Mutiawani et al. (2018) Malik et al. (2019) Halim & Phon (2020)	6	30%
6	Teaching methods and tools used	Kyfonidis et al. (2017) Stigall & Sharma (2017) Kurniawan et al. (2018) Lepp et al. (2018) Tan & Lim (2019) Oyelere et al. (2019) Noor et al. (2020)	7	35%

Table 2 reveals that approximately 35 percent of the selected studies have addressed the issue of low achievers' lack of interest and motivation in learning this subject, while another 35 percent have addressed the issues and challenges associated with the teaching methods and tools used in

introductory programming courses. The conducted reviews also revealed that only 20 percent of the selected studies addressed the issues and difficulties associated with a lack of cognitive skills and logical thinking among low achievers. Low achievers' lack of cognitive skills and logical thinking necessitates additional research into the effectiveness of TEL in addressing this shortfall.

Moreover, despite the fact that issues 3 and 6 appeared to be investigated frequently, there is room for improvement in these areas. Additionally, according to this analysis, motivation and engagement in learning introductory programming appeared to be the primary reason for low achievers' lack of interest in this subject. Therefore, additional research is required to determine how TEL can be fully utilized to improve low achievers' motivation and interests in learning introductory programming.

How TEL is used to facilitate the teaching and learning of introductory programming course, especially for the low achievers? Table 3 shows the utilization of TEL in facilitating learning of introductory programming for low achievers based on the reviews done for each study.

Table 3: Utilization of TEL to facilitate the teaching and learning of introductory programming for low achievers

Paper	Studies	Results
Khaleel et al. (2017)	Integrating gamification elements and techniques in programming learning requirements	Students have agreed that learning via gamification has increased motivation and interest towards programming courses.
Halim & Phon (2020)	Using Learn C mobile application to learn programming	Results showed that there were no significant differences between pre and post-tests although several factors were identified during the experiment.
Malik et al. (2019)	A web-based application named PROBSOL that facilitates learning of pseudocode technique.	Positive impact on student learning outcome and attrition.
Hidayanto et al. (2017)	Development of adventure-based multimedia learning to learn introductory programming	Increased in students' understanding and satisfaction towards software, learning and visual communication.
Figueiredo & Garcia-Penalvo (2020)	The use of gamifications techniques such as leaderboard, points, immediate feedback and badges in learning programming.	Has increased students' motivation, class attendances and participation as well as improving grades.
Lepp et al. (2018)	Using troubleshooter to give hints and examples to students while learning	Most of the students found troubleshooting concepts in MOOC is very helpful in learning programming. Improves engagement.
	programming through MOOC	
Rogers et al. (2021)	Utilizing PeerWise online gamification app. Hexad survey was used to measure user preferences towards gamification elements.	Personalization of gamified learning attributes cannot be automatically decided using Hexad survey.
Carbonaro (2018)	Using a web-based peer code review and delivery of feedbacks system.	Has improved students' engagement, competency, and time management capabilities.
Winanti et al. (2020)	Developing a gamification framework for programming courses to increase motivation, pleasure and satisfaction.	The experiment results were not thoroughly discussed although it was reported that students' learning outcomes have increased.
Tan & Lim (2019)	Apply block-based programming (visual programming approach) using Code Studio as a preparatory course for Diploma students.	Has increased students' motivation, interest, and confidence level.
Facey-Shaw et al. (2020)	Using gamification digital badges to improve motivation, participation and recognition of learners.	Students agreed on using digital badges. However, the quantitative results showed digital badges did not directly affect motivation.
Chan et al. (2019)	Using mobile-learning platform with messaging and social media applications to enhance students' performance in programming.	Has increased students' confidence and motivations through visual learning experience.
Karnalim & Ayub (2017)	PythonTutor is a web-based app that supports program visualization used in learning introductory programming.	Students were able to conduct several sub-tasks even in advanced topics. Effective in improving students' understanding.
Kyfonidis et al. (2017)	Using a block-based programming app named Block-C for learning C programming language.	Students were more focused on the logic of programming instead of the syntax of C language.
Mutiawani et al. (2018)	Markas C is an interactive web-based e-learning with multimedia for introductory	Students were satisfied with the quality of the application (usability, information quality and service interaction).

Paper	Studies	Results
	programming C language.	
Noor et al. (2020)	Using Lego Mindstorm as a supporting educational tool in learning basic programming to encourage higher-order thinking and critical thinking.	Students claimed better understanding in computational thinking and problem-solving.
Stigall & Sharma (2017)	Using Virtual Reality (VR) instructional modules based on gaming metaphors to learn OOP.	Students agreed that the application is easy to use and effective in helping them learn introductory programming.
Kurniawan et al. (2018)	Using physical robot and robot simulator in learning introductory programming.	Results showed no significant differences between these two learning approaches.
Oyelere et al. (2019)	Using MobileEdu-Puzzle application in learning programming.	Has a positive impact towards students' understanding in learning programming and is easy to use.
Yassine et al. (2017)	Utilizing "Perobo", a serious game in teaching and learning C programming	The instructional design of "Perobo" has used the gameplay technique and pedagogical approach to increase students' engagement.

According to Table 3, the majority of studies focused on increasing low achievers' interests, motivation, and performance through a range of TEL approaches, with gamification appearing to be the primary research focus (Khaleel et al., 2017; Yassine et al., 2017; Figueiredo & Garcia-Penalvo, 2020; Winanti et al., 2020; Facey-Shaw et al., 2020). Meanwhile, the majority of the experiments in the studies were conducted on controlled groups of respondents who were enrolled in Computer Science studies but had insufficient programming skills. Thus, the TEL-based learning strategy appears to be beneficial at increasing students' knowledge and understanding of the course while also successfully promoting engagement and motivation (Hidayanto et al., 2017; Lepp et al., 2018; Carbonaro, 2018; Malik et al., 2019; Tan & Lim, 2019; Noor et al., 2020).

Nonetheless, none of the studies examined how TEL can benefit low achievers' cognitive abilities and logical thinking. As a result, this has created the possibility of future research on how to design and use TEL to boost low achievers' cognitive abilities and logical reasoning in introductory programming courses.

What are the TEL tools/technologies used in those experiments? From the reviews derived in Table 3, the trends of TEL tools and technologies used in teaching and learning introductory programming can be finally concluded as depicted in Table 4. We have listed six main TEL tools and technologies from the reviews, which are i) web-based learning/MOOCs/e-Learning, ii) Mobile learning application, iii) Multimedia learning application, iv) Gamifications/Game-based learning, v) Block-based/visual programming, and vi) Virtual reality/robot simulation.

Table 4: TEL tools/technologies used in each study

No	TEL Tools/Technologies	Studies	Count	Percentage
1	Web-based learning/MOOCs/e-Learning	Karnalim & Ayub (2017) Carbonaro (2018) Mutiawani et al. (2018) Malik et al. (2019) Rogers et al. (2021)	5	25%
2	Mobile learning application	Yassine et al. (2017) Chan et al. (2019) Halim & Phon (2020)	3	15%
3	Multimedia learning application	Hidayanto et al. (2017) Mutiawani et al. (2018)	2	10%
4	Gamifications/Game-based learning	Khaleel et al. (2017) Hidayanto et al. (2017) Yassine et al. (2017) Oyelere et al. (2019) Facey-Shaw et al. (2020) Figueiredo & Garcia-Penalvo (2020) Noor et al. (2020) Winanti et al. (2020) Rogers et al. (2021)	9	45%
5	Block-based/visual programming	Karnalim & Ayub (2017) Kyfonidis et al. (2017) Tan & Lim (2019)	3	15%
6	Virtual reality/robot simulation	Stigall & Sharma (2017)	2	10%

No	TEL Tools/Technologies	Studies	Count	Percentage
		Kurniawan et al. (2018)		

As shown in Table 4, gamification or game-based learning has garnered the interest of researchers to explore in the teaching and learning of introductory programming, with a 45 percent score from the total number of selected studies. For multimedia learning applications, there were only two studies conducted between 2017 and 2021 that specifically addressed the issues of low achievers, accounting for only 10 percent of the studies reviewed here. This is another potential area that can be investigated further and combined with other tools, such as gamification.

V. Conclusion and Future Research

As a conclusion of this comprehensive systematic review, the findings suggest that future research should concentrate on the use of TEL in teaching and learning introductory programming, with an emphasis on meeting the needs of students with low cognitive skills and logical reasoning, such as through the hybridization of multimedia learning and gamification. In addition, the review identified rising tendencies in the use of TEL tools and technologies in gamification or game-based learning, as well as in web-based learning platforms such as e-learning and MOOCs. Thus, gamification in e-learning or MOOCs platforms can be investigated further, where efforts to increase low achiever students' motivation and cognitive skills can be recognised.

References

Bennedsen, J. & Caspersen, M.E. (2019). Failure rate in introductory programming. *ACM Inroads*, 10, 30 – 36.

Carbonaro, A. (2018). Good practices to influence engagement and learning outcomes on a traditional introductory programming course. *Interactive Learning Environments*, 2018, 1–8.

Chan, T. L., Wei, K., Sio Kei, K. I., Gomes, A., Jose Mendes, A. & Marcelino, M. J. (2019). Students' characteristics in programming learning and the design of a mobile learning platform, *International Journal of Mobile Learning and Organisation*, 13:4, 352-391.

Facey-Shaw, L., Specht, M., Van Rosmalen, P. & Bartley-Bryan, J. (2020). Do Badges Affect Intrinsic Motivation in Introductory Programming Students? *Simulation & Gaming*, 51(1), 33–54. https://doi.org/10.1177/1046878119884996.

Figueiredo, J. & Garcia-Penalvo, F. J. (2020). Increasing student motivation in computer programming with gamification. *IEEE Global Engineering Education Conference (EDUCON)*, pp. 997-1000.

Halim, N. F. A & Phon, D. N. E. (2020). Mobile learning application impact towards student performance in programming subject, *IOP Conf. Ser.: Mater. Sci. Eng*., 769 012056.

Hidayanto, D. R., Munir, Rahman, E. F. & Kusnendar, J. (2017). The application of ADDIE model in developing adventure game-based multimedia learning to improve students' understanding of basic programming. *3rd International Conference on Science in Information Technology (ICSITech)*, pp. 307-312.

Ivanovic, M., Xinogalos, S., Pitner, T. (2017). Technology enhanced learning in programming courses – international perspective. *Educ Inf Technol* 22**,** 2981–3003.

Jamil, M.G. & Isiaq, S.O. (2019). Teaching technology with technology: approaches to bridging learning and teaching gaps in simulation-based programming education. *Int J Educ Technol High Educ* 16, 25. https://doi.org/10.1186/s41239-019-0159-9.

Karnalim, O. & Ayub, M. (2017). The Effectiveness of a Program Visualization Tool on Introductory Programming: A Case Study with PythonTutor. *CommIT (Communication & Information Technology) Journal*, 11(2), 67–76.

Khaleel, F.L., Ashaari, N.S., Meriam, T.S., Wook, T. & Ismail, A. (2017). Programming learning requirements based on multi perspectives. *International Journal of Electrical and Computer Engineering*, vol. 7, pp. 1-8.

Kitchenham, B. (2004). *Procedures for performing systematic reviews*. Keele, UK, Keele University, vol. 33, no. 2004, pp. 1–26.

Kurniawan, O., Lee, N. T. S, Datta, S., Sockalingam, N. & Leong, P. K. (2018). Effectiveness of Physical Robot Versus Robot Simulator in Teaching Introductory Programming. *IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)*, pp. 486-493.

Kyfonidis, C., Moumoutzis, N. & Christodoulakis, S. (2017). Block-C: A block-based programming teaching tool to facilitate introductory C programming courses. *IEEE Global Engineering Education Conference (EDUCON)*, pp. 570-579.

Lepp, M., Palts, T., Luik, P., Papli, K., Suviste, R., Sade, M., Hollo, K., Vaherpuu, V. & Tonisson, E. (2018). Troubleshooters for Tasks of Introductory Programming MOOCs. *International Review of Research in Open and Distributed Learning*, 19(4), https://doi.org/10.19173/irrodl.v19i4.3639.

Malik, S. I., Mathew, R. & Hammood, M. M. (2019). PROBSOL: A Web-Based Application to Develop Problem-Solving Skills in Introductory Programming, in Al-Masri A., Curran K. (eds) *Smart Technologies and Innovation for a Sustainable Future. Advances in Science, Technology & Innovation (IEREK Interdisciplinary Series for Sustainable Development)*, Springer, Cham.

Margulieux, L.E, Morrison, B.B. & Decker, A. (2020). Reducing withdrawal and failure rates in introductory programming with subgoal labeled worked examples. *International Journal of STEM Education*, 7, 1-16.

Mehmood, E., Abid, A., Farooq, M.S & Nawaz, N.A. (2020). Curriculum, teaching and learning, and assessments for introductory programming course. *IEEE Access*, vol. 8, pp. 125961-125981.

Mutiawani, V., Elfa, V., Jumadin, J., Amiren, M., Fauzie Afidh, R.P. & Subianto, M. (2018). Markas C: An E-Learning Media for Introductory Programming. *International Conference on Electrical Engineering and Informatics (ICELTICs)*, 2018, pp. 127-132.

Noor, F. H., Mohamad, F. S. & Minoi, J. L. (2020). Learning Programming using Lego Mindstorms: Analysis of Learner Experiences. *IEEE 8th R10 Humanitarian Technology Conference (R10-HTC)*, 2020, pp. 1-6.

Othman, M., Rosmani, A., Mohd Fauzi, S., & Mazlan, U. (2019). The Impact of Pair Programming on Students Logical Thinking_A Case Study on Higher Academic Institutions. *Social And Management Research Journal*, *16*(1), 85-98. doi:10.24191/smrj.v16i1.6085.

Oyelere, S. S., Agbo, F. J., Sanusi, I. T., Yunusa, A. A. & Sunday, K. (2019). Impact of Puzzle-Based Learning Technique for Programming Education in Nigeria Context. *IEEE 19th International Conference on Advanced Learning Technologies (ICALT)*, pp. 239-241.

Patrick, S. & Doris, W. (2019). Influential factors for technology-enhanced learning: professionals' views. *Journal of Research in Innovative Teaching & Learning*, Vol. 12 No. 3, pp. 268-294 Emerald Publishing Limited 2397-7604.

Rogers, M., Yao, W., Luxton-Reilly, A., Leinonen, J., Lottridge, D. & Denny, P. (2021). Exploring Personalization of Gamification in an Introductory Programming Course. *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education (SIGCSE '21).* Association for Computing Machinery, New York, NY, USA, 1121–1127.

Sen, A. & Leong, C.K.C. (2020). Technology-Enhanced Learning. In *Tatnall A. (eds) Encyclopedia of Education and Information Technologies*. Springer, Cham.

Silva, L., Mendes, A.J. & Gomes, A. (2020). Computer-supported collaborative learning in programming education: A systematic literature review. *IEEE Global Engineering Education Conference (EDUCON)*, Porto, Portugal, pp. 1086-1095, 27–30 April, 2020.

Stigall, J. & Sharma, S. (2017). Virtual reality instructional modules for introductory programming courses. *IEEE Integrated STEM Education Conference (ISEC)*, pp. 34-42.

Tan, E. X. & Lim, E. L. (2019). Block-based programming: The way to introduce computer programming. *Proceedings of the International University Carnival on e-Learning (IUCEL) 2019*, pp. 33-34.

Winanti, Abbas, B.S., Suparta, W., Heryadi, Y. & Gaol, F. L. (2020). Gamification Framework for Programming Course in Higher Education, *Journal of Game, Game Art and Gamification*, Vol. 05, No. 02.

Xinogalos, S., Ivanovic, M., Savic, M. & Pitner T. (2020). Technology-Enhanced Learning in Programming Courses, Role of. In: Tatnall A. (eds) *Encyclopedia of Education and Information Technologies*. Springer, Cham. https://doi.org/10.1007/978-3-319-60013-0_218-1.

Yassine, A., Chenouni, D., Berrada, M., & Tahiri, A. (2017). A Serious Game for Learning C Programming Language Concepts Using Solo Taxonomy. *International Journal Of Emerging Technologies In Learning (IJET), 12(03)*, pp. 110-127.