

Readiness for the Implementation of Ubiquitous Learning in Programming Course in Higher Education

<https://doi.org/10.3991/ijim.v15i18.24547>

Lucia Sri Istiyowati^(✉), Zulfiati Syahrial, Suyitno Muslim
Universitas Negeri Jakarta, Jakarta, Indonesia
LuciaSriIstiyowati_7117157702@mhs.unj.ac.id

Abstract—The use of information technology in education is inevitable by the increasing utilization of gadgets and the internet. The emergence of generation Z has driven the need for transformation in learning process, including in Higher Education area. Learning programming is important and fundamental for informatics students. Numerous endeavors have been executed to achieve satisfactory results that is absorption of graduates within the society and industry. However, there are still many efforts that must be done to meet the high need for informatics graduates who are experts in programming. Ubiquitous learning (u-learning) is a mean for learning anywhere, anytime and through anything that aims to provide the right information at the right time and place to accommodate a lifestyle by utilizing technology. In order to harmonize the current generation Z learning styles, u-learning is considered suitable to comply with their style. This research is a descriptive study design to describe the current situation as a basis for finding facts. The results showed that the students' perceptions showed that the support of the u-learning environment could support them in improving programming learning outcomes and u-learning could be applied in programming learning.

Keywords—u-learning, perception, programming, instructional

1 Introduction

Currently, many programming experts are needed by the industry. Many universities produce graduates in this field but it is still difficult for the industrial world to find suitable resources to meet their needs [1], [2], especially in Indonesia. The large number of IT students who choose not to wrestle with “coding” when they study, this causes the phenomenon of the lack of IT graduates who are proficient in programming when they graduate [3]. Higher education institutions that produce IT resources can make this a driving force in preparing students who have not only technical abilities but also holistic abilities [4]. It takes a strong foundation for students who will later become professionals in technology, a field with fast change. One of the challenges in informatics education is programming, the challenge in understanding the programming process and the practice of a programmer to transfer knowledge and skills effectively. Programming courses are core courses in the informatics field of study programs.

Learning to program is considered difficult and challenging [5–9], even for students majoring in informatics. This view makes students “afraid” to pursue programming courses, even though the need for programmers is getting higher. Learning programming is known to be difficult because students have to learn cognitive activities and practical activities, both of which are elements that are foreign to beginners [10]. The results of research conducted by J. Bennedsen and ME Caspersen, the passing rate of programming courses was 69% [11], which was then re-examined by Watson and Li [12] with results that were not much different. the pass rate of the course. This is almost the same in almost all countries in the world, which is around 67.7% [12]. Similar to students who are just starting to learn programming, for teachers, teaching programming is a big challenge [13]. Educators/lecturers face many challenges as they have to take responsibility for the academic performance of their students and the opportunities that exist to teach outside of traditional methods through the use of innovative learning media and technologies.

Along with the development of technology, came Gen Z (born in 1995), a generation born in the digital era that uses technology for everything. The results of research from Barnes and Noble College (2015) show that Gen Z students are very independent and open to new ways of presenting material both inside and outside the classroom. They prefer practical or hands-on learning and discussion methods they prefer, as well as collaborative learning environments that are not only limited to direct interaction but also use digital tools such as Skype and online forums. These learner characteristics provide opportunities for universities to connect with their students in more exploratory and meaningful interactions in all respects including those leading to better open learning opportunities and distance learning from cloud-based platforms [14]. The use of technology for the learning needs to be continuously developed considering the involvement of technology occurs in almost all human activities, which causes information sources to be easily obtained anytime and anywhere. Almost everyone is connected and likes to use electronic devices such as computers, smartphones, iPad, Tabs, and others. This can be used as a concern and a means for educators in developing learning so that technology devices can be used more as a means of learning.

Based on a survey conducted by the Indonesian Internet Service Providers Association in 2017, the largest internet users are aged 19–34 years, this age range is the age of Higher Education students, and the devices that are widely used to access them are smartphones or tablets (mobile devices). The use of mobile devices is one way of implementing learning anywhere at any time that is commonly used by students and has advantages and disadvantages and has a positive influence on distance learning (e-learning) [15]. Learning activities with this device can involve students more in the learning process. Students in the classroom learn passively become learners who are really actively involved, intellectually and emotionally involved in their learning tasks [16]. The application of learning using a mobile device must be prepared and designed. Higher education must implement strategic efforts to build its implementation plans, such as design guidelines, development phases, and norms, and consider the current level of readiness of students [17] so that the learning process can go well. The use of information and communication

technology is expected to improve the learning process, causing learning to occur anytime and anywhere, so that in the end it can improve the abilities of students. The learning system needs to be designed properly so that students can adapt and are interested in continuing to learn so that the students' abilities increase and meet the expected skill competencies.

Ubiquitous learning (u-learning) is the development of E-learning, with the use of technology in various aspects of life and the flexibility of mobile devices. One of the goals of U-learning is to provide the right resource at the right time and in the best way for learners. According to Aljohani et al. [18], u-learning can be done with new tools (not only with mobile devices), involves many users, and allows the continuation of the relationship between teachers and students mediated by a large number of devices that act in appropriate situations, collectively to support learning [18]. U-learning has several characteristics that can support 21st-century learning with Gen Z as students, including permanent, accessible anywhere, information that can be obtained immediately, the interaction between students, teachers, and experts, also understanding the learning environment [19]. The application of the u-learning concept to formal education in Indonesia can be carried out effectively and efficiently if it is carried out using a systemic and holistic approach. Universities in Indonesia must have the courage to answer this challenge by making transformations in adopting u-learning into their learning models [20].

Based on the challenges of higher education, especially in the field of information technology, especially for programming courses, the characteristics of students as generation Z, this study aims to see an overview of the programming learning process that has occurred and the readiness of students to apply u-learning.

2 Literature review

2.1 Instructional

Learning and instructional. Learning is a change in a person's abilities, beliefs, knowledge, and/or skills [21], it can also be defined as the development of new knowledge, skills, or attitudes as an individual's interaction with information and the environment [22]. Learning according to Schunk is a change in behavior or in the capacity to behave in a certain way, resulting from practice or other forms of experience, it is also stated that the learning criterion is that learning involves change, learning persists over time, learning occurs through experience [23]. Based on existing definitions, learning is defined as a process of gaining abilities, attitudes, behaviors, knowledge, and/or skills so as to cause change for a person through practice or other forms of experience such as individual interaction with information and the environment.

Instructional is anything that is done deliberately to facilitate learning [24] and performance, oriented towards organizational goals [21]. Instructional is a series of activities that are planned in advance by education providers or by teachers and are directed at certain learning outcomes. In general, Instructional can be defined as anything that is deliberate, planned, and directed to facilitate student learning. In the development of learning, principles are used so that effective learning activities occur [25].

In facilitating learning, a learning model is needed. A learning model is a method designed to promote specific learning outcomes related to the standards required in an academic discipline through the use of a series of specially designed activities through specific steps that are deliberately structured and support the achievement of specific cognitive, psychomotor, and/or effective goals [26]. The learning model is a description of the learning environment that is used to plan curriculum, lectures, units, or lessons to design learning materials because the model provides learning tools for students. The learning model is a way to build an ecosystem that maintains and stimulates students to learn by interacting with its components. Models are used to plan and use lessons, units, and curricula to design learning materials [27]. A learning model consists of a syntax that describes learning activities in full, a social system that describes the roles and relationships of learning actors, reaction principles related to reaction rules to student responses in-class interactions, and a support system that includes the teaching tools used. The learning model is a description of the learning environment that facilitates learning using a series of activities specifically designed to stimulate students to learn by interacting with its components such as student learning tools to achieve specific goals.

2.2 Ubiquitous learning (U-learning)

The term ‘ubiquitous learning’ appears along with the rapid development of information and communication technology and has a lot to do with human life. Ubiquitous Learning (U-Learning) is based on Ubiquitous computing [20], [28]. The purpose of u-learning is to provide the right information at the right time and place to accommodate lifestyle and work [19]. U-learning is the development of E-learning, and by utilizing technology in various aspects of life and the flexibility of mobile devices [29]. The difference between u-learning and its predecessors such as e-learning and m-learning lies in how technology is able to predict information or knowledge and what kind of learning context is needed by students [30].

U-learning is a learning paradigm in a ubiquitous computing environment that enables learning something at the right time, place, and within the right way [19], made possible in part by the capabilities of digital media Cope & Kalantzis, [28] which provides a learning resource (network) with communication and connectivity anytime, anywhere based on the situation of students using smart devices such as smartphones, tablet PCs, and smart PCs [31]. Another definition of u-learning is any learning environment that allows students to access learning content in any location at any time, regardless of whether wireless communication or mobile devices are used [30]. Inssaf El Guabassi et al. [32] Defining u-learning as a way of using new technologies to enhance learning and broadening the traditional perspective of the learning process, the main objective of learning is to provide the right source at the right time and in the best way [32]. Based on the above definition, it can be concluded that u-learning is a learning model that provides learning resources with communication and connectivity that can be accessed anywhere and anytime, thus enabling learning to be in the right place. and time in the right way and made partially possible by the capabilities of digital media.

Yahya and Jalil concluded that there are 5 (five) characteristics of u-learning [19], namely permanency, the information needed will remain unless students deliberately delete it; accessibility, information is always available and can be used at any time; Immediacy, information can be retrieved as soon as possible; Interactivity can interact with lecturers, experts and between students efficiently and effectively through different media; Context-awareness, an environment that can adapt to the real situation of students to provide adequate information for students. Meanwhile, the u-learning characteristics expressed by Jung [31] are Omnipresence (ubiquity), communication and connectivity in the learning environment whenever and wherever; Context Customization (CC), the extent to which u-learning provides useful content based on the learning environment of students; Interactivity (INT), the interaction of technology between learners and technology tools; Self-Directed Learning (SDL), a process in which students take the initiative to find learning needs, declare the learning objectives needed, identify learning resources, select and apply them to learning; Perceived Enjoyment (PE), the extent to which the use of u-learning is considered enjoyable, in addition to any performance consequences of using the system. Virtanen [33] defines the u-learning environment as a learning system with context awareness, interactivity, personalization, and supported with more sophisticated technology components, such as learning management systems (LMS), functional objects, sensing technology, wireless networks, mobile devices, etc.

So it can be concluded that u-learning has the following characteristics:

- Permanency, the information required will remain unless students deliberately delete it.
- Omnipresence (ubiquity), communication and connectivity in the learning environment so that information is always available and can be used at any time.
- Immediacy, information can be retrieved as soon as possible.
- Interactivity, can interact with lecturers, experts among students, and technology tools efficiently and effectively through different media.
- Context-awareness, an environment that can adapt to the real situation of students to provide adequate information based on the learning environment of students.
- Self-Directed Learning (SDL), a process in which students take the initiative to find learning needs, declare the learning objectives needed, identify learning resources, select and apply them to learning.
- Perceived Enjoyment (PE), the extent to which the use of u-learning is considered enjoyable, in addition to any performance consequences of using the system.

2.3 Learning programming concepts

Learning to make a program is divided into two perspectives, namely psychological/educational, and software engineering perspective. The software engineering perspective focuses on expert and professional programmers, frequently working in teams, and how to work on projects effectively. The educational perspective is more on the initial lessons such as programming, mental models, knowledge, and skills needed to create programs, including the basis for good software development practice [34].

Programming courses are concerned with the ability to convert abstract real-world problems into program code that can be interpreted and executed by a computer. The difficulty in describing the flow of program code execution is one of the main factors in the difficulty of understanding programming concepts because students do not / have no experience. To help students understand abstract concepts about programming, it is necessary to provide support in learning such as the visualization program [35], [36].

Learning to program means using a programming language as a medium in which someone describes concepts accurately, formulates algorithms, and solutions. The use of several programming languages can be done simultaneously or separately, it is necessary to understand the programming models that underlie different languages and make informed design choices in languages that support several complementary approaches. Programmers need to learn new programming languages and programming constructs, understand the basic principles of how programming language features are defined, construct, and implement [4], [37]. Students need to open their horizons about the programming process because it is an important part of an introductory programming course, this is not found in traditional teaching materials such as textbooks, lecture notes, whiteboards, slide presentations, and others [13], [38].

Based on the explanation above, it can be concluded that learning to program means learning to solve problems and developing programs with certain appropriate programming languages. The problem that is solved usually has several possible solutions. The program development process starts from describing concepts, formulating algorithms, and reasons for solutions. And that requires better teaching materials than textbooks, lecture notes, whiteboards, or slide presentations.

2.4 Research question

This study answers the following two research questions:

1. How have the students been through the programming learning process?
2. What are the students' perspectives of using u-learning?

3 Methodology/materials

This study aims to describe the current situation so that it can be used as a basis for investigating the necessary facts, therefore descriptive research is conducted. The necessary facts are investigated with data collection carried out through a review of the learning outcomes of students who have taken programming courses embedded in faculty. Apart from past learning data, questionnaires were also distributed using Google Forms. The questionnaire instrument used before distribution was validated first by the instrument expert and then tested by potential respondents to test the readability of the instrument. After the data is collected and the questionnaire is distributed, an analysis and description of the data obtained are carried out to obtain conclusions and results.

3.1 Participants

The data population is students of the Information Technology Faculty from one of the private university in Indonesia. Samples were selected by purposive sampling, from students who had taken programming courses.

3.2 Data collection

The data collected from the faculty is data on student programming scores for 2 (two) study programs, namely Informatics 2017–2018 and Information System 2014–2018. The questionnaire results data are obtained after being distributed and collected for processing and analysis in order to describe the results obtained. Data collection via google form which is distributed to sample students.

3.3 Data analysis

The data score collected is grouped based on the value of the student's end which is obtained from midterm exam scores, assignment scores, and final semester exams. As the standard/limit value is 70 because the value of 70 is the value of criterion B (Good), then it is sorted out which ones are less and which are more than the value of 70. The data from the questionnaire results are calculated as the average response to each statement. Based on the results obtained, the data were analyzed according to each statement.

4 Results and findings

The achievement score of students who have taken programming courses based on data obtained from the faculty in 2014–2018, can be seen in Figure 1 that the scores of students who achieve the minimum a B grade (Good) or more than 70% decrease both in the Information Systems (SI) study program and Information Engineering (IT).

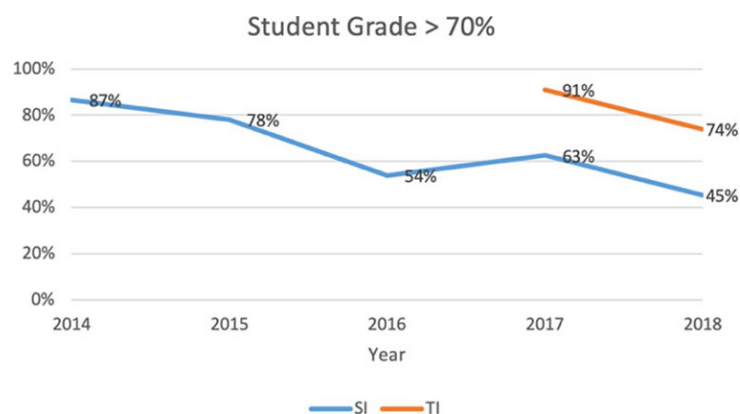


Fig. 1. Graph of achievement of student programming scores

Students' perceptions about programming learning that have been done are shown in Figure 2, the results show that 73.47% of students stated that learning programming was difficult.

Persepsi Mahasiswa tentang Mata Kuliah Pemrograman

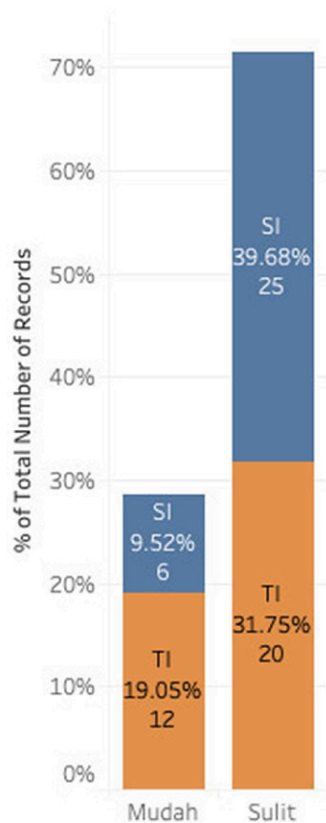


Fig. 2. Students' perceptions of programming courses

An overview of the programming learning process that has taken place can be seen in Table 1. Based on the data collected, it can be seen that learning has been carried out well with a value of 81.31% in accordance with the stages of learning that are good and correct. From the responses given by respondents, the least thing done by lecturers according to students was giving an explanation of the content of the course, student responses were 70.48%. Overall, the students felt that the last thing the lecturers did was in the preliminary and closing activities. Preliminary activities are activities in which the lecturer briefly explains the subject and the benefits and objectives of learning. While the closing activity is a feedback activity and a follow-up to the feedback.

Table 1. Students’ perceptions of the learning activities that have been carried out

Categories	Statement	Response	Percentage
Introduction: The sequence of Activities	The lecturer provides a brief explanation of the content of the course	3,52	70,48%
	The lecturer explains the relevance and benefits of the course material	3,97	79,37%
	The lecturer explains the learning objectives	4,16	83,17%
Presentation: The sequence of Activities	The lecturer explains course material that covers theories, concepts, principles, and procedures	3,98	79,68%
	Lecturers always provide the opportunity to ask questions or submit opinions	4,29	85,71%
	The lecturer provides examples	4,32	86,35%
	The lecturer provides exercises	4,10	81,90%
	The lecturer gives a mastery test of the material	3,83	76,51%
	The lecturer delivers the summary of the lecture	4,14	82,86%
	The lecturer submits a list of technical terms in lectures and their meanings	4,13	82,54%
Closing: The sequence of Activities	Lecturers provide feedback on the tests given	3,97	79,37%
	The lecturer follows up on the feedback given	3,97	79,37%
Learning Content	The material presented by the lecturer is in accordance with the learning objectives stated in the Lesson Plan	4,19	83,81%
	Learning content is in accordance with the learning objectives that were conveyed at the beginning of the meeting	4,16	83,17%
Learning methods	Teaching methods used by lecturers to attract interest in learning	4,08	81,59%
	Media and tools used by lecturers (such as books, presentations, videos, etc.) that have been prepared by the lecturer to assist in learning	4,14	82,86%
	Media and tools used by lecturers that have been prepared by lecturers are easy to obtain.	4,17	83,49%
Media and Learning Tools	the location of the time set in learning in accordance with the material and assignments given	4,06	81,27%
	Average	4,07	81,31%

In terms of learning programming, the biggest difficulty faced by students is the difficulty in finding errors (bugs) from the program being made followed by studying arrays, while making models with flowcharts to solve problems is considered the least difficult. Situations that can help students learn more effectively are situations where there is consultation or discussion with teachers, seniors, or friends. Besides that, studying in groups is also effective in helping learn. Meanwhile, the situation that does not help effective learning is when it comes to learning on your own. In terms of support that can help in learning, according to the students, the thing that helps them the most is

by providing examples of programs besides the explanation of the solution of the given assignment and the material that is easy to find on the web is the support that is considered helpful after the examples given. Correspondingly, according to respondents, the lack of examples of practical use is a factor that causes the failure to learn to program. In detail, the responses from respondents regarding their experience in learning programming can be seen in Table 2.

Table 2. About learning programming

Categories	Statement	Response	Percentage
Difficulty when study programming	Difficult to design a program/make a model (flowchart) to solve certain problems	2,65	53,02%
	Difficult to divide functionality into procedures or functions	3,00	60,00%
	Difficult to understand the basic concepts of programming (data types, identifiers, variables, constants, expressions, and operators)	2,92	58,41%
	Difficult to studying program structures such as sequences, loops, and branching	3,13	62,54%
	Difficult to learn programming language syntax	3,24	64,76%
	Difficult to find errors (bugs) from the program that is made	3,35	66,98%
	Difficult to use a program development environment (programming language)	3,21	64,13%
	Difficult to learn Array	3,27	65,40%
Situations that can help in learning programming more effectively	Practice in the laboratory helps in learning programming more effectively	4,06	81,27%
	Consultation or discussion with teachers, seniors, or friends helps in learning programming more effectively	4,35	86,98%
	Learning in groups helps in learning programming more effectively	4,27	85,40%
	Learning to program yourself helps in learning programming	3,13	62,54%
	Lectures in class help in learning programming more effectively	3,43	68,57%
	Ease of obtaining learning materials helps in learning programming more effectively	3,76	75,24%
The support that can help in learning	Providing program examples can help in learning	4,24	84,76%
	Exercises and assignments can help in learning	3,92	78,41%
	Explanation of the solution to the given task can help in learning	4,16	83,17%
	Textbooks/dictates can help in learning	3,44	68,89%
	Materials that are easy to find on the web can help in learning	4,16	83,17%
	Transparency (presentation) of lecturers helps in learning	3,92	78,41%
	Discussions (Forums) can help in learning	4,05	80,95%

(Continued)

Table 2. About learning programming (*Continued*)

Categories	Statement	Response	Percentage
The factors that cause the failure to study programming	Few examples show the practical use	4,57	91,43%
	The focus of the syllabus is too much on theory	3,86	77,14%
	The scope of the syllabus per semester is too broad	3,51	70,16%
	Students are less interested in learning	3,48	69,52%
	The learning environment is not conducive	3,29	65,71%
	Presentation of lecturers and their attention to students	3,40	67,94%
	Less effective teaching methodology	3,44	68,89%
	The computer provided in the laboratory is not functioning properly	3,33	66,67%

Regarding the perception of the respondent’s readiness in terms of supporting u-learning technology in learning programming, the results obtained are shown in Table 3. In general, based on the results obtained, respondents agree that u-learning technology supports learning programming with an average approval level of 4, 08 (81.52%). The use of technology in supporting learning has the highest level of approval while moving places is the least approved thing in the learning process even though the level of agreement is still 70%. This shows that u-learning is ready to be applied in learning in higher education, especially for programming learning.

Table 3. Support of u-learning technology in learning programming

Statement	Response	Percentage
Learning resources that can be obtained and accessed easily will help in learning	4,06	81,27%
Support system (learning management system) is needed for learning	4,00	80,00%
Personal learning (according to personal needs) is required in learning	3,81	76,19%
Active support in learning is needed	4,08	81,59%
It is necessary to move places in the learning process	3,51	70,16%
Learning in/from the real world (work environment) is required	4,14	82,86%
Interaction and communication with lecturers, experts, and other students can help learn	4,27	85,40%
Learning resources can be reused after completing courses (anytime)	4,27	85,40%
The use of technology can support learning	4,54	90,79%
Average	4,08%	81,52%

5 Conclusion

Learning programming has been a challenge for informatics education from the past and continues to be a challenge today. The development of information technology and the birth of generation Z who are very familiar with the digital world. Ubiquitous learning is a learning paradigm that provides learning resources (networks) with

communication and connectivity that can be accessed at any location at any time, thus enabling learning to be right at the right place and time in the right way and partly made possible by the capabilities of digital media. This fits with the current generation. Based on the discussion that has been done, this study concludes that the learning activities that have been carried out well by the teachers. Students are also ready to apply ubiquitous learning technology.

6 References

- [1] Agung, B. (2017, July 28). *Indonesia Darurat Tenaga Programmer*. Teknologi. <https://www.cnnindonesia.com/teknologi/20170728094848-185-230919/indonesia-darurat-tenaga-programmer>
- [2] Triwijanarko, R. (2019, November 1). *Indonesia Masih Butuh Banyak Tenaga Programmer*. *Marketeers—Majalah Bisnis & Marketing Online—Marketeers.Com*. <https://marketeers.com/indonesia-masih-butuh-banyak-tenaga-programmer/>
- [3] Senjaya, A. (2015, April 3). *Programmer dan Pertumbuhan Startup Teknologi Indonesia*. LinkedIn Pulse. <https://www.linkedin.com/pulse/programmer-dan-pertumbuhan-startup-teknologi-andreas-senjaya>
- [4] ACM Computing Curricula Task Force (Ed.). (2013). *Computer Science Curricula 2013: Curriculum Guidelines for Undergraduate Degree Programs in Computer Science*. ACM, Inc. <https://doi.org/10.1145/2534860>
- [5] Tan, P. H., Ting, C. Y., & Ling, S. W. (2009). Learning Difficulties in Programming Courses: Undergraduates' Perspective and Perception. *2009 International Conference on Computer Technology and Development, 1*. <https://doi.org/10.1109/ICCTD.2009.188>
- [6] Vujošević-Janičić, M., & Tošić, D. (2008). The role of programming paradigms in the first programming courses. *The Teaching of Mathematics, 21*, 63–83.
- [7] Robins, A. (2015). The ongoing challenges of computer science education research. *Computer Science Education, 25*(2), 115–119. <https://doi.org/10.1080/08993408.2015.1034350>
- [8] Le, N.-T. (2016). A Classification of Adaptive Feedback in Educational Systems for Programming. *Systems, 4*(2), 22. <https://doi.org/10.3390/systems4020022>
- [9] Xinogalos, S. (2014). Designing and deploying programming courses: Strategies, tools, difficulties and pedagogy. *Education and Information Technologies, 21*(3), 559–588. <https://doi.org/10.1007/s10639-014-9341-9>
- [10] Nuutila, E., Törmä, S., Kinnunen, P., & Malmi, L. (2008). Learning Programming with the PBL Method—Experiences on PBL Cases and Tutoring. In *Reflections on the Teaching of Programming*. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-77934-6_5
- [11] Bennedsen, J., & Caspersen, M. E. (2007). Failure rates in introductory programming. *ACM SIGcSE Bulletin, 39*(2), 32–36. <https://doi.org/10.1145/1272848.1272879>
- [12] Watson, C., & Li, F. W. B. (2014). Failure rates in introductory programming revisited. *Proceedings of the 2014 Conference on Innovation & Technology in Computer Science Education—ITiCSE '14*, 39–44. <https://doi.org/10.1145/2591708.2591749>
- [13] Bennedsen, J., Caspersen, M. E., & Kölling, M. (Eds.). (2008). *Reflections on the teaching of programming: Methods and implementations*. Springer. <https://doi.org/10.1007/978-3-540-77934-6>
- [14] Malat, L., Vostok, T., & Eveland, A. (2017). *Gen Z Exploring Middle And High Schoolers' Expectations For Higher Education*. <https://www.bncollege.com/wp-content/uploads/2018/09/Gen-Z-Report.pdf>

- [15] Alhassan, R. (2016). Mobile Learning as a Method of Ubiquitous Learning: Students' Attitudes, Readiness, and Possible Barriers to Implementation in Higher Education. *Journal of Education and Learning*, 5(1), 176. <https://doi.org/10.5539/jel.v5n1p176>
- [16] Wang, M., Shen, R., Novak, D., & Pan, X. (2009). The impact of mobile learning on students' learning behaviours and performance: Report from a large blended classroom. *British Journal of Educational Technology*, 40(4), 673–695. <https://doi.org/10.1111/j.1467-8535.2008.00846.x>
- [17] Cheon, J., Lee, S., Crooks, S. M., & Song, J. (2012). An investigation of mobile learning readiness in higher education based on the theory of planned behavior. *Computers & Education*, 59(3), 1054–1064. <https://doi.org/10.1016/j.compedu.2012.04.015>
- [18] Aljohani, N. R., Davis, H. C., & Loke, S. W. (2012). A comparison between mobile and ubiquitous learning from the perspective of human–computer interaction. *International Journal of Mobile Learning and Organisation*, 6(3–4), 218–231. <https://doi.org/10.1504/IJMLO.2012.050046>
- [19] Yahya, S., Ahmad, E. A., & Jalil, K. A. (2010). The definition and characteristics of ubiquitous learning: A discussion. *International Journal of Education and Development Using Information and Communication Technology*, 6(1), 1.
- [20] Indrajit, R. E. (2016). *Membangun Ekosistem Ubiquitous Learning di Indonesia*. <http://pro-fekoindrajit.blogspot.com/2018/11/membangun-ekosistem-ubiquitous-learning.html>
- [21] Spector, J. M. (2015). *Foundations of Educational Technology: Integrative Approaches and Interdisciplinary Perspectives*. Routledge. <https://doi.org/10.4324/9781315764269>
- [22] Smaldino, S. E., Lowther, D. L., Russell, J. D., & Mims, C. (2015). *Instructional technology and media for learning* (Eleventh Edition). Pearson.
- [23] Schunk, D. H. (2012). *Learning theories: An educational perspective* (6th ed). Pearson.
- [24] Reigeluth, C. M. (Ed.). (2009). *Building a common knowledge base*. Routledge.
- [25] Suparman, A. (2014). *Desain Instruksional Modern Panduan Para Pengajar dan Inovator Pendidikan* (Keempat). Penerbit Erlangga.
- [26] Kilbane, C. R., & Milman, N. B. (2014). *Teaching models: Designing instruction for 21st century learners*. Pearson.
- [27] Joyce, B. R., Weil, M., & Calhoun, E. (2015). *Models of teaching* (Ninth edition). Pearson.
- [28] Cope, B., & Kalantzis, M. (2009). Ubiquitous learning: An agenda for educational transformation. *Ubiquitous Learning*, 1–14.
- [29] Zhang, J.-P. (2008). Hybrid learning and ubiquitous learning. *International Conference on Hybrid Learning and Education*, 250–258. https://doi.org/10.1007/978-3-540-85170-7_22
- [30] Hwang, G.-J., Tsai, C.-C., & Yang, S. J. H. (2008). Criteria, Strategies and Research Issues of Context-Aware Ubiquitous Learning. *Journal of Educational Technology & Society*, 11(2), 81–91. <https://doi.org/10.2307/jeductechsoci.11.2.81>
- [31] Jung, H.-J. (2014). Ubiquitous Learning: Determinants Impacting Learners' Satisfaction and Performance with Smartphones. *Language Learning & Technology*, 18(3), 97–119.
- [32] Guabassi, I. E., Achhab, M. A., Jellouli, I., & Mohajir, B. E. E. (2016). Towards adaptive ubiquitous learning systems. *International Journal of Knowledge and Learning*, 11(1), 3–23.
- [33] Virtanen, M. A., Haavisto, E., Liikanen, E., & Kääriäinen, M. (2018). Ubiquitous learning environments in higher education: A scoping literature review. *Education and Information Technologies*, 23(2), 985–998. <https://doi.org/10.1007/s10639-017-9646-6>
- [34] Robins, A., Rountree, J., & Rountree, N. (2003). Learning and Teaching Programming: A Review and Discussion. *Computer Science Education*, 13(2), 137–172. <https://doi.org/10.1076/csed.13.2.137.14200>

- [35] Isa, N. A. M., & Derus, S. R. M. (2017). Students experience in learning Fundamental Programming: An analysis by gender perception. *Advanced Journal of Technical and Vocational Education*.
- [36] Mustapha, I., Van, N. T., Shahverdi, M., Qureshi, M. I., Khan, N. Effectiveness of Digital Technology in Education During COVID-19 Pandemic. A Bibliometric Analysis. *International Journal of Interactive Mobile Technologies*. 2021 Aug 1;15(8). <https://doi.org/10.3991/ijim.v15i08.20415>
- [37] Qureshi, M. I., Khan, N., Raza, H., Imran, A., Ismail, F. Digital Technologies in Education 4.0. Does it Enhance the Effectiveness of Learning? A Systematic Literature Review. *International Journal of Interactive Mobile Technologies*. 2021 Apr 1;15(4). <https://doi.org/10.3991/ijim.v15i04.20291>
- [38] Khan, N., Qureshi, M., Mustapha, I., Irum, S., Arshad, R. A systematic literature review paper on online medical mobile applications in Malaysia.

7 Authors

Lucia Sri Istiyowati, Perbanas Institute, Universitas Negeri Jakarta, Indonesia.
E-mails: LuciaSriIstiyowati_7117157702@mhs.unj.ac.id; lucia.istiyowati@perbanas.id.
Zulfiati Syahrial, Universitas Negeri Jakarta, Indonesia.
Suyitno Muslim, Universitas Negeri Jakarta, Indonesia.

Article submitted 2021-06-04. Resubmitted 2021-07-22. Final acceptance 2021-07-23. Final version published as submitted by the authors.