



# PRELIMINARY ANALYSIS OF EXPERIMENTAL RESULTS USING DRAG-AND-DROP AND LOG DATA ON PSEUDOLEARN IN LEARNING LOGIC PROGRAMMING

Vivin Ayu Lestari<sup>1</sup>, Farid Angga Pribadi<sup>2</sup>, Eka Larasati Amalia<sup>3</sup>, Ardila Lukita Sari<sup>4</sup>, Eksa Lailia Maulidina<sup>5</sup>,

<sup>1,2,3,4</sup> Information Technology Department, State Polytechnic of Malang, Malang-Indonesia

**Abstract.** Logical and systematic thinking to complete case studies is needed so that the coding structure developed is easily understood by the system. But in practice there are still many students have difficulty understanding the concept of programming algorithms. PseudoLearn is an application that is used as a programming logic learning that models case studies into pseudo code. PseudoLearn uses a drag and drop approach by reconstructing a pseudo code algorithm that is equipped with log data so that it can track student activities and abilities. The hypothesis developed for this study is that students who use the PseudoLearn application get a better understanding than using conventional methods. Thirty-nine students involved in the experiment from the Department of Informatics Engineering, State Polytechnic of Malang. In the first stage, an initial test was carried out which was used as the basis for dividing the groups. Students were divided into 3 groups are control, experiment 1, and experiment 2. ANOVA statistical analysis was used to ensure that the groups used had the same abilities. The experimental results are in accordance with the hypothesis, where the experimental group 1 and 2 exceed the control, besides that it was also found that there was an increase in the pre-test to post-test score of the experimental group which was better than the control. Another interesting thing that was found from the log data analysis is that novice students tend to have more steps to get the right answer and have a longer time learning programming logic.

**Keywords:** ANOVA, Drag and Drop, Log Data, Programming Algorithm, Pseudocode

## 1 Introduction

Java programming was first developed in 1991 in California by James Gosling with the initial goal of a stand-alone platform language for building household electronic devices [1]. One of the advantages of the Java programming language is that it is compatible to run on all platforms that compile byte code [1]. This means that when

using java, it is necessary to install the JVM (Java Virtual Machine) platform to run the application. Java programming is an object-oriented programming language that is very portable to use [2]. Programming is very important if we have a career in IT [3]. Programming is challenging for computer science students, as they must have strong analytical and reasoning skills to create effective programs. [4]. Learning programming requires an understanding of concepts to achieve program goals, this causes difficulties in learning computer programming [5]. For beginner students, most of them have difficulty learning the Java programming language [6]. Beginner students have difficulty understanding concepts when they first learn programming, because students need to understand how to use program code to solve problems [7]. Another reason students have difficulty learning programming is that students tend not to be able to change from a case study to a problem solving program [6].

The use of innovative digital learning technology is very important to be carried out in various forms according to its function and purpose [8] [9]. The impact of using this innovative technology can improve the quality of teaching and self-learning of students because they have used digital platforms [10]. One of the innovations in the use of learning technology is the drag and drop feature in an application because it can improve students' cognitive ability in solving the given test questions [11]. In learning technology, the use of drag and drop can also affect the accuracy of student performance and affect the speed of student performance [12]. In addition, students feel satisfied and happy when learning with applications that use a drag and drop user interface [13]. The implementation of using drag and drop is also carried out in the close test interface design for filling empty elements in questions compared to filling empty elements using conventional interfaces [14]. The results show that the use of interface design with the addition of drag and drop has a faster processing time. Previous research related to the use of drag and drop was on the development of ThaiLabanXML to support the automatic creation of ThaiLabanXML files, where the result is that it can reduce the problem of user errors when creating ThaiLabanXML files [15]. The use of drag and drop on the user interface has also proven to be more effective in increasing student performance and speed in conducting online tests [12]. Thus the drag and drop approach can be applied in learning Java programming logic to facilitate students' understanding. Technological approaches can be adopted and new strategies can be implemented.

In this research, the development of PseudoLearn was carried out to help learn programming logic using the drag and drop concept approach which is complemented by log data. Based on this, there are 2 research questions:

RQ1: Do students who use PseudoLearn get a better understanding than using conventional methods?

RQ2: Are there any differences in the habits of novice and good students when using Pseudolearn in learning programming logic?

## 2 PseudoLearn

This application is used by students to understand the pseudo code algorithm through the reconstruction of the pseudo code algorithm from the case study to complete the algorithm steps by dragging and dropping the answers on the left to develop the right algorithm.

This application contains case studies and algorithm solutions in the form of pseudo code, where students are asked to sequence the completion steps by filling in the blank steps and specifying the data type as needed. In carrying out the reconstruction of the algorithm can be done by drag and drop into the empty elements of the several alternative answers that have been provided. Students can check whether the answers to solving case studies with pseudocode are correct or wrong, and the system will correct answers automatically and provide feedback which answers are right and wrong. This application is designed in the form of levels of workmanship, which consists of 5 levels. The level indicates the level of difficulty of the case study based on the material. Level 1 is the first time you work on it and students cannot proceed to the next level before completing that level.

In addition, this application is also equipped with a data log. The data log is used as a record keeper of students working on case studies using this application, starting from the number of experimental steps, levels, time, and work history of how many times the wrong and correct answers were done.

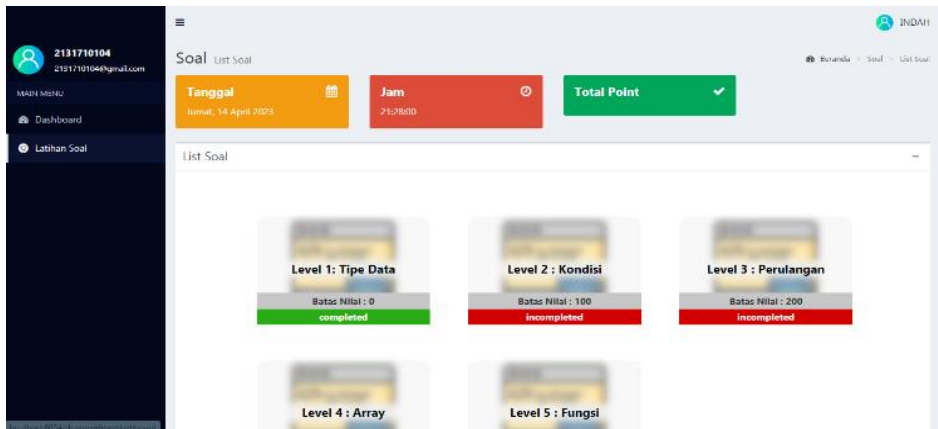


Fig. 1. Case Study Level View

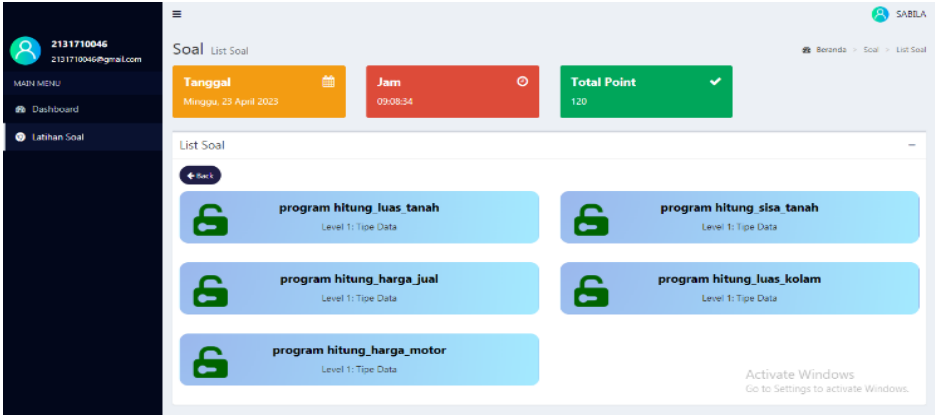


Fig. 2. View of the Case Study List at Level 1

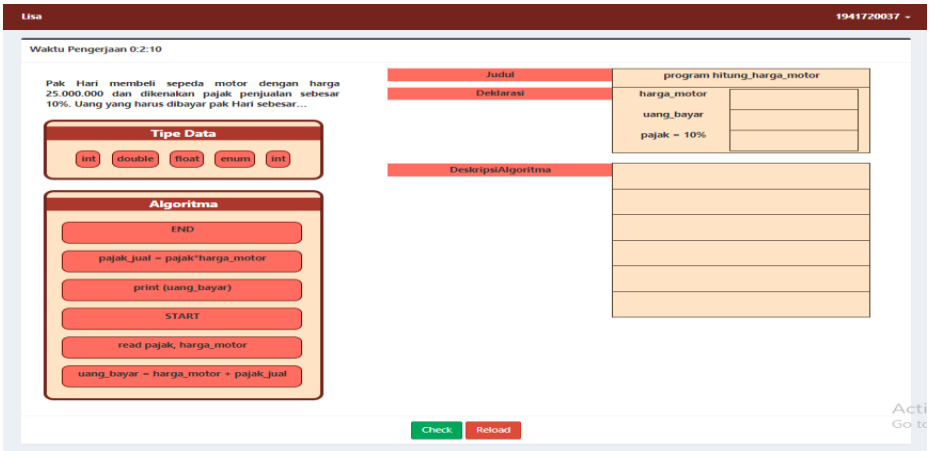


Fig. 3. View the Case Study Exercise using Drag and Drop

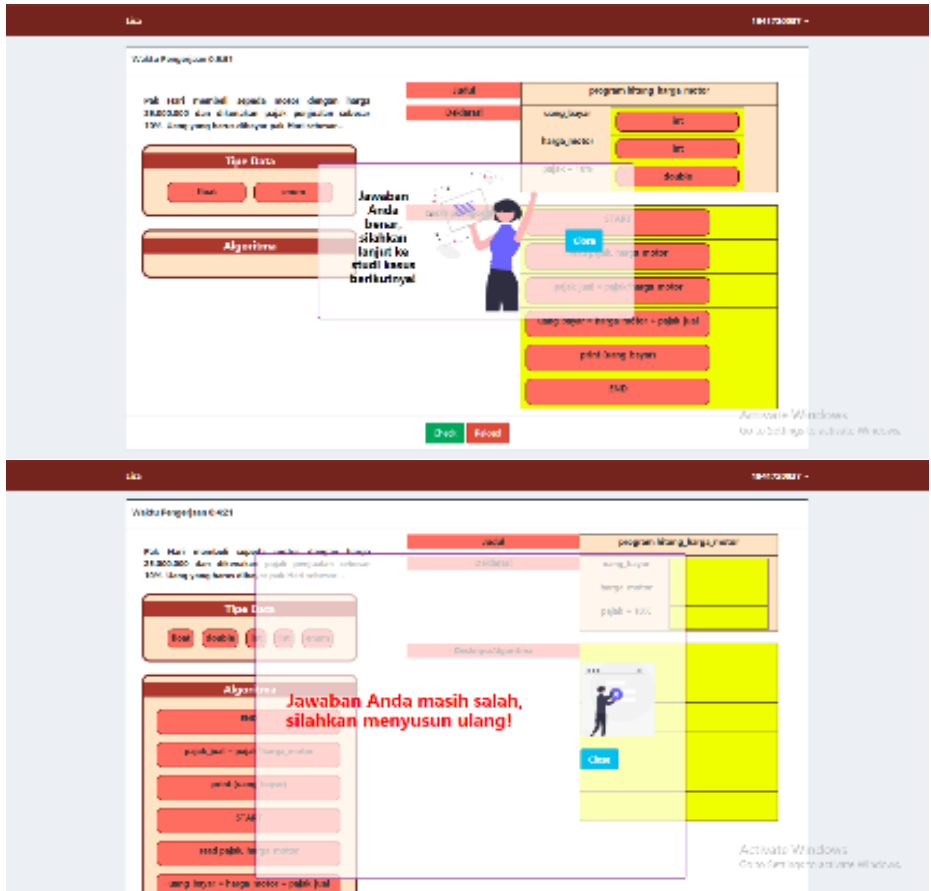


Fig. 4. Display Check Incorrect and Correct Answers

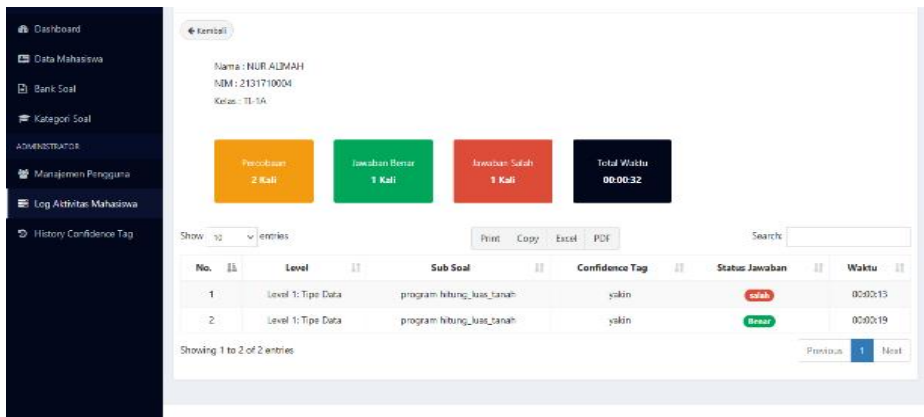


Fig. 5. Data Log View

### 3 EXPERIMENTAL SETTING

The experimental setting is designed to support the research question. A controlled experimental environment is a design with several activities to ensure groups are doing activities equally. Students will only follow the sequence of activities that have been designed.

#### 3.1 Procedure

The number of participants in this study were 39 students of the Business Information Systems study program, Department of Information Technology, State Polytechnic of Malang. The research was conducted within 2 weeks. In the first week, students carry out 3 activities which can be seen in Fig. 6.

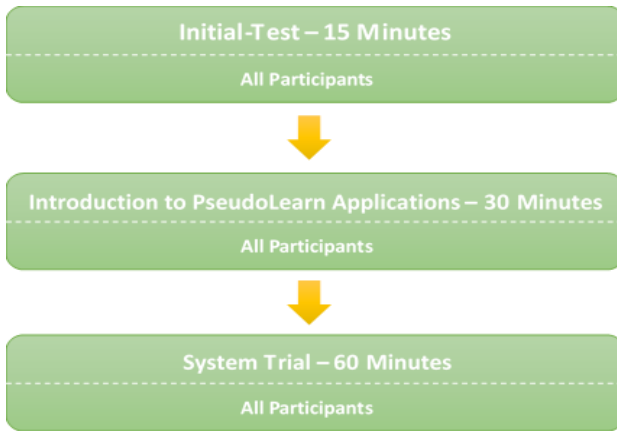


Fig. 6. Experiment Design - Week 1

At the initial test stage, students are given 10 questions to complete within 15 minutes through the LMS (Learning Management System) of the State Polytechnic of Malang. The questions given contain material about basic knowledge of algorithms in Java programming language. The analytical method in this research is ANOVA which is used to test the hypothesis for the average comparison between groups [16]. The results of the initial test will later be used to divide students into 3 groups are control, experiment 1, and experiment 2 based on the results of the single factor ANOVA statistical analysis. The next stage is the introduction of the PseudoLearn application where students are given information regarding the features and how to use the application for 10 minutes. System trial, after students understand the flow of using the PseudoLearn application, students will be given approximately 1 hour to try the application.

Table 1 shows that the average values of the three groups are almost the same. Table 2 shows the results of the ANOVA calculations for the three groups which are not

significant where the p-value = 0.984622. It can be concluded that the three groups have the same programming ability.

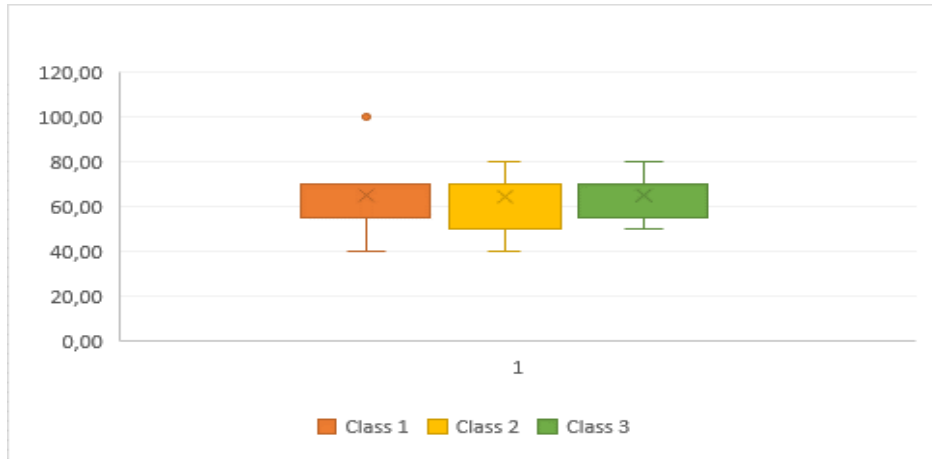
**Table 1.** Initial Test – Descriptive Analysis

<i>Groups</i>	<i>Count</i>	<i>Sum</i>	<i>Average</i>	<i>Variance</i>
Class 1	13	840	64,61538	243,5897
Class 2	13	830	63,84615	158,9744
Class 3	13	840	64,61538	93,58974

**Table 2.** Initial Test – Anova Table

<i>Source of Standard Deviation</i>	<i>SS</i>	<i>Df</i>	<i>MS</i>	<i>F</i>	<i>P-value</i>	<i>F crit</i>
Between Groups	5,12820	2	2,56410	0,0155	0,984622	3,25944
Within Groups	5953,84	36	165,384			
Total	5958,97	38				

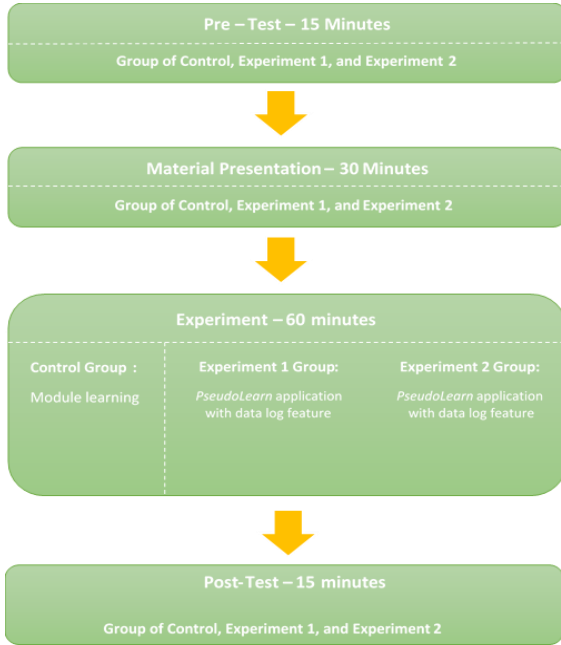
Fig. 7 is the distribution of the initial test scores of the three groups. Class 3 is the control group, class 2 is the experimental group 1, and class 1 is the experimental group 2. The lower limit of experimental group 1 is lower than that of the control group and experiment 2, but the three groups have the same mean.



**Fig. 7.** Box Plot for Initial Test

In the second week, several series of activities will be carried out with the aim of comparing the control group, experiment 1, and experiment 2. Comparisons are made based on the treatment given to each group. Experimental groups 1 and 2 were given access to the PseudoLearn application which had been equipped with a data log, the

control group is students who are given conventional learning without using the PseudoLearn application. Fig. 8 is the activities for the second week.



**Fig. 8.** Experiment Design - Week 2

### 3.2 Result

Pre-test and post-test data from experimental results were analyzed to answer research questions. The analysis was performed using 2-way ANNOVA. The use of 2-way ANOVA itself was carried out because in this study the variable ability of students as the dependent variable would be influenced by 2 other variables are the test and group variables, so that for this condition it was not possible to carry out 1-way ANOVA analysis because there would be 2 factors that influence dependent variable. Before carrying out a 2-way ANOVA analysis, we first tested the normality of the data for the pre-test and post-test values of each group. Table 3 shows that the Shapiro-Wilk significance value is 0.107 which is higher than the 5% (0.05) or Sig. >0.05. So it can be concluded that the pre-test and post-test data values of each group are normally distributed.



**Table 3.** Data Normality Test-Shapiro Wilk

Tests of Normality						
	Kolmogorov-Smirnov <sup>a</sup>			Shapiro-Wilk		
	Statistic	df	Sig.	Statistic	df	Sig.
Standardized Residual for student abilities	.139	78	.001	.974	78	.107
a. Lilliefors Significance Correction						

The next stage will be testing the homogeneity of the post-test value data using the Levene's Test method. The homogeneity test was not carried out for the pre-test value data because it came from the same sample data. Table 4 shows the post-test homogeneity test.

**Table 4.** Post-Test Homogeneity Test Results

Levene's Test of Equality of Error Variances <sup>a</sup>			
Dependent Variable: student Abilities			
F	df1	df2	Sig.
.116	2	36	.891
Tests the null hypothesis that the error variance of the dependent variable is equal across groups.			
a. Design: Intercept + Test + Group + Test * Group			

Based on the Table 4 the Significance value (Sig.)  $0.891 > 0.05$  so it can be concluded that the variation in data between the control group, experiment 1, and experiment 2 is the same or homogeneous. Because the results of the analysis prerequisite test showed that the normality test and homogeneity test met the criteria, the researcher continued the parametric statistical analysis, namely the 2-way ANOVA method.

**Table 5.** Results Of 2-Way Anova Test

Tests of Between-Subjects Effects							
Dependent Variable: student abilities							
Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared	
Corrected Model	5410.256 <sup>a</sup>	5	1082.051	6.890	.000	.324	
Intercept	446282.051	1	446282.051	2841.633	.000	.975	
Test	4020.513	1	4020.513	25.600	.000	.262	

Group	1202.5 64	2	601.282	3.829	.026	.096
Test * Group	187.17 9	2	93.590	.596	.554	.016
Error	11307. 692	7 2	157.051			
Total	463000 .000	7 8				
Corrected Total	16717. 949	7 7				

a. R Squared = .324 (Adjusted R Squared = .277)

Based on Table 5, it can be seen that the ANOVA calculation shows that there are significant differences in student abilities in terms of the pre-test and post-test scores obtained by the participants as a whole, this can be seen from the Significance (Sig.) Test value of  $0.000 < 0.05$ . For a comparison of the control group, experiment 1, and experiment 2, it was also found that there were significant differences in student abilities, where the value of Sig. Group  $0.026 < 0.05$ . This shows that the special treatment given to each group during the experimental activity had an influence on students' understanding or ability. Furthermore, for the Test x Group comparison (comparison of the interaction between the group and the test) no significant interaction was found due to the Sig value. Test\*Group  $0.554 > 0.05$ . The absence of this interaction indicates that the characteristics of the differences between the PseudoLearn and conventional learning models for all levels of students' initial abilities are the same. Students who have high initial ability or understanding tend to get high results, both with conventional and PseudoLearn learning media.

Log data analysis was also carried out to be used as a preliminary analysis that describes certain student behavior. The log data used is the number of steps taken by students and the time to reach the correct answer. The application requires the correct answer to be able to work on the next question and the next level. Students can try to answer by rearranging the pseudocode algorithm without any limit until the answer is correct. Pearson Correlation Analysis was performed for the initial, step, and time tests taken by students. Tables 6 and 7 show that a negative correlation exists between the initial test and the steps taken as well as the initial test and time.

**Table 6.** Pearson Correlation Initial-test And Step

	<i>Initial-test</i>	<i>Step</i>	
Initial-test		1	
Step	-0,1887		1

**Table 7.** Pearson Correlation Initial-test And Time

	<i>Initial-test</i>	<i>Time</i>	
Initial-test		1	
Time	-0,379920853		1

### 3.3 Analysis

Based on the results found above, the experimental groups 1 and 2 had better post-test average scores than the control group. Where the two experimental groups received treatment using the PseudoLearn application. Although scores increased from pre-test to post-test for the control group was also found, no significant increase was found in experimental groups 1 and 2. These results provide an understanding that applications can help students to understand better when learning programming logic. The plot of the Mean Pre-test and Post-test can be seen in Fig. 9.

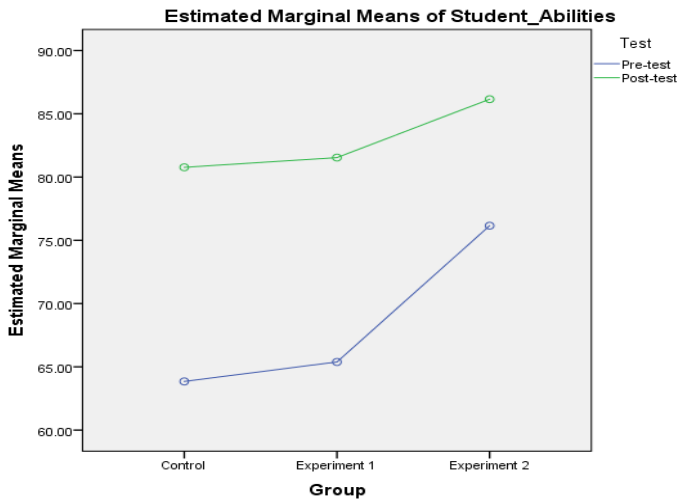


Fig. 9. Mean From Pre-Test and Post-Test

The boxplot results above show changes in the pre-test and post-test values of each group together or not in contradiction. This is what affects the Group\*Test interaction, there is no significant difference. However, if it is reviewed based on the test variables and also the groups separately, there are significant differences between experiment 1, experiment 2, and the control group. This proves that the use of applications for learning pseudocode has a fairly good influence on increasing student understanding.

Scatter plots are used to explain the pearson correlation results. Based on Fig. 10 and 11 it shows that a negative correlation occurs in students who get low scores on the initial test. Under these conditions, the value of the initial-test has an influence on the large number of experiments carried out by students and the time needed to complete the practice questions. The higher the initial test score, the smaller the number of experiments carried out and the time needed for students to find the right answer, and vice versa, the lower the initial test score, the greater the number of experiments carried out and the time needed to find the correct answer. This proves that novice students will have a moderate chance of taking higher steps and time compared to stu-

dents who have higher initial scores. This shows that students with low scores have difficulty learning programming logic.

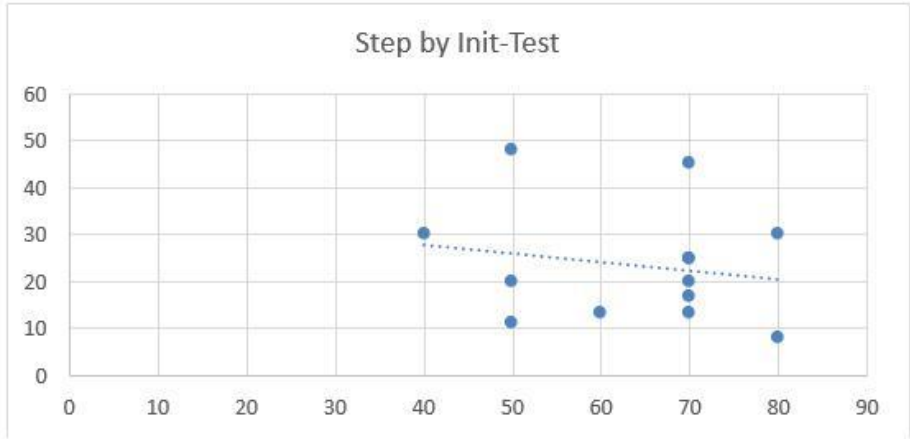


Fig. 10. Scatterplot for step vs init-test score

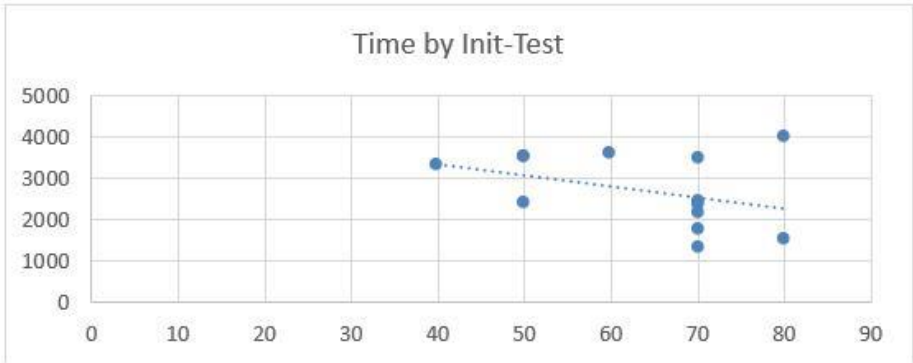


Fig. 11. Scatterplot for time vs init-test score

#### 4 CONCLUSION

Based on the results of the research and discussion that has been done, it can be concluded that the use of the PseudoLearn application that implements the drag and drop feature can have a positive impact on students' understanding of programming material. This can be observed from the results of the ANOVA test which shows that there is a significant effect in terms of group division. PseudoLearn, which is equipped with

log data, can provide analysis even though it is still at an early stage, but can provide a small perspective on students' thinking efforts. Answer to RQ1: "Do students who use PseudoLearn get a better understanding than students who use conventional methods?". The results showed that the post-test scores for the experimental group 1 and 2 were higher than the control. In all three groups there was an increase in scores, but the increase in scores in the control group did not exceed experimental groups 1 and 2. RQ2's answer: "Is there a difference in the habits of novice students and good students when using Pseudolearn in learning programming logic?". The results show that novice students need a lot of steps and time when learning programming logic. Based on this, it can be concluded that novice students have difficulty getting the correct answer. Even though the log data cannot describe the thinking process of novice students, it can show the number of trials and time needed to find the right answer. In further research, log data analysis can be carried out which is used to describe which part of the student's difficulties when learning programming logic.

## References

1. R. Hornthvedt and T. Akesson, "Java, Python and Javascript, a comparison," *Indep. Proj. 15 Credit. degree Bachelor Sci. with a major Comput. Sci. Spring 2019 Fac. Nat. Sci.*, 2019, [Online]. Available: [www.hkr.se](http://www.hkr.se)
2. X. Lu, N. Funabiki, S. T. Aung, Y. Jing, and S. Yamaguchi, "An Implementation of Java Programming Learning Assistant System in University Course." *IEEE*, 2023. doi: 10.1109/ICIET56899.2023.10111422.
3. S. Khoirom, M. Sonia, B. Laikhuram, J. Laishram, and D. Singh, "Comparative Analysis of Python and Java for Beginners Cite this paper Comparative Analysis of Python and Java for Beginners," *Int. Res. J. Eng. Technol.*, vol. 07, no. 08, pp. 4384–4407, 2020, [Online]. Available: [www.irjet.net](http://www.irjet.net)
4. M. D. Siti Rosminah and M. A. Ahmad Zamzuri, "Difficulties in learning Programming: Views of students," *1st Int. Conf. Curr. Issues Educ.*, no. September, pp. 74–78, 2012, doi: 10.13140/2.1.1055.7441.
5. H. L. Fwa, "Predicting non-completion of programming exercises using action logs and keystrokes," *Proc. - 2019 Int. Symp. Educ. Technol. ISET 2019*, pp. 271–275, 2019, doi: 10.1109/ISET.2019.00064.
6. J. Maiga, "Gamification for Teaching and Learning Java Programming for Beginner Students — A Review," *J. Comput.*, vol. 14, no. 9, pp. 590–595, 2019, doi: 10.17706/jcp.14.9.590-595.
7. M. Majalin, A. Baharum, R. Ismail, I. Ismail, E. G. Mounq, and N. A. M. Noor, "A Preliminary Study of Difficulties in Learning Java Programming for secondary school," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 9, no. 1.4 Special Issue, pp. 302–306, 2020, doi: 10.30534/ijatcse/2020/4591.42020.
8. M. Basri, A. A. Patak, A. Musdariah, and A. Abduh, "Innovative Learning Technology (ILT) in Indonesian Vocational Higher Education," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 10, no. 6, pp. 2599–2605, 2020, doi: 10.18517/ijaseit.10.6.11348.
9. A. Haleem, M. Javaid, M. A. Qadri, and R. Suman, "Understanding the role of digital technologies in education: A review," *Sustain. Oper. Comput.*, vol. 3, no. February, pp. 275–285, 2022, doi: 10.1016/j.susoc.2022.05.004.

10. R. S. Manzano Pérez, T. E. López Pérez, R. J. Manzano Pérez, and M. V. Pérez López, "Technological innovation and education: a brief review of the literature," *Ibero-American J. Educ. Soc. Res.*, vol. 3, no. 1, pp. 25–30, 2023, doi: 10.56183/iberoeds.v3i1.596.
11. B. Arslan, Y. Jiang, M. Keehner, T. Gong, I. R. Katz, and F. Yan, "The Effect of Drag-and-Drop Item Features on Test-Taker Performance and Response Strategies," *Educ. Meas. Issues Pract.*, vol. 39, no. 2, pp. 96–106, 2020, doi: 10.1111/emip.12326.
12. H. R. Ponce, R. E. Mayer, and M. S. Loyola, "Effects on Test Performance and Efficiency of Technology-Enhanced Items: An Analysis of Drag-and-Drop Response Interactions," *Journal of Educational Computing Research*, vol. 59, no. 4, pp. 713–739, 2022. doi: 10.1177/0735633120969666.
13. C. Phewkum, J. Kaewchaiya, K. Kobayashi, and K. Atcharyachanvanich, "Scarmbel SQL: A Novel Drag-and-drop SQL Learning Tool," *Int. Comput. Sci. Eng. Conf.*, 2020.
14. H. R. Ponce, R. E. Mayer, J. Sitthiworachart, and M. J. López, "Effects on response time and accuracy of technology-enhanced cloze tests: an eye-tracking study," *Educ. Technol. Res. Dev.*, vol. 68, no. 5, pp. 2033–2053, 2020, doi: 10.1007/s11423-020-09740-1.
15. Y. Tongpaeng, P. Maphon, and P. Sureephong, "Improving Thai LabanXML standard for the automatic Thai laban drag drop system," *ECTI DAMT-NCON 2019 - 4th Int. Conf. Digit. Arts, Media Technol. 2nd ECTI North. Sect. Conf. Electr. Electron. Comput. Telecommun. Eng.*, pp. 168–173, 2019, doi: 10.1109/ECTI-NCON.2019.8692313.
16. U. Singh and C. M. Pandey, "Application of student 's t-test , analysis of variance , and covariance Application of Student ' s t - test , Analysis of Variance , and Covariance," *Ann. Card. Anaesth.*, no. October, 2019, doi: 10.4103/aca.ACA.

**Open Access** This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (<http://creativecommons.org/licenses/by-nc/4.0/>), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

