Development of Interactive Learning Application for Basic Programming Based on Technological Pedagogical Content Knowledge Framework

Daniel Yulius Soetjipto[1], Hendra Dinata[2*], Melissa Angga[3], Jovan Adriel Widjaja[4]

[1,2,3,4]Informatics Department, Universitas Surabaya, Surabaya, East Java, Indonesia
E-mail: [1]daniel.yulius@gmail.com, [2*]hdinata@staff.ubaya.ac.id, [3]melissa@staff.ubaya.ac.id, [4]jovanaw@gmail.com

(Received: 30 Sep 2024, revised: 15 Nov 2024, accepted: 18 Nov 2024)

Abstract

Information technology students must take Algorithms and Programming. Research shows that 28% of US students fail their basic programming subject, which is essential to mastering programming. In line with the previous study, 39% of students in the Informatics Engineering department's basic programming course at campus X in the odd semester of 2022/2023 failed the course. The learning process should be able to integrate technology into it. An interactive learning application was developed utilizing the Technological Pedagogical and Content Knowledge (TPACK) framework, incorporating a pedagogical paradigm in its design through simulation elements and animated visuals. Through an extensive design, this learning application enhances student engagement by 78.3%, encouraging continued utilization in their educational process. The trial involving the group of students utilizing this application revealed that 5 out of 34 students failed the course, in contrast to 7 out of 33 students from the group that studied without the application.

Keywords: Learning Application, TPACK, Basic Programming, Simulation.

I. INTRODUCTION

The Algorithms and Programming course is mandatory for students pursuing a degree in information technology. Regrettably, research findings indicate that 28% of students do not succeed in passing their introductory programming courses, which are considered fundamental in the learning of programming [1]. Consistent with the findings of the previously mentioned research conducted in the United States, the study conducted on the introductory programming course within the Informatics Engineering department at campus X during the odd semester of 2022/2023 revealed that 39% of participants in this course obtained grades that fell below the established minimal passing threshold. Campus X mandates a minimum grade of C as a threshold for successfully completing this foundational course, as it is essential for students to acquire fundamental competencies that will enable the students to comprehend further programming courses in future semesters.

A supplementary interview was conducted with 20 Informatics Engineering students who still needed to pass this introductory programming course. The interview aimed to ascertain the specific challenges encountered by the students that led to their failure in this course. Based on the interview findings, 17 participants expressed the opinion that the reading material was excessively technical, which posed challenges in comprehending the content while working on tasks. In short, the students needed help understanding the current textbook's content. Furthermore, other factors were identified that hindered students' ability to work on topics autonomously due to their lack of prior programming experience and the absence of study partners.

Guan et al. [2] stated that to enhance the quality of student learning, the utilization of educational applications is advisable. This statement is aligned with a study conducted on elementary students to learn basic English using a multimedia-based learning application that could improve their understanding of English subjects in their school [3]. A study conducted by Haryadi and Pujiastuti [4] revealed that the inclusion of simulation aspects in learning applications might enhance student comprehension by as much as 37%. Studies indicate that the utilization of simulation enables students to more effectively comprehend system behavior compared to solely relying on textbook-based learning. Additionally, algorithm and programming learning programs that include animated movies can significantly enhance student comprehension by allowing them to replay the films multiple times [5]. Furthermore, Ghofur and Youhanita [6] underscored the significance of educational applications that might help enhance students' enthusiasm to learn. Hence, interactive

DOI: 10.34148/ teknika.v14i1.1084

elements are crucial to improve students' motivation. Conversely, Limanto and Dinata highlighted the significance of incorporating practice question elements in a learning application to enable students to practice and comprehend questions independently [7].

Upon examining the issues raised by the students, it is evident that merely providing students with learning materials that fail to create a sense of comfort and motivation for independent learning is insufficient. Students must experience the perception of a lecturer or learning companion despite being in solitude. TPACK is a pedagogical framework that assists educators in the efficient integration of technology into their instructional methodologies. Students will benefit from incorporating the TPACK framework into an interactive learning application. It is anticipated that students will experience the presence of a lecturer as they engage in the learning process, which will be achieved through the interactive and simulation features [8]. This study addresses the lack of research in this area by utilizing the Technological Pedagogical and Content Knowledge (TPACK) framework to analyze an interactive learning application. TPACK is a pedagogical paradigm that guides educators on effectively integrating technology into their teaching practices. Therefore, this study will be undertaken to address the subsequent research questions:

1. How to help improve the understanding of basic programming course for higher education students?
2. How does the TPACK framework play a role in efforts to improve the learning process of basic programming course for higher education students?

II. LITERATURE REVIEW

A. Basic Programming

Programming is the implementation of logic to perform computational operations and functionality [9]. It is about programming, which is the basis of competency that students in the field of information technology must learn. Programming for beginners could be divided into six essential topics [10]. These six essential topics are:

1. Data and Operators
   It consists of variables, constants, and operators for arithmetic expressions.
2. Conditions
   Statements that allow the execution of different blocks of code based on specific criteria
3. Functions
   Blocks of reusable and organized code that usually perform a single, related action
4. Repetition
   Lines of code will be run multiple times
5. Array
   An array is a data structure consisting of a collection of elements (values or variables) identified by at least one array index or key.
6. Strings and Files
   A sequence of characters, either as a literal constant or as some variable/

B. Technological Pedagogical and Content Knowledge (TPACK)

TPACK is a pedagogical paradigm that guides educators on effectively integrating technology into their teaching practices [8]. Hamam & Hysaj [11] in his study has shown that the use of the TPACK framework can also be used in higher education. The study shows that the TPACK framework has a positive impact on the learning process carried out for the higher education level. A further research in Indonesia examined the application of the TPACK framework in higher education [12]. The study's results indicated that TPACK offers advantages for educators in facilitating the teaching and learning process through the integration of available technology.

Koehler et al. [13] categorize the TPACK framework into three primary components: Technological Knowledge (TK), Content Knowledge (CK), and Pedagogical Knowledge (PK). Furthermore, four additional sub-components will be generated due to the interaction between these three components. Those four sub-components are:

1. Pedagogical Content Knowledge (PCK)
   PCK is the result of interaction between CK and PK, suggesting that distinct materials necessitate distinct teaching methods.
2. Technological Content Knowledge (TCK)
   TCK is the result of interaction between TK and CK, which suggests that it is essential that educators are not only acquainted with the material they instruct but also know how to apply technology to the learning material being delivered [14].
3. Technological Pedagogical Knowledge (TPK)
   TCK is the result of interaction between TK and PK, which suggests that technology can be beneficial for specific pedagogical purposes and assist instructors in selecting the tools that are most pertinent to a specific pedagogical approach. In other words, technology can facilitate the execution of specific classes and offer new methods and venues to pursue.
4. Technological Pedagogical Content Knowledge (TPACK)
   TPACK is a comprehensive approach that integrates all the previously mentioned elements, with a particular emphasis on developing technology that is uniquely applicable to address the pedagogical requirements of teaching content or material in a specific context.

Figure 1 below shows how the components interact in the TPACK framework. The inclusion of these components in the learning application is anticipated to enhance student motivation by providing appropriate learning content, creating a sense of having a learning companion through simulations that offer step-by-step instructions like a classroom lecturer, and ultimately enhancing student comprehension.

DOI: 10.34148/teknika.v14i1.1084

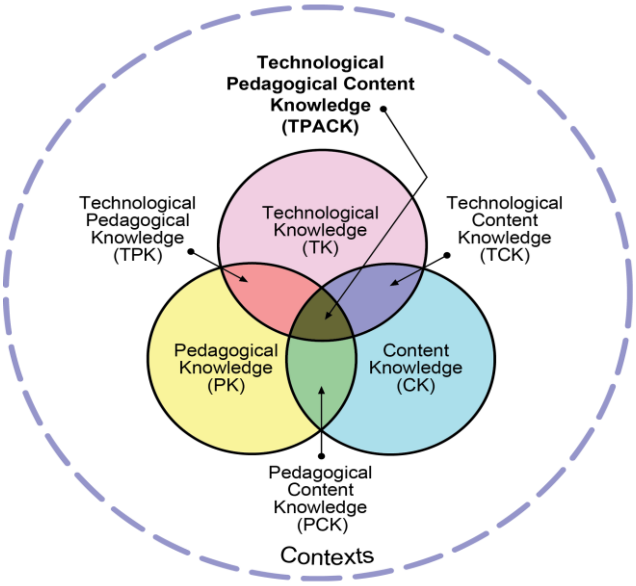


Figure 1. TPACK Components
(source: http://tpack.org/)

III. METHODOLOGY

The design and development of interactive learning applications for basic programming learning are carried out following the Software Development Life Cycle (SDLC) steps using the waterfall model.

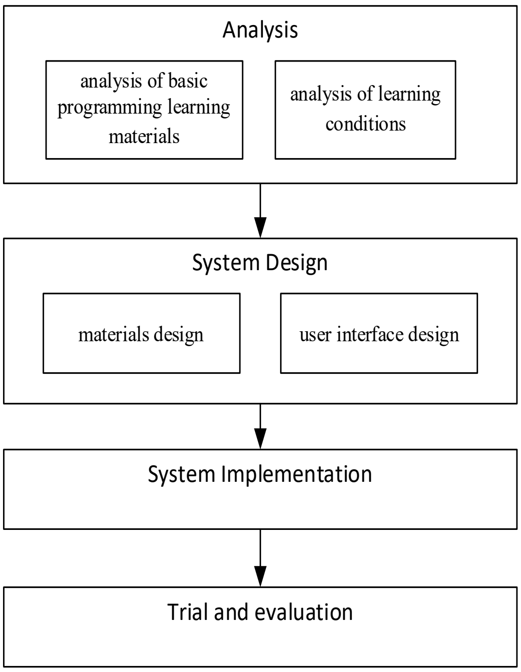


Figure 2. Research Methodology

Figure 2 above shows the flow of this research method. These stages consist of:

1. Analysis of basic programming learning materials and analysis of basic programming learning conditions among first-semester students of the Informatics Engineering study program to determine the difficulties faced by students,
2. Design of basic programming learning application needs that are adjusted to the TPACK framework,
3. Implementation and
4. Trial and evaluation

Specifically, this trial and evaluation stage is carried out for one semester for students who failed the Algorithm and Programming course in the previous first semester. In this second semester, these students retake the Algorithm and Programming course and are divided into two groups: students who repeat while learning using the application and those who repeat without using the application. A survey will be conducted to assess the utilization and effectiveness of this application, as well as to measure the impact it has achieved.

IV. RESULT AND DISCUSSION

Based on an analysis of interviews with two lecturers of Algorithm and Programming classes, the learning material covered in the Algorithm and Programming course spans foundational concepts, including basic algorithms, as well as programming topics such as variables, input/output, conditionals, loops, arrays, and methods. Among those subjects, the material that students found most challenging includes loops and methods. The difficulty arises from the lack of logical reasoning and computational thinking skills among students. However, this finding slightly differs with the results of a student survey, which identified the top three most challenging topics as arrays, methods, and loops.

Another key finding from the analysis process from the lecturer is that a lack of logical and computational thinking is a major barrier preventing students from understanding how the program works. This result aligns with the findings from the student survey, which indicated that a lack of logical thinking is the primary barrier preventing students from understanding programming. Additionally, students who struggle with the subject are often too shy to ask the lecturer for help, which exacerbates the issue over time.

Both lecturers found that using examples related to students' daily lives can significantly enhance their understanding. Additionally, providing exercises along with clear explanations on how to solve them helps students grasp the topic more effectively.

The TPACK framework is utilized in the design process of the learning application system. This application is designed on the Android platform, allowing students to utilize it flexibly. This flexibility will benefit students by enabling them to continue their education beyond the classroom. The integration of TPACK with basic programming learning applications, covering the design of basic programming media, material preparation, instruction, and technology utilization, can ideally enhance fundamental programming education. Table 1 below describes the application of the TPACK framework in the design of a basic programming education application system.

DOI: 10.34148/ teknika.v14i1.1084

Table 1. Implementation of TPACK Framework

TPACK Component	Implementation
Content Knowledge (CK)	Utilizing programming textbooks as reference resources for the basic programming learning applications.
Pedagogical Knowledge (PK)	1. Offer illustrations to enhance users' comprehension of the topic. 2. Impose a penalty in the form of waiting time before the user can attempt the practice questions again if they deemed failed in their previous attempt. 3. Implement the chunking principle by subdividing the material into many segments.
Technological Knowledge (TK)	Utilizing technology through mobile applications as learning media for basic programming.
Pedagogical Content Knowledge (PCK)	1. Organizing materials that comply with user requirements derived from problem analysis that lacks basic programming skill. 2. Integrating graphics and programming simulations in each resource
Technological Content Knowledge (TCK)	Develop a basic C# programming simulation system for each topic.
Technological Pedagogical Knowledge (TPK)	1. Develop a practice question system within a fundamental programming education application. 2. Develop a C# programming simulation system as a platform for practicing the learned information. 3. Develop additional interactive graphics within the program to enhance user comprehension of the topic. 4. Implement a waiting period within the application for users who have not successfully completed the practice questions. 5. Establish a system that prevents users from advancing to subsequent materials until they have been officially deemed to have passed the content previously studied.

In terms of content knowledge, the learning material is drawn from the entire Algorithm and Programming course, covering variables and data types, conditionals, loops, arrays, and methods. Emphasis is placed on loops, arrays, and methods, as these topics were identified by students as the most challenging.

For pedagogical knowledge, the approach uses everyday items as analogies to explain programming concepts. For example, a fruit basket is used to illustrate variables, and a machine serves as an analogy for an operator.

For technological knowledge, the application will be developed as a mobile app. All data will be stored locally on the device, eliminating the need for an internet connection.

In terms of pedagogical content knowledge, the main issue identified is the lack of logic and computational thinking. As suggested by the lecturers, this problem requires regular practice, so a set of exercises has been provided. If a student fails an exercise, they will be given a 30-minute break before being allowed to try again. This approach applies the pedagogical concept of 'pausing for effect' to help reinforce learning. Moreover, to enhance student understanding, the application would incorporates graphics and simulations for each topic.

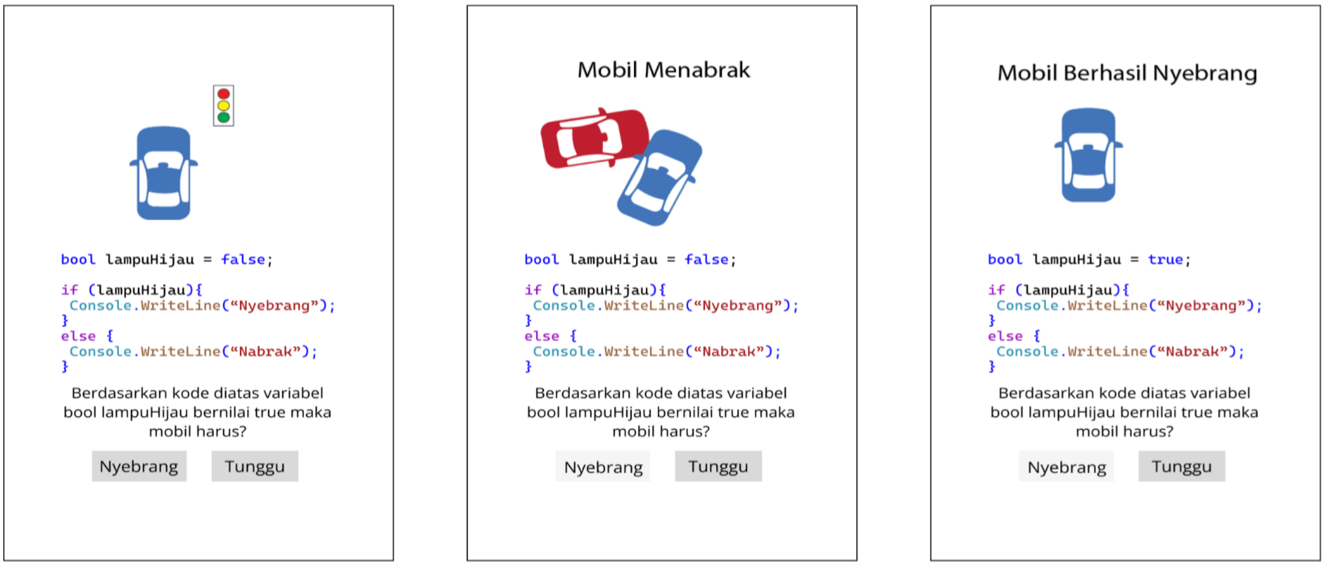
Technological content knowledge is applied through simulations that help students visualize the effects of adding or altering specific lines of code. This approach aids in build students' logical and computational thinking, particularly for complex topics such as arrays, loops, and methods, which students often find challenging.

The proposed technological pedagogical content involves creating an application that incorporates practice sessions within the programming lessons, using graphics and simulations to enhance understanding. Additionally, the system prevents students from progressing to more complex material unless they successfully complete the simpler first, ensuring to avoid frustration. At the same time, the system includes a waiting period, preventing students from repeatedly attempting the same exercise without a break.

This learning application includes six fundamental concepts as articulated by Busbee and Braunschweig [10]. Each topic contains three parts that help to facilitate the learning process. The three parts are 1) material explanation, 2) quiz, and 3) simulation. The material explanation part will present animated pictures, explanations, and program code samples. In the quiz part, the application will randomly present a selection of questions, each featuring animated graphics and explanations for the respective answers. Lastly, there is a part of the simulation in which the application presents a snippet of program code, and the user must complete a portion. This approach provides the user with an authentic programming application environment since it includes a simulation of what arises from the execution of the previously completed program code.

Figure 3 below shows the application screenshot for the second part of the Conditional topic. In this second part, the user chooses the best answer following the question given. Figure 3 shows that screenshot (a) is the initial condition before the user submits the answer. The user is given a boolean variable and must submit the correct answer based on the variable's value. Screenshot (b) previews the animation when the user submits the wrong answer, while screenshot (c) displays the animation when the user submits the correct answer.

DOI: 10.34148/teknika.v14i1.1084



(a) (b) (c)

Figure 3. Screenshot for the Quiz of the Conditional Topic

Figure 4 below shows the simulation part of the Repetition topic. There is a small section on the screen, namely the value section of a given variable, where the user can determine the desired value (see the red arrow in Figure 4). The application will play an animation of a car traversing the circuit repeatedly, corresponding to the value inputted by the user for the specified variable. The car's recurring animation illustrates the program's conditions constructed with the For syntax. Consequently, the user will more readily comprehend the significance of the Repetition characteristic on this topic. This strategy aligns with a previous study that revealed that animation in learning applications is a supporting factor for success in the learning process using applications [15].

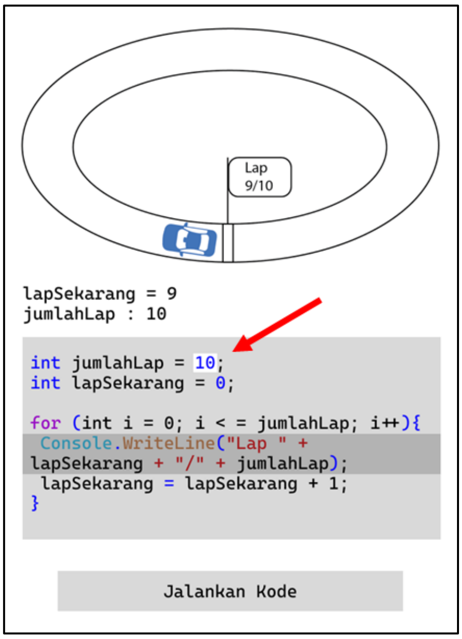


Figure 4. Screenshot for the Simulation of the Repetition Topic

Table 2 below shows the statistics of the final score for the Algorithm and Programming subject in the even semester of 2024. There are 67 students divided into two parallel classes: A and B. The class A is a group of 34 students who utilize this learning application while taking this subject. Meanwhile, the class B is a group of 33 students who do not use the application. The threshold score to pass this subject is 55.

All students in both parallel classes possess equivalent proficiency in this Algorithm and Programming course. These students are those who did not pass the prior semester and must retake the Algorithm and Programming course. This test can be conducted by engineering learning in both student groups, based on the assumption of comparable skill levels.

Table 2. Statistics of the Final Score

	Class A	**Class B**
Students Number	34	33
Mean	68.1765	65.9394
Min	36	38
Max	89	88
SD	12.5522	11.1628
Below Threshold	5	7

Upon completion of the learning process engineering at the end of the semester, the final scores of the two student groups were compared. The mean values of both classes indicate a slight disparity. The disparity in mean value is only 2.2371 points. The independent T-Test comparison yielded a p-value of 0.4527 for the two classes. Consequently, since the p-value exceeds 0.05, it may be concluded that the values of the two classes are not significantly different.

Despite the lack of significant score disparities between the two classes, it is noteworthy that class A possesses a marginally better mean value and has fewer students scoring below the threshold. Interestingly, interviews with the five failed students revealed their admission of never utilizing this

DOI: 10.34148/ teknika.v14i1.1084

learning application in their learning process. Consequently, a survey was conducted to examine further advantages gained by these pupils through the utilization of learning applications. The findings of this survey are presented in Table 3 below.

Table 3. Survey Result by Learning Application Users

Question	Result
I consistently utilize this educational tool in my learning process	78.3%
I feel more confident when facing exams after using this learning application	82.1%
This learning application helps me comprehend the lessons	84.5%
This learning application is easy to operate	90.4%
Utilizing this learning application feels like having a companion for studying.	80.1%
The simulation part with animated images in this learning application enhances my comprehension of the study material	88.2%
After utilizing this learning application, I can give precise answers to the quiz feature	80.1%

The survey employed a 1-5 Likert scale format and involved 34 students from class A. At the conclusion of the survey, students were given the chance to compose open-ended statements regarding their experiences with this learning application. The students exhibited a highly positive response to the presence of this learning application. Students regularly employ this application while they are studying outside the classroom. The students discovered that this learning application offered advantages in their learning process. This application's animation and simulation features were deemed highly beneficial for students in enhancing their comprehension of the course content. These features align with other research indicating that animation and simulation significantly enhance learning applications, supporting the application's intended objectives [3], [7], [15].

One respondent claimed that the presence of this animation enhanced his comprehension of the topic of Repetition more effectively than just listening to his lecturer's explanation in class. *"I feel that the animation of the car moving repeatedly actually makes me understand what the Repetition material means."* Another student also expresses the same thing about the beneficiary of the animation. *"I hope the lecturer can also show animations like this when teaching in class."*

This study implies that students require tools to facilitate their basic programming learning process. Interactive learning software can enhance their self-confidence. Students need not consistently depend on the presence of their instructors. Learning applications with simulation features offer students an interactive experience similar to having peers in their learning process, as these applications can respond to student inputs and provide feedback. Consequently, through this feedback, students learn to recognize that every output generated is contingent upon the input parameters supplied.

Conversely, the campus might establish the presence of this learning application as an emerging standard. Lecturers in basic programming courses should actively promote using this learning application as a supplementary resource for students studying beyond the classroom to minimize the failure rate in this course.

V. CONCLUSION

This study highlights the importance of an interactive learning application designed to support students who face challenges with fundamental programming concepts. The application, which was built using the TPACK (Technological Pedagogical Content Knowledge) framework, proved to be effective in emphasizing the integration of technology in a way that mirrors the instructional support typically provided by teachers and peers while allowing students to learn independently. By incorporating simulation tools and animated visualizations, the application achieves its goals to foster greater student engagement, boost confidence, and improve learning outcomes. The result is a notable reduction in failure rates, highlighting the effectiveness of interactive, technology-enhanced learning environments for programming education.

Having said that, while the significantly higher pass rates among students using the application suggest its positive impact, the disparity in average marks between students using the application and those not using it is not substantial. This indicates the need for further research to identify which specific pedagogical elements are most effective in supporting students' understanding of algorithms and programming.

REFERENCES

[1] J. Bennedsen and M. E. Caspersen, "Failure rates in introductory programming," *ACM Inroads*, vol. 10, no. 2, pp. 30–36, Apr. 2019, doi: 10.1145/3324888.

[2] N. Guan, J. Song, and D. Li, "On the Advantages of Computer Multimedia-aided English Teaching," *Procedia Comput. Sci.*, vol. 131, pp. 727–732, 2018.

[3] H. Dinata and E. Evelyn, "Rancangan Aplikasi Pembelajaran Bahasa Inggris Berbasis Multimedia Untuk Siswa Kelas 4 Sekolah Dasar," *BINA Insa. ICT J.*, vol. 9, no. 1, pp. 63–72, 2022.

[4] R. Haryadi and H. Pujiastuti, "PhET simulation software-based learning to improve science process skills," *J. Phys. Conf. Ser.*, vol. 1521, no. 2, p. 022017, Apr. 2020, doi: 10.1088/1742-6596/1521/2/022017.

[5] R. W. Arifin, "Media Pembelajaran Berbasis Video Animasi Pada Mata Kuliah Logika Dan Algoritma 1," *BINA Insa. ICT J.*, vol. 4, no. 1, pp. 83–94, 2017, [Online]. Available: https://ejournal-binainsani.ac.id/index.php/BIICT/article/view/827

[6] A. Ghofur and E. Youhanita, "Interactive Media Development to Improve Student Motivation," *IJECA (International J. Educ. Curric. Appl.*, vol. 3, no. 1, p. 1, Apr. 2020, doi: 10.31764/ijeca.v3i1.2026.

[7] S. Limanto and H. Dinata, "Aplikasi Multimedia Untuk Media Pembelajaran Menggambar Grafik Fungsi Dengan Operasi Penskalaan, Pergeseran, dan

Pencerminan," *Teknika*, vol. 7, no. 2, pp. 102–107, Nov. 2018, doi: 10.34148/teknika.v7i2.136.

[8] M. J. Koehler and P. Mishra, "Introducing Technological Pedagogical Content Knowledge," in *AACTE Committee on Innovation and Technology (Eds.)*, New York: Routledge, 2008, pp. 3–29.

[9] M. Rouse, "Programming." Accessed: Jan. 19, 2024. [Online]. Available: https://www.techopedia.com/definition/13128/programming

[10] K. L. Busbee and D. Braunschweig, *Programming Fundamentals: A Modular Structured Approach*, 2nd ed. Creative Commons Attribution ShareAlike, 2018.

[11] D. Hamam and A. Hysaj, "Technological Pedagogical and Content Knowledge (TPACK): Higher Education Teachers' Perspectives on the Use of TPACK in Online Academic Writing Classes," in *Stephanidis, C., Antona, M., Ntoa, S. (eds) HCI International 2021 - Posters*, Communications in Computer and Information Science, vol 1421. Springer, Cham, 2021. doi: https://doi.org/10.1007/978-3-030-78645-8_7.

[12] A. N. Astarina, M. L. Sujatna, and H. Heryono, "Technological Pedagogical Content Knowledge (TPACK): The Readiness of Higher Education Educators Facing The Digital Transformation In Teaching And Learning," *Prof. J. English Educ.*, vol. 7, no. 5, pp. 1153–1162, 2024.

[13] M. J. Koehler, P. Mishra, M. Akcaoglu, and J. Rosenberg, "The Technological Pedagogical Content Knowledge Framework for Teachers and Teacher Educators," in *ICT Integrated Teacher Education Models*, 2013.

[14] P. Mishra and M. J. Koehler, "Technological Pedagogical Content Knowledge: A Framework for Teacher Knowledge," *Teach. Coll. Rec. Voice Scholarsh. Educ.*, vol. 108, no. 6, pp. 1017–1054, Jun. 2006, doi: 10.1111/j.1467-9620.2006.00684.x.

[15] T. Adelia and S. Limanto, "Pemanfaatan Multimedia Sebagai Media Pembelajaran Untuk Meningkatkan Pemahaman Siswa Terhadap Materi Fisika Elektromagetik," *Teknika*, vol. 6, no. 1, pp. 35–42, 2017.