JURNAL UPI

Jurnal Pendidikan Multimedia (EDSENCE)

Jurnal Pendidikan Multimedia EDSENCE Education, Science, and Creative Technology

Journal homepage: https://ejournal.upi.edu/index.php/Edsence

DliLearn: LMS-Assisted PBL for Critical Thinking in Algorithms and Programming: One-Group Pretest–Posttest Study

Fadli Ridwansyah, Andini Setya Arianti, Munir Munir

Universitas Pendidikan Indonesia, Indonesia

Correspondence: E-mail: fadliridwansyah28@upi.edu

ABSTRACT

Low student participation and weak critical thinking skills in algorithm and programming learning motivated this study. Teacher-centered learning and limited learning media often hinder students from understanding programming logic and concepts. Therefore, this study implemented a Problem-Based Learning (PBL) model combined with a Deep Learning supported by a Learning Management System (LMS) called DliLearn. This study employed Research and Development (R&D) employing the ADDIE framework model and a one-group pretest-posttest design. The participants were 25 students of class XI-3 at SMAN 1 Katapang. Data were acquired through observation, interviews, questionnaires, pretests, and posttests, and were analyzed using the Shapiro-Wilk normality test, the Wilcoxon Signed Rank Test, and N-Gain calculation. The Wilcoxon test revealed a significance level of $p < .001$), suggesting the existence of a statistically significant difference between pretest and posttest scores after implementation. The average N-Gain score was 0.708, slightly above the high-criteria threshold, with 14 students in the high criteria, 11 in the moderate criteria, and none in the low criteria. These findings indicate improved critical thinking skills after implementation, although causal claims should be made cautiously.

ARTICLE INFO

Article History:
Submitted/Received 29 May 2026
First Revised 07 Jun 2026
Accepted 15 Jun 2026
First Available online 26 Jun 2026
Publication Date 26 Jun 2026

Keyword:
Problem-Based Learning, Deep Learning Approach, Learning Management System, Critical Thinking Skills, Algorithm and Programming, One-Group Pretest–Posttest Design.

1. INTRODUCTION

In algorithm and programming learning, students are required to understand logical structures, analyze problems, and develop systematic solutions. These abilities are closely related to critical thinking constitutes a fundamental competency requisite for the 21st century, facilitating students to analyze information and determine choice based on logical reasoning (Damayanti et al., 2020). However, in practice, algorithm and programming learning is often currently characterized by teacher-centered instruction, theoretical explanation, and syntax memorization. Such conditions limit students' opportunities to actively solve problems, test ideas, and evaluate program logic. Therefore, learning needs to be designed to enhance students' critical thinking skills, particularly in topics that require logic and problem solving, such as algorithm and programming.

Various studies indicate that the critical thinking capabilities of Indonesian learners remain classified at a low level. One study using essay tests referencing Facione's dimensions of critical thinking, namely interpretation, analysis, evaluation, inference, explanation, and self-regulation, indicated that students' critical thinking skills remained low, with an overal mean value of 42.95 out of 100. The lowest scores were found in the inference indicator at 30.77 and the evaluation indicator at 34.62, indicating that learners continue to face challenges when performing analysis problems and formulating rational inferences (Zebua et al., 2024). Another study involving tenth-grade senior high school students also reported similar findings, where the explanation indicator reached only 9.68% and interpretation 38.71%, both categorized as low (Qohar & Sulandra, 2021). These findings suggest that learners' critical thinking capabilities require further enhancement, especially in learning that requires logical reasoning and problem-solving abilities, such as algorithm and programming.

In algorithm and programming learning, instructional practices that still emphasize memorization and conventional teaching methods are among the factors contributing to students' low critical thinking skills. Such conditions make students less accustomed to analytical thinking and logical problem solving, resulting in limited understanding of programming logic concepts. A preliminary needs analysis conducted before the treatment, involving 25 eleventh-grade students through a five-point Likert scale questionnaire, showed that approximately 73% of students reported difficulties in understanding algorithm and programming materials. This percentage refers to students' perceived learning difficulties, particularly in understanding programming logic, syntax, and problem-solving procedures.

The learning process, which still largely applies a teacher-centered approach, is a key element driving this phenomenon, causing students to become passive listeners rather than active participants in classroom activities (Rozali et al., 2022). Several factors contributing to students' low critical thinking skills include conventional learning methods, low learning motivation, and limited problem-solving activities during instruction. Furthermore, the utilization of instructional media that is still limited to PowerPoint presentations (PPT) and PDF modules has not successfully stimulated students to take an active role throughout the educational process. As a result, a gap emerges between students' learning needs and the learning experiences they receive.

Based on the same preliminary needs analysis questionnaire, which was distributed to 25 eleventh-grade students before the treatment was implemented, students showed a need

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

for more interactive, collaborative, and integrated learning media. The questionnaire item allowed students to select more than one relevant answer. The results showed that 84% of students expected media that supported communication and collaboration, 68% needed media that integrated all learning activities in one platform, 56% expected more interactive and visual media, 52% needed easier access to learning materials, and 36% preferred website-based media. These findings provide an empirical basis for developing LMS that integrates learning materials, assignments, discussions, coding practice, and collaborative activities. Therefore, there is a critical need for an instructional framework that facilitates dynamic dialogue, issue resolution, and teamwork, which can be effectively addressed by adopting Problem-Based Learning (PBL).

The implementation of the Problem Based Learning framework represents considered one way to improve this situation. This instructional method is capable of help enhance critical thinking skills with encourage students' active participation in the learning process because the materials and problems discussed are closely related to real-life situations (Prasetyo & Kristin, 2020). The PBL model consists of five stages: problem orientation, organizing students, guiding investigation, exhibiting the final findings, followed by reviewing and critiquing the entire problem-solving sequence. These stages encourage learners to think dynamically, engage in dialogue, work synergistically, and resolve complex issues to real-life contexts. In algorithm and programming learning, the PBL model can encourage students to understand concepts more contextually while also training critical thinking, problem-solving, communication, and teamwork skills. To optimize applying the PBL framework to enable students to reach deeper understanding and do not merely memorize procedures, this model needs to be strengthened through a learning approach that promotes awareness and meaningful learning.

An approach relevant to be integrated with the PBL model is the Deep Learning approach. In this study, Deep Learning refers to a teaching strategy that emphasizes meaningful learning, mindful learning, and joyful learning, rather than deep learning within the realm of artificial intelligence or neural networks (Diputera et al., 2024). This approach emphasizes deep conceptual understanding so that students avoid merely committing information to memory without deep comprehension but also understand relationships between concepts and apply them in real-life contexts (Qurrota et al., 2025). Through the incorporation of the PBL model with Deep Learning as a pedagogical approach, learners are expected to become more active in thinking, discussing, solving problems, and reflecting on their learning process. However, to support an interactive and collaborative learning process, appropriate learning media that utilize technology and integrate all learning activities within a single platform are needed.

One technology capable of addressing these needs is the Learning Management System (LMS). LMS is a digital learning medium that enables structured and connected learning processes between teachers and students within a single platform, including material delivery, discussions, assignment submission, and learning evaluation (Mahabu et al., 2025). Through the Cognitive Theory of Multimedia Learning (CTML), posits that instructional activities becomes more optimal when materials are presented through both visual and verbal channels simultaneously because the human brain processes information through two

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

different channels with limited capacity (Mayer, 2001). Therefore, LMS can serve as an interactive learning medium that supports students' learning processes.

This condition aligns with the attributes of the Problem-Based Learning model, which emphasizes discussion, collaboration, and collaborative problem solving, as well as the Deep Learning approach, which encourages students to understand concepts more meaningfully, deeply. Therefore, the incorporation of LMS with the PBL model and the Deep Learning approach is expected to establish a more interactive, learner-focused educational atmosphere while improving students' critical thinking skills in algorithm and programming.

Although previous studies have examined LMS-assisted Problem-Based Learning, studies that integrate the Problem-Based Learning model with a Deep Learning approach in algorithm and programming learning at the Indonesian senior high school level remain limited. The novelty of this investigation centers on the incorporation of an LMS-based learning environment with the five stages of Problem-Based Learning and the principles of meaningful, mindful, and joyful learning. This integration is applied in algorithm and programming learning, which requires students to use logical reasoning, analyze problems, and develop systematic solutions. Furthermore, this study analyzes students' critical thinking skills not only based on overall score improvement but also based on Facione's critical thinking indicators through N-Gain analysis for each indicator. Therefore, this study aims to implement an LMS-assisted PBL model with a Deep Learning approach and to analyze the enhancement of students' critical thinking skills in algorithm and programming learning at SMAN 1 Katapang.

2. METHODS

This investigation employed a Research and Development (R&D) methodology, specifically adopting the ADDIE development framework. This method was employed because the study not only focused on measuring the enhancement of students' critical thinking skills, but also on implementing an educational platform developed in the form of a Learning Management System (LMS) integrated with the PBL model with the Deep Learning approach.

2.1. Research Framework

This investigation utilized a one-group pretest–post-test design as preliminary classroom research. In the initial stage, learners were given a pretest to identify their baseline critical thinking capabilities prior in algorithms and programming. Next, learners participated in the learning process as a treatment using the developed LMS, DliLearn, which integrated the PBL model and a Deep Learning approach. Following the conclusion of the instructional activities, a post-test was administered to the learners to identify changes in their critical thinking skills after the treatment.

Nonetheless, a key limitation of this research layout lies in its internal validity, given that it lacks a control group to isolate the true impact of the intervention. Therefore, the improvement in post-test scores cannot be interpreted as the sole effect of the treatment. Other factors, such as students' maturation, testing effects from repeated exposure to similar test items, teacher influence during the learning process, and students' familiarity with the learning materials may also have contributed to the observed gains.

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

The measurement of critical thinking skills referred to the indicators proposed by Facione (1990), namely: interpretation, analysis, evaluation, inference, explanation, and self-regulation. Each indicator was operationalized into algorithm and programming test items. The operationalization of the indicators is showcased in Table 1.

Table 1. Operationalization of Critical Thinking Indicators in Algorithm and Programming Test

Critical Thinking Indicators	Operationalization in Algorithm and Programming	Example of Test Item	Scoring Criteria	Cognitive Level
Interpretation	Students identify relevant information from a programming problem.	Students are given a problem situation and asked to identify input, process, and output.	Accuracy in identifying input, process, output, and problem requirements.	C2–C3
Analysis	Students analyze the structure of an algorithm or program code.	Students examine a pseudocode/program segment and identify the logic or sequence of steps.	Accuracy in explaining the flow, variables, conditions, and relationships between steps.	C4
Evaluation	Students judge whether an algorithm or program solution is correct and efficient.	Students are asked to check whether a given algorithm produces the correct output.	Accuracy in detecting errors, evaluating logic, and giving reasons for the judgment.	C5
Inference	Students draw logical conclusions based on algorithmic processes or program outputs.	Students predict the output of a program based on given input values.	Accuracy in tracing the algorithm and concluding the correct output.	C4–C5
Explanation	Students explain the reasoning behind an algorithmic solution.	Students are asked to explain why certain steps or conditions are used in an algorithm.	Clarity, completeness, and logical consistency of the explanation.	C4–C5

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

Critical Thinking Indicators	Operationalization in Algorithm and Programming	Example of Test Item	Scoring Criteria	Cognitive Level
Self-regulation	Students review and revise their own algorithm or program solution.	Students identify weaknesses in their solution and propose improvements.	Ability to reflect on errors, revise the solution, and justify the improvement.	C5

Quantitative data analysis commenced with a normality test to assess the distribution of the dataset. For normally distributed data, a Paired Sample t-Test was employed, whereas non-normally distributed data were evaluated using the Wilcoxon Signed Rank Test; both statistical measures aimed to detect significant discrepancies between pretest and posttest scores. Furthermore, the normalized gain (N-Gain) score was calculated to quantify the enhancement of students' critical thinking capacities post-intervention. The specific N-Gain formulation is outlined below:

$$< g >= \frac{Posttest\ Score - Pretest\ Score}{Ideal\ Maximum\ Score - Pretest\ Score}$$

The criteria for the N-Gain score are as follows:

a. $<g> > 0,7$ = High
b. $0,7 \geq g \geq 0,3$ = Moderate
c. $<g> < 0,3$ = Low

2.2. Population and Sample

The target population for this research comprised all eleventh-grade senior high school students who studied Informatics. Population refers to the entire group of subjects with particular attributes that directly correspond to the intended objectives of the study, allowing the research findings to serve as the basis for drawing conclusions (Sugiyono, 2019). All eleventh-grade students participating in Informatics learning were included in the research population because they shared similarities in learning materials, educational level, and curriculum.

The research sample was drawn from class XI-3 at SMAN 1 Katapang, comprising 25 students who were studying algorithms and programming. This specific class was selected through a purposive sampling technique, based on specific criteria that because it met the needs and objectives of the study. Specifically, class XI-3 was considered appropriate because the students were learning the targeted material, participated in the same classroom learning process, and were available to complete the pretest, treatment, and posttest stages.

However, because the sample was selected purposively and involved only one class, class XI-3 was not intended to statistically represent all students at SMAN 1 Katapang or other schools. Consequently, the outcomes of this research ought to be construed strictly within

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

the boundaries of class XI-3 and should not be generalized beyond this classroom context without further research involving broader samples and comparison groups.

This study employed a purposive sampling technique, whereby participants are selected based on predetermined criteria and considerations according to the research needs. This technique was used because the study was conducted without random assignment of subjects. Instead, the selected class was considered the most suitable for the research conditions and capable of representing the problems being investigated (Sugiyono, 2019).

The selection of class XI-3 was based on the subject teacher's recommendation by considering class readiness, the suitability of the learning schedule, and the ease of implementing the treatment. The use of purposive sampling was deemed highly suitable given that the investigation took place in an actual classroom learning situation.

2.3. Research Stages

The research stages in this research were conducted adopting the ADDIE development model. According to Branch, as cited in (Hidayat & Nizar, 2021), ADDIE is a development model used as a systematic framework for designing and developing learning processes. The ADDIE model comprises five core phases, namely Analysis, Design, Development, Implementation, and Evaluation. In this research, each phase produced particular outputs that were used as the basis for the next phase. The outputs included the needs analysis results, LMS design, learning tools, LMS prototype, validated instruments, implementation data, and evaluation results. The research stage flow in the ADDIE model is showcased in Figure 1.

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

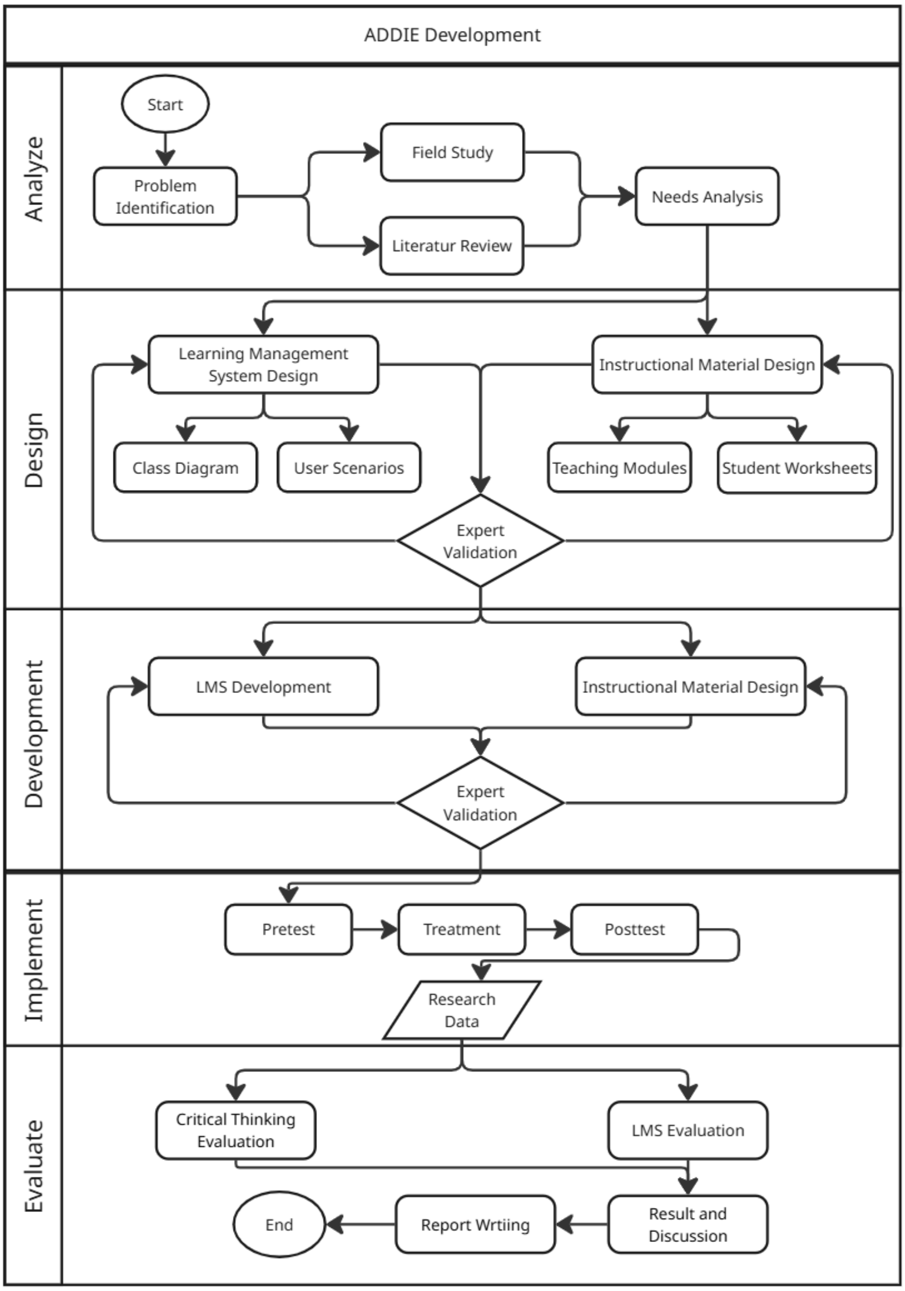


Figure 1. Research Stages

2.3.1 Analysis Stage

At this phase, the developer executed a learning needs analysis to identify problems occurring during the learning process, particularly in algorithm and programming materials. The analysis was executed through observation, questionnaires, interviews, and literature review. The outputs of this stage were the identification of students' learning difficulties, students' needs for digital learning media, teacher needs, and the initial requirements for the LMS to be developed.

Observations and questionnaires were conducted with eleventh-grade students at SMAN 1 Katapang to uncover the academic impediments experienced by learners in understanding algorithm and programming materials and their needs regarding learning media. The

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

questionnaire was used to examine students' perceived learning difficulties, preferences for learning media, and the need for communication and collaboration features in learning. In addition, a literature review was executed by examining references relevant to the Deep Learning approach, the PBL model, Learning Management System (LMS), and critical thinking skills based on Facione's indicators. The outcomes of this stage were used as the basis for determining the learning scenario, system requirements, and LMS design.

2.3.2 Design Stage

At this phase, the initial design of the LMS and learning tools was prepared based on the outcomes of the analysis stage. The outputs of this stage included the LMS structure, class diagram, interface design, learning flow, LMS usage scenario, teaching modules, Student Worksheets, and assessment blueprint. The LMS was designed to support the stages of Problem-Based Learning, including problem orientation, student organization, investigation, presentation of work results, and evaluation of the problem-solving process.

The learning scenario was also designed by integrating the PBL stages with the Deep Learning approach, which emphasizes meaningful, mindful, and joyful learning. In addition, the researcher prepared learning tools based on the learning objectives and algorithm and programming materials. The assessment blueprint was developed predicated on Facione's critical thinking indicators, namely interpretation, analysis, inference, evaluation, explanation, and self-regulation.

2.3.3 Development Stage

In the development phase, the developer built the LMS based on the plans prepared in the previous stage. The process included designing the user interface, organizing the learning flow according to the syntax of the PBL model, and developing learning activities within the LMS. In addition, the developer also built supporting learning tools and research instruments, including teaching modules, Student Worksheets, and pretest-post-test questions to assess students' critical thinking skills predicated on Facione's indicators.

After the LMS prototype, learning tools, and research instruments had been developed, expert validation was conducted before the implementation stage. The validation involved material and media experts to evaluate the feasibility and appropriateness of the developed LMS and learning materials. The validation instrument was adapted from the Learning Object Review Instrument (LORI) and used a 1-to-5 Likert scale featuring five distinct response options. Several dimensions, including content quality, learning goal alignment, feedback and adaptation, and motivation, were assessed during the material validation. On the other hand, presentation design, interaction usability, accessibility, reusability, and standards compliance were assessed by the media validation. These dimensions were used to verify that the LMS was appropriate in terms of content, learning objectives, interface design, usability, accessibility, and suitability for implementation in algorithm and programming learning.

In addition to the quantitative rating, the validator also provided qualitative comments and revision suggestions. The comments were reviewed and used as the basis for improving the LMS design, learning flow, learning materials, and assessment instruments before the LMS was implemented in the classroom. Revision suggestions related to the suitability of LMS activities with the PBL syntax, the clarity of student roles, the placement of learning materials,

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

and the organization of learning activities were considered to ensure that the LMS supported problem orientation, student organization, investigation, presentation, and evaluation activities appropriately. The revised LMS was then used in the implementation stage after the validation and revision process had been completed.

2.3.4 Implementation Stage

At the implementation stage, the validated LMS, learning tools, and test instruments were applied throughout pedagogical activities. The output of this stage was the implementation data consisting of students' pretest scores, post-test scores, learning activity records, and students' responses toward the LMS. The activities began with a pretest to determine students' prior critical thinking skills. Next, students participated in learning activities using the PBL model with a Deep Learning approach assisted by LMS as the treatment. After all learning activities had been completed, students were delivered a post-test to identify the enhancement in their critical thinking skills after the implementation. The pretest and post-test outcomes were then collected and analyzed as research data. After completing the post-test, students were also asked to complete a TAM-based response questionnaire to identify their responses toward the use of LMS after the learning implementation.

2.3.5 Evaluation Stage

The evaluation phase was executed to analyze the enhancement of students' critical thinking skills after the implementation of the LMS in the learning process. The outputs of this stage included the outcomes of the normality test, hypothesis testing, N-Gain calculation, N-Gain analysis for each critical thinking indicator, and students' responses toward the LMS. The evaluation was based on the pretest results, post-test results, N-Gain calculation, and student response questionnaire.

In addition to measuring students' critical thinking improvement, this study also collected students' responses toward the use of DliLearn after the implementation. The student response instrument was a non-test questionnaire formulated grounded in the Technology Acceptance Model (TAM) introduced by Venkatesh & Davis (2000). The questionnaire consisted of 20 items covering four dimensions: perceived usefulness, perceived ease of use, attitude toward using, and behavioral intention to use. The questionnaire instrument adopted a Likert scale featuring five response categories, ranging from strongly disagree to strongly agree.

The student response score was analyzed descriptively using a percentage formula by calculating the ratio of the obtained score to the maximum score and multiplying the quotient by 100%. The results were then interpreted into response categories to determine students' acceptance of the LMS after the learning implementation. The evaluation results were discussed to identify the enhancement of students' critical thinking skills and students' responses toward LMS-assisted PBL with a Deep Learning approach in algorithm and programming learning. Based on these findings, the researcher formulated conclusions and suggestions as recommendations for future development.

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

3. RESULTS AND DISCUSSION

3.1 Result

3.1.1 Analysis Stage

1) Student Needs Analysis

The student needs analysis was conducted before the treatment through interviews and a questionnaire distributed to 25 eleventh-grade students at SMAN 1 Katapang. The questionnaire used a five-point Likert scale and aimed to identify students' perceived difficulties in learning algorithm and programming as well as their needs for digital learning media. In addition, interviews were conducted to obtain supporting information regarding students' learning experiences and classroom learning conditions. The outline of the student needs analysis is showcased in Table 2.

Table 2. Summary of Student Needs Analysis Results

Aspect Analyzed	Main Finding	Evidence
Difficulty in learning algorithm and programming	Students experienced difficulties in understanding algorithm and programming materials	Approximately 73% of students reported difficulties, particularly related to programming logic, syntax, and problem-solving procedures
Learning media needs	Students needed media that supported communication and collaboration	84% of students selected this need
Integrated learning platform	Students needed media that could integrate all learning activities in one platform	68% of students selected this need
Interactive and visual media	Students expected media that was more interactive and visual than conventional textbooks or PDF modules	56% of students selected this need
Access to learning materials	Students needed easier access to learning materials	52% of students selected this need
Website-based media	Some students preferred website-based learning media	36% of students selected this need
Learning process	Students expected learning activities that involved discussion, problem solving, and collaboration	Supported by interview and questionnaire results

Because the questionnaire item related to learning media needs allowed students to select more than one answer, the percentages represent the proportion of students who selected each option and do not total 100%. These results indicate that students needed learning media that not only presented materials but also supported communication, collaboration, access to learning resources, and integrated learning activities. Therefore, the analysis stage provided an empirical basis for developing an LMS-based learning media to support algorithm and programming learning.

2) Teacher Needs Analysis

Based on the interview with the Informatics teacher, algorithm and programming lessons were generally conducted by first delivering theoretical materials before students engaged in coding practice using computer laboratories or mobile devices. The teacher also implemented group discussions, presentations, and assessments through practical tasks and oral tests to evaluate students' understanding of the learning materials.

The teacher emphasized the need for learning media that could support a more interactive and flexible learning process, particularly media that facilitate independent learning, discussions, and coding practice within a single learning platform. Therefore, the use of a Learning Management System (LMS) was considered a potential solution to support a more collaborative and student-cantered approach to algorithm and programming learning.

3.1.2 Design Stage

To illustrate the system functions and user interactions with the LMS, the researcher developed an LMS named DliLearn. Furthermore, the researcher designed a class diagram to describe the structure of the system, including the relationships between classes, system components, and user access within learning activities. The class diagram of the DliLearn LMS is presented in Figure 2.

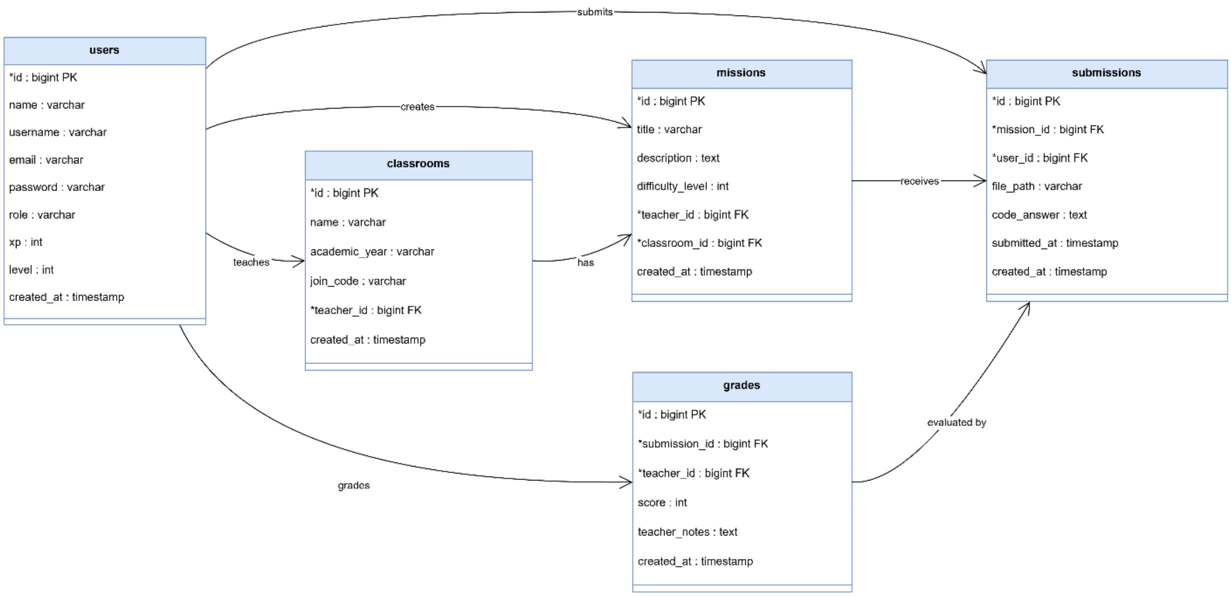


Figure 2. Simplified Class Diagram of DliLearn LMS

The user's entity represents the main actors in the system, including teachers and students. Teachers are connected to the classrooms and missions' entities because they are responsible for managing classes and creating learning missions. The classrooms entity represents the learning space where students participate in algorithm and programming learning, while the mission's entity represents structured learning activities designed according to the stages of

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

Problem-Based Learning. Through missions, students are directed to understand contextual problems, analyze information, develop solutions, and complete programming-related tasks.

The submissions entity represents students' work submitted after completing learning activities or problem-solving tasks. This component supports the development and presentation stage in Problem-Based Learning, where students submit their solutions, including files and code answers. Meanwhile, the grades entity represents teacher assessment and feedback on students' submissions. This entity supports the evaluation stage by allowing teachers to assess students' work and provide notes related to their problem-solving process.

From an educational technology perspective, these system components support the implementation of Problem-Based Learning, the Deep Learning approach, collaboration, and critical thinking activities. The mission's entity supports meaningful learning by presenting contextual algorithm and programming problems that require students to connect concepts with problem-solving situations. The submissions entity supports mindful learning by encouraging students to organize their ideas, write code, and explain their solutions. The grades entity supports reflection and evaluation through teacher feedback, which helps students review the quality of their work and improve their understanding. In relation to critical thinking, the system supports learners in interpreting problems, analyzing information, drawing conclusions, explaining solutions, and evaluating the accuracy of their program logic.

In addition, the development of teaching modules was based on the 2024 Learning Outcomes (Capaian Pembelajaran/CP) for Phase F Informatics subjects in the Algorithm and Programming element, adjusted to the learning needs at SMAN 1 Katapang. The learning materials included an introduction to algorithm and programming, data types, variables, constants, input and output, operators, and control structures such as if-else and switch-case statements. The learning model applied was Problem-Based Learning integrated with the Deep Learning approach using LMS as the learning media.

3.1.3 Development Stage

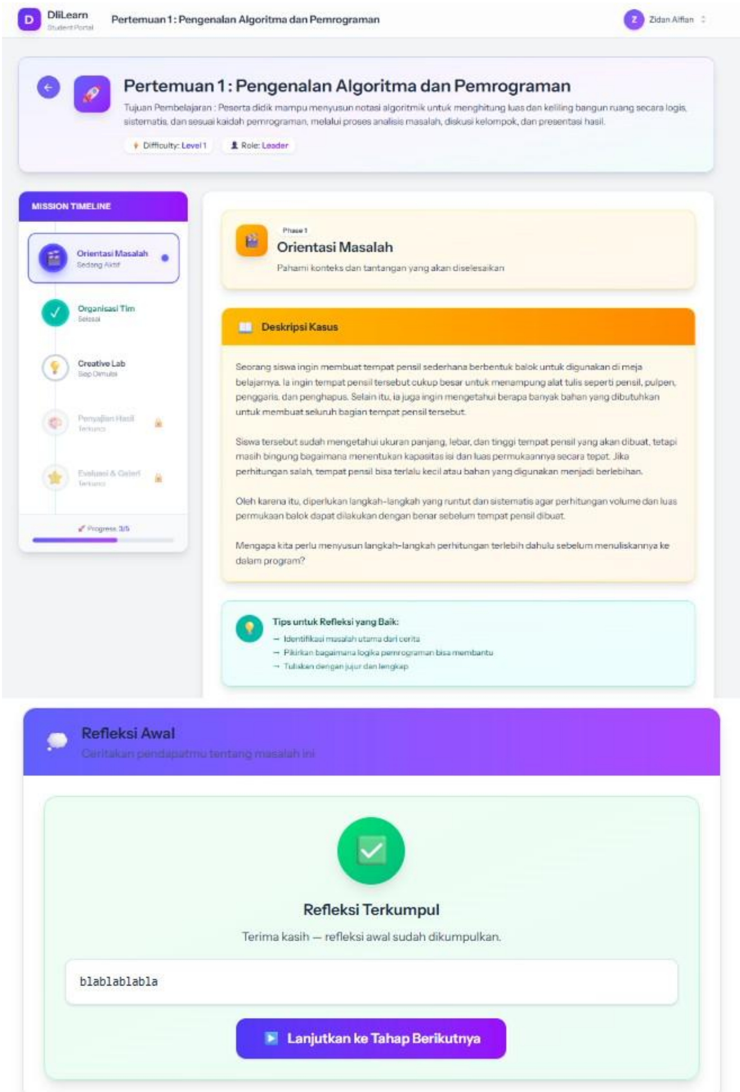
To support the realization of the Problem-Based Learning (PBL) model with the Deep Learning approach, the DliLearn LMS was developed by aligning each learning stage with the PBL syntax. Each learning stage was designed to enable students to engage in the learning process gradually, starting from understanding problems, organizing group work, conducting investigations, developing programming solutions, presenting work results, and finally evaluating and reflecting on the problem-solving process.

In the DliLearn LMS, each PBL stage was implemented through learning activities that were connected to the principles of Deep Learning, namely meaningful, mindful, and joyful learning. This alignment was intended to ensure that the LMS functioned not only utilized as a pedagogical vehicle for presenting learning materials but also as a learning environment that supported collaboration, problem solving, programming practice, and critical thinking activities. The integrated mapping of PBL stages, Deep Learning principles, LMS features, and targeted critical thinking indicators is showcased in Table 3.

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

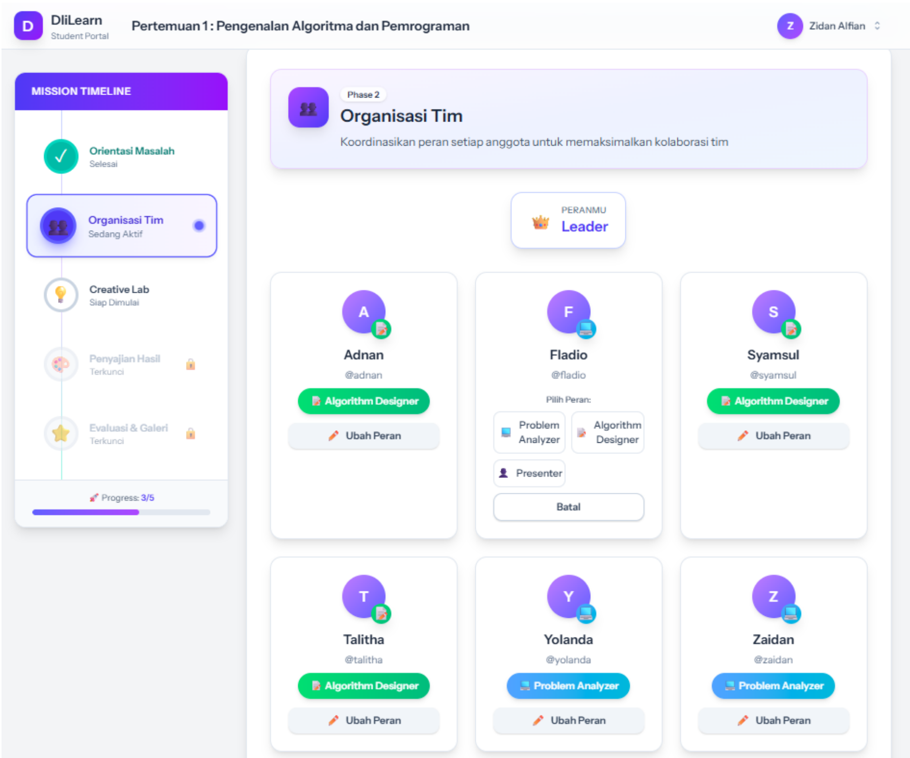
Table 3. Integration of PBL Stages, Deep Learning Principles, LMS Features, and Critical Thinking Indicators

PBL Stages	Deep Learning Principles	LMS Features / Activities	Targeted Critical Thinking Indicators
Problem Orientation	Meaningful learning, mindful learning	Students are introduced to contextual problems related to algorithm and programming. They observe the problem situation and identify important information before solving the task.	Interpretation, Analysis

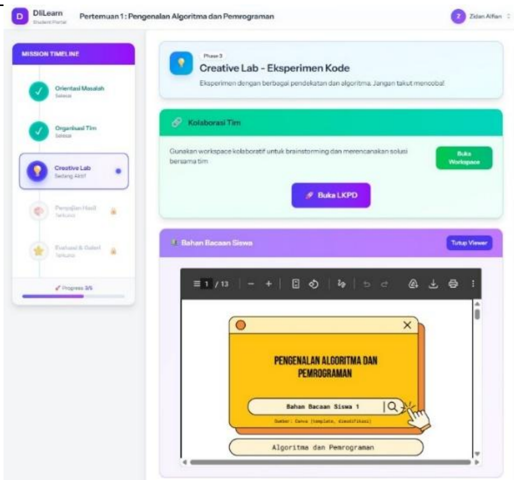
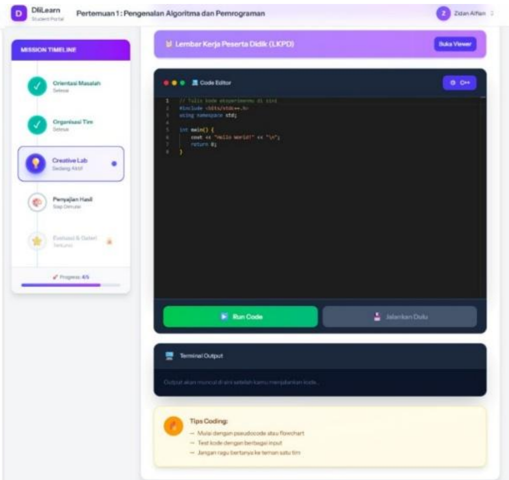
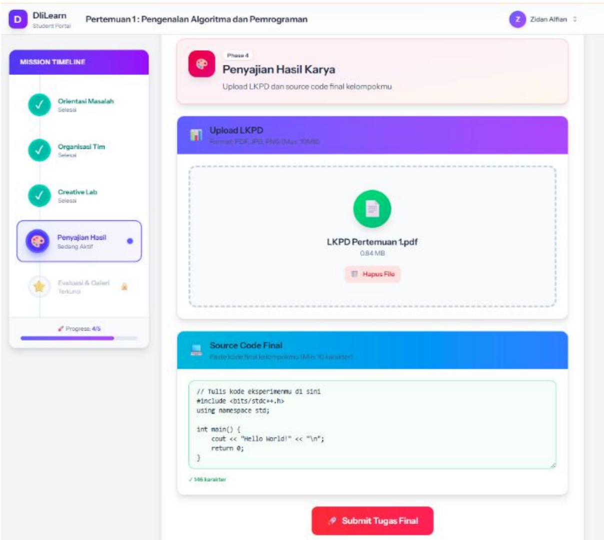


DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

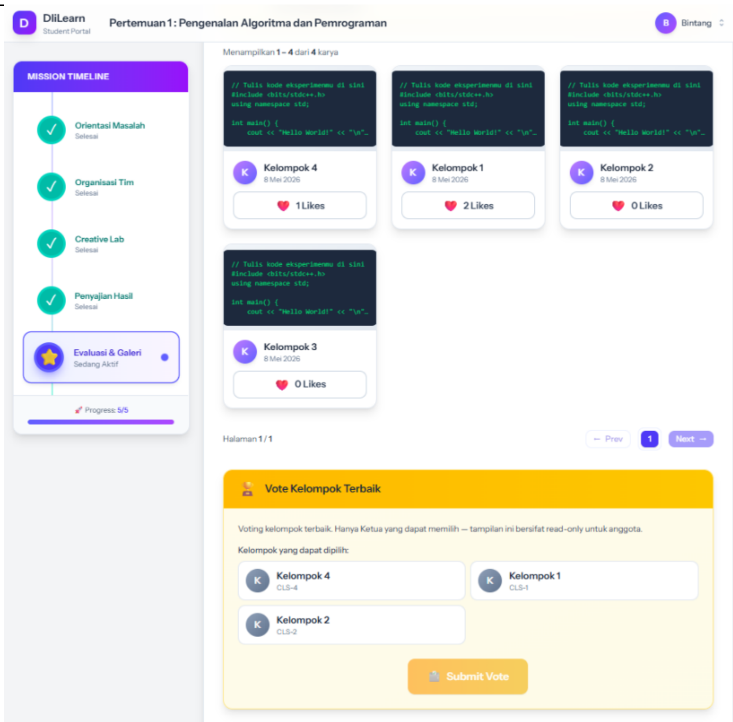
PBL Stages	Deep Learning Principles	LMS Features / Activities	Targeted Critical Thinking Indicators
Organizing Students for Learning	Mindful learning, joyful learning	Students are organized into groups and directed to understand their learning roles and responsibilities during the problem-solving process.	Self-Regulation



Guiding Individual or Group Investigation	Meaningful learning	Students explore learning materials, discuss possible solutions, develop algorithms, and conduct programming practice.	Analysis, Explanation

PBL Stages	Deep Learning Principles	LMS Features / Activities	Targeted Critical Thinking Indicators
		 	
Developing and Presenting Work Results	Meaningful learning	Students develop solutions, submit their work, and present the results of their group discussion and programming practice. 	Evaluation
Analyzing and Evaluating the Problem-Solving Process	Mindful learning, meaningful learning, joyful learning	Students evaluate the problem-solving process, receive feedback, and reflect on their learning outcomes.	Evaluation, Self-Regulation, Explanation

PBL Stages	Deep Learning Principles	LMS Features / Activities	Targeted Critical Thinking Indicators



Media expert validation was conducted to evaluate the feasibility of the developed LMS before its implementation in the learning process. The validation instrument referred to the Learning Object Review Instrument (LORI) 2.0 developed by Nesbit et al. (2009) . The LORI-based media validation covered several dimensions, including presentation design, interaction usability, accessibility, reusability, and standards compliance. The validation sheet also included qualitative comments and suggestions related to the alignment of LMS activities with the PBL syntax, such as the placement of learning videos, the clarity of student roles, and the organization of materials and worksheets in the learning flow.

Instead of reporting only the total score, the media validation results were presented by dimension to show the strongest aspects of the LMS and the aspects requiring revision. The recapitulation of the media validation results is showcased in Table 4.

Table 4. Media Validation Results Based on LORI Dimensions

LORI Dimension	Number of Items	Score Obtained	Maximum Score	Percentage	Criteria
Presentation Design	3	14	15	93.3%	Very Good
Interaction Usability	3	13	15	86.7%	Very Good
Accessibility	2	9	10	90.0%	Very Good
Reusability	1	5	5	100.0%	Very Good

LORI Dimension	Number of Items	Score Obtained	Maximum Score	Percentage	Criteria
Standards Compliance	1	4	5	80.0%	Good
Total	10	45	50	90.0%	Very Good

According to the evaluation data, the LMS garnered a cumulative score of 45 out of a maximum ideal score of 50, which corresponds to a feasibility percentage of 90%. The strongest aspect was reusability, while standards compliance obtained the lowest percentage compared with the other dimensions. Therefore, the revision process focused not only on improving the visual and technical aspects of the LMS but also on refining the learning flow to ensure that each LMS activity was aligned with the PBL syntax and supported algorithm and programming learning.

In addition to LMS development, this study also involved the development of test instruments used to assess students' critical thinking skills. The initial instrument consisted of 64 test items formulated grounded in Facione's critical thinking indicators and aligned with the Learning Outcomes (CP) and algorithm and programming materials. The instrument was first reviewed by subject matter experts from the Computer Science Education department. After expert review, the instrument was tested empirically to examine item validity, reliability, difficulty level, and discrimination index.

The item analysis showed that 45 items were valid and 19 items were invalid. The reliability test using KR-20 obtained a coefficient of 0.933, indicating that the instrument had very high reliability. The difficulty index ranged from 0.323 to 0.871, showing that the items varied from moderate to easy levels. Meanwhile, the discrimination index ranged from -0.227 to 0.900. The negative discrimination value indicated that some items in the initial instrument did not function well in distinguishing students with higher and lower abilities. Therefore, items with poor validity, inappropriate difficulty level, and weak or negative discrimination were not retained. The outline of the test instrument analysis is showcased in Table 5.

Table 5. Summary of Test Instrument Analysis

Component	Result
Initial number of items	64 items
Validity result	45 valid items and 19 invalid items
Reliability coefficient	KR-20 = 0.933
Reliability criteria	Very High
Difficulty index range	0.323 – 0.871
Difficulty level	21 easy items and 24 moderate items
Discrimination index range	-0.227 – 0.900
Final retained items	40 items
Final distribution	20 pretest items and 20 posttest items
Retention criteria	Item validity, reliability, difficulty level, discrimination index, and representation of Facione's critical thinking indicators

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

Based on the results of item validity, reliability, difficulty level, discrimination index, and the representation of Facione's critical thinking indicators, 40 items were selected as the final test instrument. These items were divided into 20 items for the pretest and 20 equivalent items for the post-test. The retained items were selected because they met the research criteria and were considered appropriate for measuring students' critical thinking skills in algorithm and programming learning.

3.1.4 Implementation Stage

The implementation phase was the phase of applying the DliLearn LMS in the learning process to identify the improvement of students' critical thinking skills. The activities were conducted with students of class XI-3 at SMAN 1 Katapang through the stages of pretest, learning implementation, and post-test. The pretest was conducted before the treatment to determine students' initial critical thinking skills in algorithm and programming materials. The pretest consisted of 20 items formulated grounded in Facione's critical thinking indicators.

The post-test was administered after all learning activities had been completed. The post-test used 20 items that were different from the pretest items but were developed with equivalent item types, indicators, and material coverage. Therefore, the post-test did not reuse the same items from the pretest. This was done to reduce the possibility of a practice effect while still maintaining the equivalence of measurement between the pretest and post-test.

After completing the pretest, students participated in learning activities using the DliLearn LMS integrated with the PBL model and the Deep Learning approach. The intervention was conducted in four meetings over two weeks, with two meetings each week. The learning schedule consisted of meetings with 3 lesson hours and 2 lesson hours, resulting in a total of 10 lesson hours for the intervention. Each meeting was adjusted to algorithm and programming materials and the stages of the Problem-Based Learning model, including problem orientation, student organization, group investigation, programming practice, presentation of results, evaluation, and reflection.

Implementation fidelity was supported by the use of teaching modules, Student Worksheets, LMS learning scenarios, and learning activities arranged according to the PBL syntax. Each meeting followed the learning flow prepared in the teaching module and was facilitated through the DliLearn LMS. In addition, the researcher documented the learning process and monitored students' activities through LMS-based tasks, group submissions, and classroom observation during the implementation. These procedures were used to ensure that the learning activities were carried out consistently with the planned intervention.

Documentation of students using the DliLearn LMS during the learning process is presented in Figure 3.

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

Figure 3. Use of DliLearn

The details of the implementation schedule, duration, and learning focus are presented in Table 6.

Table 6. Implementation Schedule and Learning Activities

Meeting	Duration	Learning Material	Main Learning Activities
1	3 lesson hours	Introduction to Algorithm and Programming	Students identified contextual problems, discussed possible solutions, and developed algorithm steps through the LMS.
2	2 lesson hours	Data Types, Variables, Constants, Input, Output, and Operators	Students analyzed problems, developed algorithms, and practiced writing simple programs collaboratively.
3	3 lesson hours	If-Else Conditional Control Structure	Students solved problems using if-else structures, conducted coding practice, and presented group discussion results.
4	2 lesson hours	Switch-Case Conditional Control Structure	Students developed program solutions using switch-case structures, submitted their

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

Meeting	Duration	Learning Material	Main Learning Activities
			work, evaluated other groups' work, and conducted reflection.

After all learning activities had been completed, students were given the post-test to identify the improvement of their critical thinking skills after the implementation of DliLearn-assisted PBL with a Deep Learning approach.

3.1.5 Evaluation Stage

To analyze the research data, the normality test, the Wilcoxon Signed Rank Test, and the N-Gain calculation were employed. Specifically, the Shapiro-Wilk method was applied within the normality test to ascertain whether the pretest and post-test data were normally distributed. The corresponding results of this normality test are detailed in Table 7.

Table 7. Results of the Pretest and Posttest Normality Test.

Type	Statistic	df	Sig.	Conclusion
Pretest	0.823	25	0.001	Not Normally Distributed
Posttest	0.870	25	0.004	Not Normally Distributed

The significance values derived from the normality test results were 0.001 for the pretest and 0.004 for the post-test. Because both significance values fell below 0.05, the pretest and post-test data were determined to be not normally distributed. Consequently, hypothesis testing proceeded using the nonparametric Wilcoxon Signed Rank Test, with the corresponding results detailed in Table 8.

Table 8. Results of the Wilcoxon Signed Rank Test.

Component	Value
Z	-4.403
Asymp. Sig. (2-tailed)	p < .001
Effect size (r)	0.88

According to the Wilcoxon Signed Rank Test results, a significant difference was observed between students' pretest and post-test scores following the implementation of learning via the Problem-Based Learning (PBL) model with the Deep Learning approach assisted by the DliLearn LMS (Z = -4.403, p < .001). Utilizing the formula $r = \frac{|Z|}{\sqrt{N}}$, the effect size was computed to be r = 0.88, denoting a large effect size and confirming that the shift between the scores was substantial. Nevertheless, given that this investigation employed a one-group pretest-post-test design, these findings ought to be interpreted as an improvement after implementation rather than definitive causal evidence. Additionally, the N-Gain calculation was performed to evaluate the level of improvement in students' critical thinking skills, with the N-Gain results compiled in Table 9.

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

Table 9. Results of the N-Gain Calculation.

Statistical Component	Average Score	Criteria
Average Pretest	62	-
Average post-test	88	-
N-Gain Score	0.708	High

The calculation results revealed an average pretest score of 62 and an average post-test score of 88. For the average N-Gain score, a value of 0.708 was obtained, placing it within the high category. Nonetheless, since this score sits only slightly above the threshold for the high criteria, a cautious interpretation of this result is recommended. Overall, these findings demonstrate that the implementation of the Problem-Based Learning model with the Deep Learning approach assisted by the DliLearn LMS led to an improvement in students' critical thinking skills.

Further analysis was executed to identify the enhancement in critical thinking skills for each Facione indicator. The N-Gain results for each indicator are detailed in Table 10.

Table 10. N-Gain Results for Each Critical Thinking Indicator.

Critical Thinking Indicator	N-Gain	Criteria
Interpretation	0.82	High
Analysis	0.39	Moderate
Inference	0.33	Moderate
Evaluation	0.28	Low
Explanation	0.89	High
Self-Regulation	0.16	Low

When examining the N-Gain results for each indicator, explanation and interpretation emerged as the areas with the highest improvements. Specifically, the explanation indicator achieved an N-Gain score of 0.89 (categorized as high), demonstrating that the implementation enhanced students' ability to clarify their reasoning, solutions, and problem-solving steps. Similarly, the interpretation indicator secured a high N-Gain score of 0.82, which reflects students' enhanced capacity to understand and identify important information from the problems presented throughout the learning process.

In contrast, the analysis indicator and the inference indicator recorded lower gains, with N-Gain scores of 0.39 and 0.33, respectively, placing both in the moderate category. These data show that while students did progress in analyzing relationships between concepts, identifying possible solutions, and drawing conclusions based on the information and problem-solving process, their growth did not match the substantial strides observed in the interpretation and explanation indicators.

Meanwhile, the evaluation indicator obtained an N-Gain score of 0.28, and the self-regulation indicator obtained an N-Gain score of 0.16. Both indicators were categorized as low. These findings indicate that students still experienced difficulties in evaluating the accuracy of solutions, considering alternative approaches, reflecting on their thinking

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

processes, and independently reviewing their learning outcomes. One possible reason for the low N-Gain scores is the limited number of test items representing these indicators, making the scores more sensitive to item difficulty and students' individual responses. In addition, evaluation and self-regulation are higher-order critical thinking indicators that require more intensive practice, such as debugging, comparing alternative solutions, peer review, and structured reflection.

Therefore, future learning scenarios using DliLearn need to provide more activities that explicitly train students to evaluate program logic, identify errors, compare different problem-solving strategies, and reflect on their own learning process. Future studies are also recommended to add more test items for evaluation and self-regulation indicators so that these aspects can be measured more comprehensively and reliably.

In addition to the analysis of students' critical thinking improvement, this study also examined students' responses toward the use of DliLearn to determine how they perceived the LMS after the learning implementation. Furthermore, student responses toward the DliLearn LMS were analyzed using a TAM-based questionnaire after the implementation. The questionnaire was completed by 25 students and consisted of 20 items using a five-point Likert scale. Therefore, the maximum possible score was 2500, obtained from 25 respondents, 20 items, and a maximum score of 5 for each item. The total score obtained was 2116, with an average score of 4.23 out of 5, equivalent to 84.6%. This result indicates that students' responses toward the DliLearn LMS were in the very good criteria, as presented in Table 11.

Table 11. Student Response Results Based on TAM Dimensions

No	TAM Dimension	Number of Items	Maximum Score	Obtained Score	Percentage	Criteria
1	Perceived Usefulness	7	875	757	86.5%	Very Good
2	Perceived Ease of Use	7	875	746	85.3%	Very Good
3	Attitude Toward Using	3	375	316	84.3%	Very Good
4	Behavioral Intention to Use	3	375	297	79.2%	Good
	Total	20	2500	2116	84.6%	Very Good

Based on Table 11, the perceived usefulness dimension obtained a percentage of 86.5%, categorized as very good. This result indicates that students perceived DliLearn as useful in supporting their understanding of algorithm and programming materials and facilitating the learning process. The perceived ease of use dimension obtained a percentage of 85.3%, was also classified under the 'very good' category, which demonstrates that students considered the LMS easy to use, with clear instructions and accessible learning features.

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

The attitude toward using dimension obtained a percentage of 84.3%, categorized as very good. This result shows that students had a positive attitude toward the use of DliLearn because it made learning more engaging and enjoyable. Meanwhile, the behavioral intention to use dimension obtained a percentage of 79.2%, categorized as good. This indicates that most students showed interest in continuing to use DliLearn and recommending it as a learning medium. Overall, the TAM results suggest that DliLearn received positive responses from students and was considered useful, easy to use, and supportive of algorithm and programming learning.

3.2 Discussion

Based on the findings, students' critical thinking skills improved after the implementation of the DliLearn LMS in algorithm and programming learning, as indicated by the increase in average scores, the Wilcoxon Signed Rank Test results, and the N-Gain scores. However, because this study used a one-group pretest-post-test design, the results should be interpreted as improvement after implementation rather than definitive causal evidence. The learning process using DliLearn provided students with structured activities aligned with the PBL model and the Deep Learning approach, including problem orientation, group investigation, programming practice, presentation of work results, feedback, and reflection.

The improvement in interpretation and explanation indicators may be related to the problem orientation and presentation activities provided in the LMS. At the beginning of each learning mission, students were introduced to contextual problems in algorithm and programming. This activity encouraged students to identify important information, understand the problem situation, and determine what needed to be solved. These activities supported the interpretation indicator. Furthermore, students were required to present their work results and explain their problem-solving steps, which likely contributed to the high improvement in the explanation indicator. Through this process, students were trained to communicate their reasoning, algorithmic steps, and programming solutions more clearly.

The moderate improvement in analysis and inference indicators may be related to group investigation and programming practice activities. During the learning process, students discussed possible solutions, analyzed relationships between input, process, and output, and developed algorithms before implementing them into program code. The Creative Lab and coding practice activities allowed students to test their ideas and observe whether their solutions produced the expected output. These activities supported students in analyzing program logic and drawing conclusions based on the results of their work. However, the improvement was still moderate, indicating that students still needed more intensive practice in comparing alternative solutions and justifying their conclusions.

The low improvement in evaluation and self-regulation indicators is also an important finding. Evaluation and self-regulation are higher-order critical thinking indicators that require students to evaluate the accuracy of solutions, identify errors, consider alternative strategies, and reflect on their own thinking processes. In this study, these indicators may not have improved optimally because the learning activities provided limited opportunities for structured debugging, peer review, and individual reflection. In addition, the number of test items representing evaluation and self-regulation was limited, making the N-Gain results for these indicators more sensitive to item difficulty and individual student responses. Therefore,

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

future learning scenarios using DliLearn should provide more explicit activities that train students to evaluate program logic, compare different solution strategies, identify errors in code, and reflect on their learning process.

The outcomes of this investigation closely align with prior research conducted by Aprilianti & Siswandari (2024), who showed that the PBL model can enhance students' critical thinking skills in computer-based learning. The findings also support Ismia & Ekohardi (2025), who found that LMS-assisted PBL can help improve students' critical thinking skills. In this study, the integration of LMS and PBL provided a structured learning environment where students could engage in problem-solving activities, collaborate with peers, submit their work, and receive feedback. These activities are relevant to algorithm and programming learning, which requires logical reasoning, analytical thinking, and systematic solution development.

The enhancement in students' critical thinking skills was also supported by the Deep Learning approach. The use of contextual problems, group discussion, coding practice, and reflection encouraged students to understand concepts more meaningfully rather than merely memorizing syntax or programming procedures. This condition aligns with the studies conducted by (Maelasari & Lusiana, 2025) and Ardhana et al. (2025), which stated that the Deep Learning approach can support conceptual understanding, critical thinking skills, and students' active involvement in learning. In this study, meaningful learning was reflected in the use of contextual algorithm and programming problems, mindful learning was reflected in students' analysis and reflection activities, and joyful learning was supported by interactive LMS-based activities and collaborative learning.

Furthermore, the use of LMS in this study was relevant to Mayer (2001) Cognitive Theory of Multimedia Learning (CTML), which explains that demonstrating that instructional delivery is most effective when it leverages dual-channel representations, specifically visual and textual elements. The DliLearn LMS presented learning materials, tasks, and programming activities in a structured digital environment, which helped students access learning resources and participate in problem-solving activities more easily. This was also supported by the TAM-based student response results, which showed that students perceived DliLearn as useful, easy to use, and supportive of algorithm and programming learning.

However, this study has several methodological limitations. First, the investigation used a one-group pretest-post-test design without a control group. Therefore, the enhancement in students' critical thinking skills cannot be attributed solely to the implementation of DliLearn-assisted PBL with a Deep Learning approach. The absence of a control group may create threats to internal validity, such as maturation and history effects. Maturation may occur because students' improvement could be influenced by their natural development, increased familiarity with the material, or learning experiences gained over time. Meanwhile, history effects may occur if students were exposed to other learning activities, teacher explanations, or external resources during the intervention period that also contributed to their improvement. Although the post-test used different but equivalent items from the pretest to reduce practice effects, the one-group design still limits causal interpretation. Therefore, the findings should be interpreted as improvement after implementation rather than definitive evidence of causal effectiveness.

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

Second, the sample size was relatively small, involving only 25 students from one class in a single school. Third, the intervention was executed over a relatively short period, consisting of four meetings in two weeks. Fourth, the implementation may have been influenced by the teacher's facilitation, classroom conditions, students' digital literacy, and internet connectivity. Finally, the data were mainly acquired from pretest-post-test scores and questionnaire responses. Future studies are recommended to use a quasi-experimental or experimental design involving a control group, larger samples, longer intervention duration, and additional data sources such as classroom observation, interviews, learning analytics, and analysis of students' programming products.

Therefore, future research should involve a control group, a larger sample, longer intervention duration, and implementation in different school contexts. Future studies should also strengthen learning activities that specifically target evaluation and self-regulation, such as structured debugging tasks, peer review, comparison of alternative solutions, and guided reflection. In addition, more test items representing evaluation and self-regulation indicators should be developed to measure these aspects more comprehensively and reliably.

4. CONCLUSION

Following the implementation of DliLearn-assisted Problem-Based Learning with a Deep Learning approach in algorithm and programming learning, the findings indicated a clear improvement in students' critical thinking skills. This advancement was evidenced by the average score rising from 62 in the pretest to 88 in the post-test. Furthermore, the Wilcoxon Signed Rank Test confirmed a significant difference between the pretest and post-test scores ($Z = -4.403$, $p < .001$), accompanied by a large effect size of $r = 0.88$. Additionally, an average N-Gain score of 0.708 was achieved, placing the growth in the high category, though this outcome warrants a cautious interpretation since it sits only slightly above the high-criteria threshold.

Based on the N-Gain results for each critical thinking indicator, the highest improvements were found in explanation and interpretation. These findings suggest that students became more capable of identifying important information from problems and explaining their reasoning, solutions, and problem-solving steps. However, the evaluation and self-regulation indicators showed low improvement, indicating that students still need more structured learning activities that train them to evaluate program logic, compare alternative solutions, identify errors, and reflect on their learning process.

The student response results also showed positive acceptance of DliLearn. The TAM-based questionnaire obtained a total score of 2116 out of 2500, equivalent to 84.6%, which was categorized as very good. This indicates that students perceived DliLearn as useful, easy to use, and supportive of algorithm and programming learning.

This study has several limitations, including the absence of a control group, a relatively small sample size, a short intervention duration, a single-school context, possible teacher effect, and reliance on pretest-post-test scores and questionnaire responses. Because this study used a one-group pretest-post-test design without a control group, the findings should

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

be interpreted cautiously due to possible internal validity threats such as maturation, history effects, and other learning experiences outside the intervention. Therefore, future studies are recommended to involve a control group, use a larger sample, extend the intervention duration, and implement the LMS in different school contexts. Future research should also strengthen learning activities and assessment items related to evaluation and self-regulation indicators.

ACKNOWLEDGMENT

The authors wish to extend their deepest gratitude to the supervisors for their invaluable guidance, support, and suggestions throughout the completion of this study. Sincere appreciation is also directed toward SMAN 1 Katapang for granting the necessary permission and support during the research process. Furthermore, particular recognition is given to the Informatics teacher and the students of class XI-3 for their active participation in this investigation. Lastly, the authors would like to acknowledge all lecturers, validators, and parties who contributed to the development and implementation of the DliLearn LMS.

AUTHORS' NOTE

Regarding the publication of this article, the authors declare that no conflict of interest exists. Additionally, the authors confirm that the paper is entirely free of plagiarism.

REFERENCES

Aprilianti, A. R., & Siswandari. (2024). *Keefektifan Model Problem Based Learning (PBL) untuk Meningkatkan Kemampuan Berpikir Kritis Siswa SMK Akuntansi pada Pembelajaran Komputer Akuntansi*. *4*, 1255–1266.

Ardhana, D., Fajrina, S., Syamsurizal, & Fitri, R. (2025). *Implementasi problem-based learning berbasis deep learning untuk meningkatkan berpikir kritis siswa pada materi sistem ekskresi di sma*.

Damayanti, S. P., Nindiasari, H., Mustafa, A. N., Matematika, J. P., Sultan, U., Tirtayasa, A., Matematika, P., Learning, P., Review, S. L., & Pendahuluan, A. (2020). *Systematic Literature Review : Efektivitas Problem Based Learning Terhadap Peningkatan*. *5*.

Diputera, A. M., Eza, G. N., Guru, P., Anak, P., Dini, U., Pendidikan, F. I., Medan, U. N., Kebidanan, A., & Husada, M. (2024). *Memahami Konsep Pendekatan Deep Learning dalam Pembelajaran Anak Usia Dini Yang Meaningful , Mindful dan Joyful : Kajian Melalui Filsafat Pendidikan*. *10*(2), 108–120.

Facione, P. A. (1990). *Critical Thinking : What It Is and Why It Counts*.

Hidayat, F., & Nizar, M. (2021). *Model ADDIE (Analysis , Design , Development , Implementation And Evaluation) Dalam Pembelajaran Pendidikan Agama Islam Addie (Analysis , Design , Development , Implementation And Evaluation) Model In Islamic Education Learning*. 28–37.

Ismia, R., & Ekohardi. (2025). *Pengembangan LMS PBL untuk Meningkatkan Kompetensi Network Engineer dan Berpikir Kritis Siswa TKJ di SMK Negeri 2 Lamongan*. *10*(3), 131–

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535

143.

Maelasari, N., & Lusiana. (2025). *Efektivitas Deep Learning Dalam Pembelajaran : Sebuah Kajian Systematic Literature Review (SLR)*. *13*(2), 298–305.

Mahabu, F., Subhan, M., Pramadita, O. I., Fahriza, A., & Ekabudi, A. (2025). *Pemanfaatan Learning Management System (LMS) Untuk Meningkatkan Efektifitas Pembelajaran Jurnal Teknologi Pendidikan Dan Pembelajaran (JTPP)*. *03*(01), 27–34.

Mayer, R. E. (2001). Multimedia Learning. *Cambridge: Cambridge University Press.*

Nesbit, J., Belfer, K., & Leacock, T. (2009). *Learning Object Review Instrument (LORI)*. 1–11.

Prasetyo, F., & Kristin, F. (2020). *Pengaruh Model Pembelajaran Problem BasedLearning dan Model Pembelajaran Discovery Learning terhadap Kemampuan Berpikir Kritis Siswa Kelas 5 SD*. https://doi.org/10.30997/dt.v7i1.2645

Qohar, A., & Sulandra, I. M. (2021). *Analisis Kemampuan Berpikir Kritis Siswa SMA Kelas X Dalam Memecahkan Masalah SPLTV*. *05*(02), 909–922.

Qurrota, F., Tamami, A., Shohib, M. W., Nur, M., & Maksum, R. (2025). *The effect of deep learning and problem based learning on active and independent learning with mediation variable*. *5*(2), 613–628.

Rozali, A., Irianto, D. M., & Yuniarti, Y. (2022). *Kajian problematika teacher centered learning dalam pembelajaran siswa studi kasus: sdn dukuh, sukabumi*. *05*(01), 77–85.

Sugiyono. (2019). *Metode Penelitian Kuantitatif, Kualitatif dan R & D*.

Venkatesh, V., & Davis, F. D. (2000). *A Theoretical Extension Of The Technology Acceptance Model: Four Longitudinal Field Studies*. 186–204.

Zebua, J. Y., Zega, Y., & Telaumbanua, Y. N. (2024). *Analisis Kemampuan Berpikir Menyelesaikan Soal Matematika Kritis Siswa dalam*. *13*(001), 587–594.

DOI: https://doi.org/10.17509/edsence.v8i1.100891
p-ISSN:2685-2489 | e-ISSN:2685-2535