

Effects of an AI-Supported LMS on Academic Achievement and Learner Perceptions in Programming Education

Programlama Eğitiminde Yapay Zeka Destekli Bir Öğrenme Yönetim Sisteminin Akademik Başarı ve Öğrenci Algıları Üzerindeki Etkileri

Nihan Arslan Namlı¹ 

¹Assistant Professor Dr. Erzin Organized Industrial Zone Vocational School, İskenderun Technical University, Hatay, Türkiye

Makale Bilgileri

Geliş Tarihi (Received)

19.06.2026

Kabul Tarihi (Accepted)

28.09.2026

Yayınlanma Tarihi (Published)

21.10.2026

*Sorumlu Yazar

Nihan Arslan Namlı

Department Of Computer
Technologies, İskenderun
Technical University, Kuzuculu
Campus, 1802 Nolu Sk., No:
1/A, 31600 Dörtöyl/Hatay,
Türkiye

nihan.arslannamli@iste.edu.tr

Abstract: This study investigated the academic success of an AI-supported learning system dedicated to a PHP programming course at the associate degree level and how students experienced this. Although course-embedded AI feedback is theoretically promising, evidence that jointly addresses academic achievement and learners' experience in authentic programming courses remains limited. 67 third-semester associate-degree Computer Programming vocational school students, participated in a convergent parallel mixed-methods study over four weeks. The experimental group (n =34) worked with the AI-supported LMS, while the control group (n =33) received traditional lecture-based instruction. The quantitative component employed a quasi-experimental design with experimental and control groups. Success was measured with a 20-item multiple-choice test (KR-20=.87) administered before and after the four-week intervention; semi-structured conversations with 12 students were analyzed using content analysis. After controlling for variation in pre-test scores using covariance analysis, experimental group achieved significantly higher post-test scores than those in the control group, $F(1, 64) = 5.671, p = .018, \eta^2 = .081$. In the interviews, it was noted that the developments related to instant and error-specific feedback increased their awareness of their errors and helped them to regulate their learning; however, no adaptive content suggestions emerged. The findings demonstrate that feedback quality and progress tracking are essential components of such systems. However, it is of great importance that AI-supported feedback can be used independently by students. Conflicting findings in the literature regarding this process indicate a need to reconcile divergent evidence in future studies.

Keywords: Artificial intelligence in education, programming education, academic achievement, learner perceptions

Öz: Bu çalışma, ön lisans düzeyindeki bir PHP programlama dersine yönelik YZ destekli öğrenme yönetim sisteminin (ÖYS) akademik başarıya etkisini ve öğrencilerin bu süreci nasıl deneyimlediğini incelemiştir. Dört hafta boyunca yürütülen yakınsayan paralel karma yöntem desenli çalışmaya 67 meslek yüksekokulu öğrencisi katılmıştır. Deney grubu (n = 34) YZ destekli ÖYS ile çalışırken kontrol grubu (n = 33) geleneksel anlatıma dayalı öğretim almıştır. Nicel bileşen, deney ve kontrol gruplarını içeren yarı deneysel bir desen olarak kurgulanmıştır. Başarı, uygulama öncesi ve sonrasında uygulanan 20 maddelik bir başarı testiyle (KR-20 = .87) ölçülmüş; 12 öğrenciyle yapılan yarı yapılandırılmış görüşmeler içerik analiziyle çözümlenmiştir. Ön test puanlarındaki değişkenlik kovaryans analizi ile kontrol edildikten sonra deney grubu, kontrol grubuna kıyasla anlamlı düzeyde daha yüksek son test puanları elde etmiştir, $F(1, 64) = 5.671, p = .018, \eta^2 = .081$. Görüşmelerde, anlık ve hataya özgü geri bildirimin öğrencilerin hatalarına ilişkin farkındalığını artırdığı ve öğrenmelerini düzenlemelerine yardımcı olduğu; ancak uyarlanabilir içerik önerilerinin ortaya çıkmadığı belirlenmiştir. Bulgular, geri bildirim kalitesi ve ilerleme takibinin bu tür sistemlerin temel bileşenleri olduğunu göstermektedir. Bununla birlikte, YZ destekli geri bildirimin öğrenciler tarafından bağımsız şekilde kullanılabilmesi büyük önem taşımaktadır. Bu süreçte ilişkin literatürde yer alan çelişkili bulgular, gelecekte yapılacak araştırmalarda farklı yöndeki kanıtların uzlaştırılması gereksinimine işaret etmektedir.

Anahtar Kelimeler: Eğitimde yapay zekâ, programlama eğitimi, akademik başarı, öğrenen algıları

Arslan Namlı, N. (2026). Effects of an ai-supported LMS on academic achievement and learner perceptions in programming education. *Erzincan Üniversitesi Eğitim Fakültesi Dergisi*, 28(2026), 1-11. <https://doi.org/10.17556/erziefd.1974881>

Introduction

With digitalization and the integration of new technologies, educational paradigms are undergoing a revolutionary transformation, and education is adapting more closely to the skills needed in the 21st century. At the forefront of these transformations are artificial intelligence (AI) technologies, one of the most important forces driving change in education. The mechanisms of learning processes are being influenced by AI through data analysis, machine learning, and natural language processing, making education more user-friendly and therefore data-driven (Xu & Ouyang, 2021; Tlili et al., 2023). Therefore, this ongoing digitalization of education is not only a technological innovation but also expands flexibility and scope in accessible personalized learning environments. In this context, the widespread use of AI in education is becoming one of the main components of the trends in digital transformation in education (Garzón et al., 2025).

AI-based learning systems are digital learning environments that can adjust content based on real-time student performance data analysis. Instead of LMSs, which typically provide limited automated feedback, AI-powered

systems can leverage performance data to generate more comprehensive instructional support. This new trend contributes to programming skills in programming education, where the learning process is often a complex cognitive process and an iterative problem-solving process. Recent research findings show that systems capable of recognizing and explaining students' logical and syntactic errors have the potential to promote learning by making errors visible (Ezzaim et al., 2023; Keuning et al., 2018). These systems guide students in identifying the source of their errors, enabling them to better understand programming skills while gaining problem-solving capacity, not just how correct they are.

Programming is a cognitively demanding subject in which novices struggle with syntax, debugging, and the iterative correction of errors, and the timely, error-specific feedback these difficulties call for is hard to deliver individually under ordinary classroom conditions, particularly in crowded courses (Keuning et al., 2018; Hao et al., 2022). Because such feedback operates through the mechanisms posited by formative-feedback and self-regulation theory (Shute, 2008; Hattie & Timperley, 2007; Zimmerman, 2002), course-

embedded AI-supported feedback is theoretically well positioned to address this need; yet reported effects on programming achievement remain heterogeneous across studies, and academic outcomes are rarely examined together with how learners actually experience the feedback in authentic, semester-long courses (Zheng et al., 2022; Hu, 2024).

A systematic, course-based evaluation that jointly investigates achievement and learner experience is therefore needed to clarify how such systems should be integrated into a specific programming course. The current technological paradigm necessitates the use of AI-supported learning systems, such as updating the technical infrastructure in digital learning environments. However, beyond the mere transfer of learning content, solution-oriented AI applications have evolved into dynamic instructional architectures that guide education through student academic performance metrics that transform system logs, system records, and statistical outputs into content streams and tasks.

This transformation allows for the modeling of student behavior data, the evaluation of their effectiveness, and the provision of automated feedback measures in Artificial Intelligence in Education (AIED) research; and the analysis of instructional content to suit individual performance levels. Particularly in error-oriented and process-oriented disciplines such as programming, knowledge of how to process student performance data enables the creation of a much more accurate and adaptable instructional process.

Based on different learning theories, it is possible to evaluate the impact of AI-supported feedback systems on learning. Formative feedback theory also states that timely and relevant feedback on students' own performance encourages learning (Shute, 2008). Furthermore, it emphasizes that the effect of feedback on learning is related not only to whether feedback is given, but also to when it is given and how specific it is to the student (Hattie & Timperley, 2007). In fields requiring complex cognitive processes, such as programming education, rapid feedback on the nature of student errors helps to identify and correct misunderstandings early. On the other hand, metacognition is closely related to students' active participation in the process of recognizing and correcting their own errors. In other words, it is a person's ability to monitor and control their own learning and thinking processes (Flavell, 1979). This supports students' conscious and active participation in the process of recognizing and correcting their errors. In addition, the fact that AI-supported systems provide hints and guidance instead of directly offering solutions is consistent with the basic principles of the scaffolding approach and the proximal developmental zone theory. It emphasizes that students receive temporary support during the learning process and that it encourages independent learning (Vygotsky, 1978; Wood et al., 1976). Furthermore, from a learning theory perspective, the ability of students to monitor and manage their own learning is crucial. Self-regulated learning is the ability of an individual to self-determine tasks and goals, monitor their performance (goal-oriented learning), and analyze both the process and their learning by evaluating their performance (Zimmerman, 2002). According to these principles, AI-supported LMSs address the theoretical processes responsible for supporting learning, integrating them into a single learning process by creating an approach with features such as personalized feedback, guided support, and progress tracking.

Although research on AI in education is expanding rapidly, three interrelated needs remain inadequately met and motivate

the present study. First, much of the existing work relies on general-purpose AI tools that are not tied to a specific course, yet effective programming instruction requires support aligned with the objectives, content sequence, and characteristic error patterns of a particular course, which calls for AI-supported LMSs designed for concrete teaching contexts (Xu & Ouyang, 2021; Topham et al., 2025). Second, prior work has tended to foreground the technical performance of isolated algorithmic modules, such as automated code evaluation and error detection, whereas instructional practice requires understanding how these modules operate together across a complete teaching cycle that spans content delivery, adaptive feedback, and the assessment of learning outcomes (Chen et al., 2025; Keuning et al., 2018). Third, evidence on academic performance in AI-powered environments has been produced predominantly through quantitative designs, whereas designing and refining such systems also requires insight into how learners interpret and experience them, an aspect that has received comparatively less attention (Kölemen & Şahin, 2026; Sezer & Korucu, 2019). Taken together, these needs for course-specific design, an integrated instructional perspective, and a learner-centred account of the experience motivate the present study. Addressing this gap, the present study is among the few to evaluate, within a single authentic PHP programming course, both the achievement effects and the learner experience of a course-embedded AI-supported LMS rather than the isolated algorithmic performance or general-purpose tools that characterize much prior work. The aim of this research is to address these needs by developing an AI-powered LMS for PHP programming using a convergent parallel mixed methods approach. The specific contribution of this study is to provide integrated quantitative and qualitative evidence from a single, course-embedded AI-supported LMS, linking achievement outcomes to learners' lived experience of the feedback an integration rarely offered in prior work. This study also focuses on evaluating the benefits the system provides to students and understanding their thoughts and real-life experiences regarding the system's operation. In line with this holistic approach, this research problem statement helps answer the following questions:

1. Is there a significant difference in academic achievement scores between students using the AI-supported LMS and those receiving traditional instruction?
2. Is there a significant difference between pre-test and post-test academic achievement scores of students using the AI-supported LMS?
3. How do students experience the process of using the AI-supported LMS?
4. How do students assess the feedback mechanisms provided by the LMS?
5. How do students perceive the contribution of the AI-supported LMS to their learning process?
6. What are students' views regarding the content and technical operation of the LMS?
7. What challenges do students encounter while using the AI-supported LMS, and what suggestions do they have for improvement?

Method

This study adopts a mixed-methods research design to examine the impact of an AI-supported LMS for programming education on students' academic achievement and perceptions of the system. A convergent parallel mixed-methods design was applied to collect and interpret both quantitative and

qualitative data holistically within the same study (Creswell & Plano Clark, 2022). This design involves the simultaneous collection and analysis of quantitative and qualitative data, followed by the integration and interpretation of the findings. Therefore, quantitative data were obtained from the experimental process, and qualitative data from semi-structured interviews. The results were combined to provide a complete answer to both quantitative and qualitative research questions (RQ).

Participants were verbally informed that they would use an AI-supported learning system and took part voluntarily; however, the informed-consent process did not explicitly state that the code and queries submitted through the system would be transmitted to and processed by a third-party service (the Google Gemini API). The information sent to the service included the programming code and task-related queries entered by students together with their names, which constitute directly identifying personal data. The study group consisted of 67 students who were accessible to the researcher and voluntarily participated in a PHP programming course at a vocational school of a state university and were selected using a convenience sampling method (Creswell & Plano Clark, 2022; Yıldırım & Şimşek, 2021). The participants were enrolled in the Computer Programming associate-degree program and were in their third semester; their ages ranged from 18 to 22 years. Of the 67 students, 39 (58.2%) were male and 28 (41.8%) were female, and 45 (67.2%) reported prior programming experience whereas 22 (32.8%) did not.

Purposive sampling was used for qualitative data collection; participants were selected from the experimental group based on two criteria: voluntary participation and active system interaction during the intervention period, which was determined through the instructor's observation of regular attendance and engagement with the system rather than through automated usage logs. Experimental and control groups were formed during the research process. Due to the need to maintain the natural structure of the existing classroom arrangement, students could not be randomly assigned to groups; therefore, a quasi-experimental design was used for the study. An academic achievement test was used as a pre-test to compare the programming levels of the groups before the intervention. Quantitative analyses were performed to observe the experimental and control groups, while participants in the qualitative data collection sample were selected from the experimental group on a voluntary basis.

This study utilized an LMS designed by the researcher and enhanced with AI-powered interactive elements, hands-on exercises, homework workflows, and automated feedback platforms. The experimental group participated in the lessons through the AI-based LMS, while the control group continued with the same course content using traditional teaching methods. The teaching process was designed by the same instructor, and both groups received the same number of hours. The intervention was implemented over four weeks, with three class hours per-week.

The AI-powered LMS designed in this research aims to facilitate students' access to different learning modes, course resources, exercises, and interaction with the AI-powered feedback system on both mobile and web platforms. It also includes a mobile application interface that allows students to participate in the study anytime, anywhere. The AI component of the LMS is powered by Google Gemini 2.5 Flash API. A predefined system command was used to configure the model's instructional behavior: the model was tasked with acting like a PHP programming instructor, identifying syntactic and logical

errors in the code submitted by the student, explaining the nature and cause of each error in simple language, and providing guiding hints instead of direct solutions. The model was accessed and used between March and April 2026 with the provider's default generation parameters. An example of the system prompt used to configure the model is as follows: "You are a PHP programming instructor. When a student submits code, identify the syntactic and logical errors it contains and, for each error, explain its nature and cause in simple language that a beginner can understand. Do not provide the corrected code or the direct solution; instead, give guiding hints that help the student locate and fix the error independently." Each time a student submitted code, the submission was transmitted to the Google Gemini 2.5 Flash model along with this system command, and the generated feedback was returned to the student in real time. This architecture ensured that the feedback was dynamically generated and specific to each student's submission. In the mobile environment, users could securely log in and register, view course materials through their personal accounts, and receive real-time support using an AI-supported interactive module (Figure 1).

Thanks to the integrated structure of mobile and web environments, learning data obtained from students is centrally collected, and the artificial intelligence module processes each submitted code example, generating real-time, context-specific formative feedback. The system did not adapt the selection or order of instructional content to the students; the difference among students was the feedback generated for each submission. The student's academic success in the programming course was assessed with an academic achievement test consisting of 20 multiple-choice questions covering PHP programming topics, created by the researcher. Content validity was ensured through expert review by two domain experts specializing in programming education and measurement and evaluation; the questions were revised based on feedback prior to implementation. Due to the binary structure of the questions, the Kuder-Richardson Formula 20 (KR-20) was used to estimate internal consistency. The resulting coefficient is 0.87, indicating high internal consistency (Nunnally & Bernstein, 1994; Tabachnick & Fidell, 2019). In the qualitative section, a semi-structured interview tool was used to investigate students' experiences and perceptions regarding the AI-supported LMS.

Considering the fundamental issues, interview questions were developed to examine the system's usage processes, feedback mechanisms, contribution to the learning process and academic achievement, system design and content structure, challenges encountered, and suggestions for improvement. The interview form was subjected to expert review before data collection. Qualitative data were collected through individual interviews with 12 participants (P1–P12) selected from the experimental group via purposive sampling. Interviews lasted an average of 28 minutes and were audio-recorded with the participants' consent. All recordings were transcribed and subjected to content analysis. Coding was performed independently by two researchers, and inter-rater reliability was obtained using Miles and Huberman's (1994) consensus formula with a coefficient of 0.88. This coefficient was computed as the ratio of the number of coding agreements to the total number of coding decisions; the few disagreements were resolved through discussion until consensus was reached. As a result of the analysis, five main themes emerged, and interviews continued until theoretical saturation was reached, with no new codes or themes emerging (Creswell & Poth, 2018).

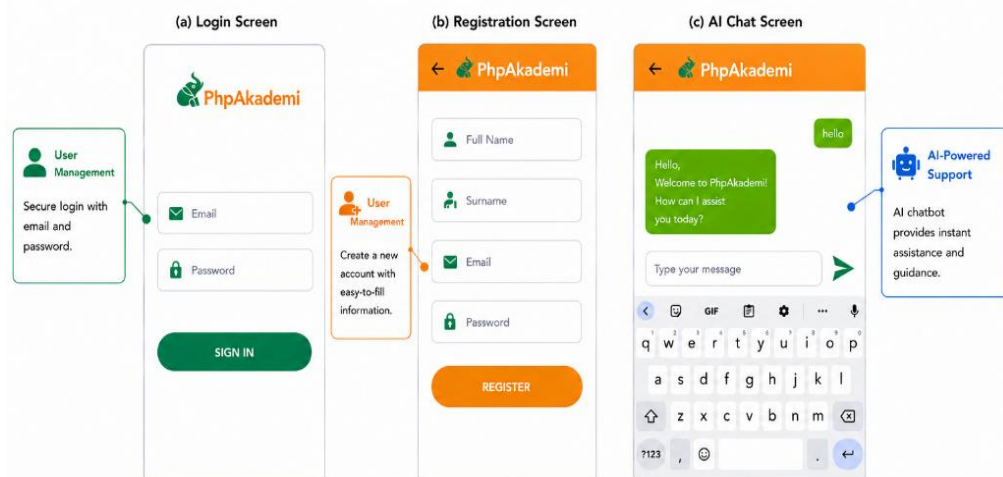


Figure 1. Mobile application login interface, user registration interface, and chat support screen

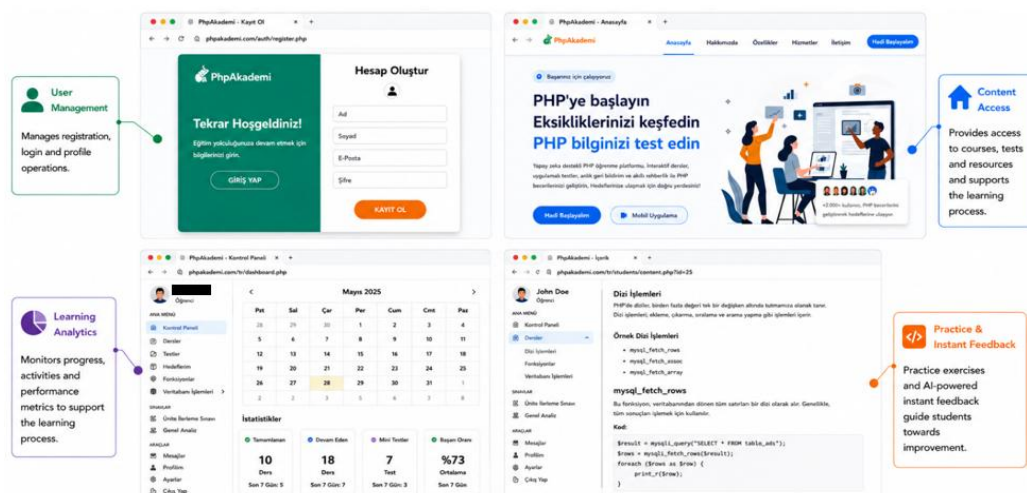


Figure 2. Web-based system registration interface, homepage, student control panel, and learning progress tracking screen

Descriptive statistics were initially performed for the analysis of quantitative data. Skewness and kurtosis coefficients were examined to determine whether the data were normally distributed and whether parametric assumptions were met. Parametric statistical tests were applied to compare pre-test and post-test scores between the experimental and control groups. To control for potential fundamental differences between groups, pre-test scores were treated as covariates and analysis of covariance (ANCOVA) was performed. Effect size values were calculated to interpret the effect size on academic achievement (Cohen, 1988). In the qualitative analysis, a deductive content analysis was conducted, in which the main themes were derived from the research questions and the semi-structured interview protocol, whereas the subcategories and codes under each theme were developed inductively from the interview data (Hsieh & Shannon, 2005; Merriam, 2002). Similar codes were grouped, themes and sub-themes were created as the data were analyzed. The created themes were then combined with the

quantitative results for a comprehensive and integrated interpretation of the findings.

Results

This section presents the quantitative findings for the experimental group ($n = 34$), which used an AI-supported LMS, and the control group ($n = 33$), which received traditional instruction.

Comparison of Academic Achievement Scores Between Experimental and Control Groups

Descriptive Statistics

Descriptive statistics for the pre-test and post-test academic achievement scores of both groups are presented in Table 1. The mean of the pre-test for the experimental group was $M = 47.35$ ($SD = 10.17$) and for the control group was $M = 45.15$ ($SD = 11.21$). The mean score of the experimental group on the post-test after the intervention was $M = 56.18$ ($SD = 12.00$) while the mean score of the control group was $M = 49.55$ ($SD = 10.49$).

Table 1. Descriptive statistics for pre-test and post-test academic achievement scores by group

Group	n	Measure	M	SD	Skewness	Kurtosis
Experimental	34	Pre-test	47.35	10.17	-0.31	-0.77
		Post-test	56.18	12.00	-0.28	0.63
Control	33	Pre-test	45.15	11.21	-0.22	-0.45
		Post-test	49.55	10.49	-0.06	-0.92

Note. M = arithmetic mean; SD = standard deviation. Skewness and kurtosis values were computed using the unbiased formula.

Table 2. Independent-samples t-test results for pre-test scores

Group	n	M	SD	df	Levene F	t	p	Cohen's d
Experimental	34	47.35	10.17	65	0.016	0.842	.403	0.21
Control	33	45.15	11.21					

Note. M = arithmetic mean; SD = standard deviation. Levene F refers to the homogeneity of variance test for pre-test scores. Cohen's d: 0.20 = small, 0.50 = medium, 0.80 = large effect (Cohen, 1988). 95% CI = [-3.02, 7.42].

Table 3. ANCOVA results and adjusted group means for post-test scores

Source of Variance	SS	df	MS	F	p	η^2
Corrected Model	3715.600	2	1857.800	—	—	—
Pre-test (Covariate)	3272.730	1	3272.730	41.904	< .001	.396
Group (Experimental/Control)	442.870	1	442.870	5.671	.018	.081
Error	4998.400	64	78.100	—	—	—
Corrected Total	8714.000	66	—	—	—	—

Adjusted post-test means			
Group	n	M	Adjusted M
Experimental	34	56.18	55.46
Control	33	49.55	50.29

Note. SS = sum of squares; df = degrees of freedom; MS = mean square; η^2 = partial eta squared. Effect size benchmarks: $\eta^2 \geq .01$ small, $\geq .06$ medium, $\geq .14$ large (Cohen, 1988). SE = standard error; difference between adjusted means = 5.17 points.

Table 4. Paired-samples t-test results for pre-test and post-test scores by group

Group	Measure	n	M	SD	df	t	p	Cohen's d
Experimental	Pre-test	34	47.35	10.17	33	4.890	< .001	0.839
	Post-test		56.18	12.00				
Control	Pre-test	33	45.15	11.21	32	3.053	.005	0.531
	Post-test		49.55	10.49				

Note. M = arithmetic mean; SD = standard deviation. Cohen's d: 0.20 = small, 0.50 = medium, 0.80 = large (Cohen, 1988). Experimental group 95% CI (difference scores) = [5.15, 12.49]. Control group 95% CI (difference scores) = [1.46, 7.33].

Normality and Homogeneity of Variance Assumption Tests

Before parametric analyses, normality was confirmed using the Shapiro-Wilk test. The results were as follows: experimental group pre-test, $W = 0.955$, $p = 0.178$; experimental group post-test, $W = 0.970$, $p = 0.472$; control group pre-test, $W = 0.961$, $p = 0.270$; control group post-test, $W = 0.956$, $p = 0.201$. When these results are evaluated together with the skewness and kurtosis coefficients presented in Table 1, it shows that the normality assumption is met for all measurements ($p > 0.05$). Homogeneity of variance was examined using the Levene test, and the following values were obtained: $F(1, 65) = 0.016$, $p = .901$ for pre-test scores and $F(1, 65) = 0.078$, $p = .781$ for post-test scores; this confirmed that the assumption of homogeneous variance was met in both groups.

Pre-Test Equivalence Between Groups

An independent samples t-test was conducted to examine whether the two groups were equivalent in terms of academic achievement before the intervention; the results are presented in Table 2. There was no statistically significant difference between the pre-test scores of the groups, $t(65) = 0.842$, $p = 0.403$, Cohen's $d = 0.21$ [95% CI = -3.02, 7.42]. Since the mean difference between the groups was small, it indicated that there was no statistically significant difference between the two groups' academic levels at the start of the study.

Analysis of Covariance (ANCOVA): Post-Test Group Comparison

In addition, the assumptions specific to the analysis of covariance were tested. The relationship between the covariate (pre-test) and the dependent variable (post-test) was linear within both groups (experimental $r = .56$, control $r = .71$; both $p < .01$), satisfying the linearity assumption. The homogeneity of regression slopes was examined through the group $\times$ pre-test interaction term, which was non-significant, $F(1, 63) = 0.00$, p

$= .984$, indicating that the regression slopes did not differ across groups and that the covariate operated equivalently under both conditions. These results confirmed that the data met the assumptions required for the analysis of covariance.

ANCOVA was performed using pre-test scores as covariates to compare the post-test academic achievement of the groups, and the results are presented in Table 3. When pre-test scores were statistically controlled, the difference between the post-test scores of the experimental and control groups was found to be significant, $F(1, 64) = 5.671$, $p = .018$, $\eta^2 = .081$. The covariate was also found to be significantly related to post-test performance, $F(1, 64) = 41.904$, $p < .001$, $\eta^2 = .396$. The corrected post-test mean of the experimental group ($M_{adj} = 55.46$, $SE = 1.52$) was found to be 5.17 points higher than the corrected mean of the control group ($M_{adj} = 50.29$, $SE = 1.54$).

Pre-Test to Post-Test Change Within Groups

Paired-samples t -tests were conducted to examine within-group change from pre-test to post-test for each group separately; results are presented in Table 4.

The difference between the pre-test ($M = 47.35$, $SD = 10.17$) and post-test ($M = 56.18$, $SD = 12.00$) scores of the experimental group was found to be statistically significant, $t(33) = 4.890$, $p < 0.001$, Cohen's $d = 0.839$ [95% CI = 5.15, 12.49]. The difference between the pre-test ($M = 45.15$, $SD = 11.21$) and post-test ($M = 49.55$, $SD = 10.49$) scores of the control group was also found to be statistically significant, $t(32) = 3.053$, $p = 0.005$, Cohen's $d = 0.531$ [95% CI = 1.46, 7.33].

Students' Experiences of Using the AI-Supported Lms

The five themes that identified through the content analysis directly correspond to qualitative research questions RQ3 through RQ7, as shown in Table 5.

Table 5. Themes, subcategories, and frequency values identified through qualitative analysis

Research Questions	Theme	Subcategories
RQ3	System Usage Process	Ease of use, interface experience, most utilized features, difficulties encountered
RQ4	Functioning of the Feedback Mechanism	Clarity of feedback, error - specific explanations, guiding hints, benefit of instant feedback
RQ5	Contributions to the Learning Process and Academic Achievement	Contribution to PHP learning, motivational change, error awareness, learning control
RQ6	System Design and Content Structure	Content adequacy, quality of exercises, perception of personalization
RQ7	Challenges Encountered and Suggestions for Improvement	Technical issues, content deficiencies, improvement suggestions

System Usage Process

Students' usage process experiences with the AI-supported LMS were examined under four subcategories: Ease of use (n=10), interface experience (n=9), most utilized features (n=12), difficulties encountered (n=8).

Of the twelve participants, ten rated the system as easy to use, and two as partially easy. The most frequently highlighted positive features were ease of on-page navigation and accessibility from multiple platforms. One participant described how quickly they could navigate upon first logging in: "When I started using the system, I could immediately see where everything was; I didn't have to search for anything" (P3). Thanks to mobile access, learning took place outside of scheduled class hours. One student said: "I was able to log in from my phone, which was really helpful. I was even able to do some things before class"(P7). Two of the relatively easy to use participants said they needed some time to get used to the order in which the menus were placed: "The menus were a bit confusing at first, but I got used to them after a few logins"(P11). Nine users said the system's design and aesthetic appeared to be well-structured and visually non-distracting. Three respondents indicated a need to modify navigation; and emphasized the role of color palette and button placement for navigation: "The color scheme and button placement were good. I always knew where to look" (P1). P9: "I think the simplicity of the practice screens was good for me to focus; therefore, I could concentrate solely on the code,"which is positive feedback on the practice screen design as an element that facilitated focus. The biggest issue regarding the mobile device was the font: "Sometimes the font looked small, especially on the mobile screen"(P4).

Twelve participants stated that the most important advantage of the system was instant feedback. Other commonly mentioned features were the code editor (n=10) and the progress board (n=7). According to the participants, real-time feedback was defined as the ability to instantly observe the source and cause of errors in the coding process: "As soon as I entered the code, I saw what I was doing and why I was getting an error"(P6). The syntactic feedback loop in the code editor was noted to speed up error handling: "Writing in the editor while receiving syntactic feedback was very helpful because then I could see and understand exactly where I made a mistake" (P2). The progress board represented tracking

efforts on the platform: "Being able to monitor and observe my work from every angle made me want to stay on the platform" (P1).

Eight participants expressed negative opinions about the system. Delay in page loading, inadequate descriptions of the tasks the user would work on, and the lack of a dedicated help section were common problems. Slow page loading affecting the workflow was expressed as a problem: "Sometimes the page took so long to load that I would stop working" (P5). Inadequate task descriptions were the main reason for failure to navigate on initial attempts, supported by the statement: "The task description in the first exercise was useless; I didn't know what to do" (P8). One participant said they were directed to forums because there was no help section: "When I had a problem, there was no help section, so I looked at the forums" (P3).

Feedback Mechanism

Findings related to the feedback mechanism were organized under four subcategories: Clarity of feedback (n=6), error-specific explanations (n=9), guiding hints (n=9), benefit of instant feedback (n=8).

Participants had differing opinions regarding the feedback mechanism. While some participants found the system understandable, others pointed out various shortcomings. Positive feedback was given by participants who thought it was correct for the system to indicate not only that an error had occurred, but also where it occurred: "The error message didn't just say 'wrong,' it also stated what was wrong and why" (P5). On the other hand, some participants noted that technical terms were not explained: "It didn't explain the technical terms; I had to research the meaning of the concept myself" (P10).

Nine participants thought it was useful for the system to distinguish syntactic errors from logical errors. One participant explained what knowing the type of error meant in practice: "I thought I knew the difference between a syntactic error and a logical error, but I didn't fully understand it in practice. The system showed me" (P8). Knowing the type of error also increased speed: "After learning what type of error I made, I also learned how to correct it faster" (P1). This helped prevent students from repeating the same mistake: "I learned what to look out for so as not to make the same mistake again"(P11).

Nine participants said that the hints reinforced their learning instead of the direct answers: "It didn't give me the answer directly; it told me where to look. I learned better when I found the answer myself"(P12). Three were initially disappointed but changed their minds over time: "I was initially disappointed because it didn't give me the answer directly. Later I realized that I actually learned more"(P7). One participant mentioned the difference between being given an answer and finding the answer themselves: "I never forget something when I figure it out myself; but I quickly forget an answer that is given to me"(P4).

Eight participants stated that the biggest advantage of the system over traditional teaching methods was the immediate feedback. Participants compared this to the challenges of providing feedback and correction to students doing homework or exercises in a crowded classroom, commenting: "I make mistakes in class, but because the class is crowded, the feedback isn't always timely. Here, you can see it within seconds"(P6). Participants also noted that immediate feedback eliminated the disadvantages that might arise from delays in noticing and correcting errors. One participant said, "If I had received feedback in class, I would probably have already

forgotten that mistake"(P10). Furthermore, it was emphasized that detecting errors the moment they occur helps in correcting them. Indeed, as one participant stated, "I used to accumulate mistakes; now I see them instantly, and if I didn't see them, it would be much harder to get it right later"(P9), demonstrating that immediate feedback improves the learning process by preventing the accumulation of errors

The Contribution of the AI-Supported LMS to the Learning Process and Academic Achievement

Findings related to the contribution of the AI-supported LMS to the learning process and academic achievement were organized under four subcategories: Contribution to PHP learning (n=7), motivational change (n=10), error awareness (n=9), learning control (n=8).

Seven participants stated that the system taught them PHP and that hands-on exercises played the biggest role. This was expressed as a theme explaining that theoretically difficult concepts can be learned more easily through practice: "I couldn't grasp loops and explain how they work; thanks to the exercises integrated into the system, I managed to learn by doing" (P2). Furthermore, exercises based on real-life scenarios enabled understanding concepts rather than memorization: "Each exercise was based on a real-world scenario. I didn't just memorize, I also learned why I was using it" (P7). One participant who knew nothing about PHP stated that they were able to do the exercises independently: "I had no knowledge of PHP; I did all these exercises on my own." (P4).

Ten participants reported that their motivation increased as they became more familiar with the system, while two participants stated that they were quite motivated when they first used the system and did not notice much change. The panel used to track progress was shown to be the most frequently used factor among participants with higher motivation (P12). The sense of progress associated with completed exercises was identified as the motivation to continue practicing: "I felt a sense of progress every time I completed an exercise. This pushed me to do the next one" (P6). In contrast, one participant described the experience as different from doing homework (P3): "It didn't feel like doing homework; it was like discovering something."

Nine participants said the system made them aware of their own mistakes. They suggested that the visibility of repeated mistakes became a tactic of the system: "The system kept showing me that I was making the same mistake over and over. I started thinking, 'Again?'" (P5). According to the participants, identifying recurring mistake patterns was the first step in correcting them. Specifically, it allowed students to review their own habits and learning processes because it showed them that they were making mistakes repeatedly. P9 expressed this situation as follows: "Forgetting to put a semicolon had become a habit for me; I am aware of it now" (P9). It was also found that associating errors with individual performance increased students' attention levels. One participant even said, "When I saw that the mistake was mine, I started to be more careful" (P12), indicating that error awareness encouraged more diligent study behaviors.

Eight participants reported having more control over their learning processes. The opportunity to progress at an individual pace emerged as a key aspect of this autonomy: "I could practice again whenever I wanted. I didn't have to stay at the same level as everyone else in the class"(P8). The progress tracking tool was described as a mechanism that guided participants to areas requiring more attention: "I could see how much I had learned in each subject. This showed me

what I needed to work on"(P1). It was noted that learning became more relaxed and personalized when participants could work independently. One participant expressed this feeling as follows: "I repeated the exercises as many times as I wanted; nobody asked if I was bored"(P6).

System Design and Content Structure

Students' perspectives relating to the design of the system and content structure were categorized in relation to Content Adequacy (n=8), Quality of Exercises (n=10), and Perception of Personalization (n=8).

Eight participants reported that the content was adequate or very adequate, and four participants indicated that they required extra content on more advanced PHP topics and database integration. For respondents that rated the content as adequate, ease and comprehensibility in the introduction of the basic topics was perceived most of the time: "It was more than enough for beginners; everything was clearly explained" (P7). In comparison, a number of users were asking for more coverage in advanced topics. "I learned PHP at the simplest level, I just needed to see more examples for advanced learning or database integration" (P3). Another participant confirmed this stating, "It was good for learning the basics, but the advanced information was a bit superficial" (P10).

Ten participants found the exercises helpful because they addressed everyday problems, contributing to their observations as follows: "Each exercise was a real problem; we found it not abstract" (P5). The gradual escalation of difficulty was also received positively: "The difficulty level increased gradually; and I didn't feel like it suddenly got harder" (P11). But two respondents suggested that the variety in training types could be improved, stating that "Some exercises are very similar to each other; more variety could be used" (P2).

Eight participants described being partially or fully personalized and had a unique response to personalization. Error-focused feedback emerged as one of the most salient dimensions: "It provided specific feedback showing where I made a mistake; I don't think everyone got the same message" (P9). Users also found the self-paced feature useful: "I was progressing at my own pace; this allowed me to progress independently" (P4). However, four participants indicated their expectations of adaptive content suggestions were not met: "It would be great if the system told me which topics I need to review again" (P12).

Challenges Encountered and Suggestions for Improvement

The results around the challenges faced and improvement ideas were categorized into three categories: technical issues (n=9), content shortcomings (n=6) and suggestions on improvement (n=5).

Nine of the twelve participants experienced at least one technical problem while using the system. The worst issues were page crashes, session errors, and login problems. Page crashes usually led to lost code: "The page crashed intermittently; I lost the code I was writing at that moment" (P5). One participant had to reset the password twice because of login problems: "I had problems on the login page twice and had to reset my password"(P8). Technical problems also lowered motivation: "My desire to work decreased when a technical problem occurred, but I continued after the problem was resolved" (P3).

Six participants included at least one limitation in terms of content or features. Many of the most commonly requested

options were video tutorials and explanation visuals. A participant reported that video support would help them process concepts quickly: "I could grasp some concepts faster if there were video tutorials" (P11). Alternatively, another advised that interactive diagrams, and visual explanations, would be more comprehensive: "Interactive diagrams and visual explanations would be more comprehensive for me" (P2). Five respondents suggested some targeted enhancements to the technical reliability, to enrich the content, and to improve the mobile support. They emphasized that technical reliability is essential for other enhancements: "Fixing technical issues is the most important thing." A good idea is useless with bad execution" (P5). Adding real-life scenarios was mentioned as a way to enrich the content: "It would be good if more real-life scenarios were added" (P7). Mobile compatibility was highlighted as a separate theme requiring attention: "Mobile compatibility needs to be improved; using it on a phone was sometimes difficult" (P4).

Discussion

This study demonstrated a positive effect of the AI-supported LMS developed for a PHP programming course on students' academic achievement. This result supports research showing that students are successful in course performance and classroom participation in AI-supported learning environments (Hu, 2024; Wang & Luo, 2023).

Quantitative results showed an increase in academic achievement in both groups, but the difference was greater in the experimental group. Effect size values showed a large effect ($d = .84$) for the experimental group and a moderate effect ($d = .53$) for the control group. The differences between the groups were statistically significant when considering pre-test scores ($\eta^2 = .081$). It was concluded that the AI-supported platform affected students' academic performance. It should also be noted that, although the experimental group outperformed the control group, post-test achievement remained moderate in absolute terms ($M = 56.18$ on a 100-point scale). This may reflect the cognitive difficulty of PHP programming for novice learners, the demanding nature of the achievement test, and the relatively short, four-week duration of the intervention, suggesting that substantial gains in programming proficiency require sustained practice beyond a brief implementation.

According to feedback theory (Shute, 2008), feedback specific to the student's performance and provided with appropriate guidance to support the learning process increases the ability to succeed. In this context, the most satisfying aspect of the system for the students participating in the study was found to be the provision of immediate feedback. In standard instructional contexts, feedback is often delayed, and this time delay can hinder the learning process (Cavalcanti et al., 2021; Hattie & Timperley, 2007). In the developed system, however, students received feedback immediately after submitting their code; this allowed them to quickly identify and correct their mistakes. Since the system was evaluated as a whole, not individually, in the research design, it cannot be said that immediate feedback alone caused the observed increase in academic achievement in the experimental group. Instead, it is more appropriate to consider immediate feedback as one of the elements contributing to this improvement, along with the other components of the system. More broadly, the present design cannot determine whether the observed advantages stemmed from the AI-generated feedback itself, from broader access to the LMS and its mobile features, or from the combination of these components; isolating their

independent contributions remains an important direction for future research.

One of the notable findings of the study is that the system did not merely identify the error but also explained its cause. Participants regarded the separate explanation of syntactic and logical errors in particular as a feature that supported learning. This finding is consistent with the literature emphasizing that explanatory feedback supports learning (Wisniewski et al., 2019). Especially in programming education, where error-correction and problem-solving processes are intensive, feedback that explains the cause of an error rather than merely reporting its presence is reported to be more effective (Keuning et al., 2018; Hao et al., 2022). In the present study as well, students' finding the detailed explanations useful indicates that the system helped them understand their errors, see the points they had missed, and develop more accurate solution strategies.

Qualitative findings also showed that students became aware of their recurring errors over time. This is thought to be closely related to the concept of metacognition, which means an individual's ability to monitor, evaluate, and regulate their own thinking and learning processes (Flavell, 1979). The development of metacognitive awareness contributes not only to students correcting their errors but also to them questioning the reasons for these errors and avoiding similar mistakes (De Boer et al., 2018; Theobald, 2021). The system's explanatory feedback, it was determined, assisted students' attention about learning in this regard. LMS system served not only as an assessment measure but also as a self-regulating learning mechanism.

A second finding of the study is that rather than offering solutions to students, the system simply offers advice to steer them toward solutions. This strategy is supportive of scaffolding theory, which dictates that students should receive support based on their current level of competence (Wood et al., 1976). For this measure, temporary and guiding support is provided for students, and such support is gradually lowered as the student becomes more proficient. Similarly, the cues in this study also empowered students in problem-solving by making them take a more active role in the learning process.

The qualitative results indicate that students are trying to find solutions for learning by themselves, in a bid for it to become permanent and meaningful. While some students initially viewed the inability to directly access answers negatively, this perception changed over time, and they realized that cue-based guidance supported their studies. This demonstrates that although it is difficult in the short term, effortful guidance enables sustained learning on the long time-lines. This is in agreement with the zone of proximal development theory, which posits that students can learn skills that they would not ever learn alone (Vygotsky, 1978). Furthermore, the system's progress tracking has positive aspects as a tool that students can use to manage their motivation and learning. This outcome is also well supported by self-regulated learning theory highlighting that students actively engage in setting, monitoring and appraising goals (Zimmerman, 2002).

Literature shows that when systems offer ongoing and visible feedback on learning processes, it helps students develop self-regulation skills and academic outcomes (Zimmerman, 2002; Nicol & Macfarlane-Dick, 2006; Heikkinen et al., 2023). In this context, students reporting greater control over their own learning can be considered one of the system characteristics that may be associated with the higher academic achievement of the experimental group.

However, since the current design did not allow for the independent evaluation of the effects of the system components, it is not possible to claim that this effect stems solely from this characteristic. Conversely, the more limited opportunities in traditional teaching environments to regularly monitor students' progress and immediately identify their shortcomings may be one of the possible explanations for the relatively lower achievement of the control group.

On the other hand, qualitative findings reveal differences in students' perceptions of personalized learning. While some users consider the feedback developed by the system around their own mistakes as personalized, others state that they expect the system to suggest which topics to review based on their performance. This data shows that personalization of feedback and adjustment of the student learning path are perceived by students as complementary features (Hooshyar et al., 2024; Zheng et al., 2022). The developed system provides feedback based on individual mistakes in the student's work, but it does not personalize the content flow, approach, or topic suggestions according to the student's performance. This personalizes the system at the feedback level, but does not lead to the conclusion that it is a fully adaptable system when it comes to learning.

At the same time, it should be taken into account that the findings cannot be generalized to all learning contexts. The literature includes studies showing that AI-supported learning environments used in programming education can positively affect student engagement, motivation, and learning performance (Chen et al., 2025; Zhang, Jiang, et al., 2025; Ma et al., 2025). On the other hand, some studies reveal that when AI tools are not appropriately embedded in the learning process, students can become overly dependent on them, participate less in the problem-solving process, and fail to show the expected improvement in learning outcomes (Lo et al., 2024). This shows that the effectiveness of AI-supported systems depends not only on the use of the technology but also on how it is embedded in the learning process and how students interact with these tools. Additionally, the effectiveness of AI support may depend on the student's prior knowledge and AI literacy (Jing et al., 2024; Zhang, Rice, et al., 2025). It is thought that the limited prior knowledge of the participants in this study may have enhanced the effect of the structured support provided by the system. Consequently, the study provides not only outcome-oriented but also process-oriented evidence regarding the effectiveness of AI-assisted learning environments. In this respect, the research makes a unique contribution to the literature by combining quantitative and qualitative findings to explain the learning experiences that accompany improvements in academic achievement.

Limitations

It is considered that some limitations of this study should be taken into account. The study was conducted with a convenience sample of 67 students enrolled in a single vocational school; this limits the generalizability of the findings to broader populations and educational contexts. Participants contributing to the qualitative analysis were selected from among those who voluntarily participated in the system and those who routinely used it. This is thought to create a positive bias, as students with low participation or negative impacts are less likely to be included in the interview responses. In addition, detailed system-usage data (e.g., session counts, code submissions, and the number of feedback instances received) were not systematically logged; consequently, students' engagement with the AI-supported

LMS could not be quantified as an objective index of implementation fidelity, and the identification of active users relied on instructor observation rather than log data. Future research should incorporate automatic interaction logging to document usage intensity and to support the selection of information-rich cases. Since the implementation of this study was carried out over a four-week period, the extent of the long-term impact of the AI-assisted system on student achievement and motivation is not clearly understood. In addition, because the experimental and control conditions were assigned at the class level while the statistical analyses were conducted at the student level, a potential clustering effect should be acknowledged: students nested within the same class may not be fully independent, which can influence the estimation of standard errors. Future studies involving a larger number of classes could address this through hierarchical modeling. The factors of the experimental and control groups differ not only in terms of AI feedback but also in terms of the total LMS, including online and mobile device availability, monitoring status, dashboard, etc.; therefore, it is considered that it is not possible for this study to definitively determine how much of the difference is due solely to AI. These questions may be answered in the future with a third group that does not use AI feedback in the same LMS. The accuracy of AI feedback has not been verified. Large language models can sometimes detect false errors or provide misleading advice, and this study is not designed to detect this. For future research, it is thought that this need could allow for studies to be conducted to check whether the feedback for the system is accurate and useful. From an ethical standpoint, a further limitation is that directly identifying student information was transmitted to a third-party service without explicit consent for such processing; future implementations should pseudonymize or otherwise minimize personal data before transmission and obtain explicit informed consent for third-party processing.

Conclusion and Recommendations

This study investigated the impact of an AI-supported LMS developed for PHP programming education on student success and learning experience. Quantitative results for both groups showed a statistically significant increase in academic performance following the implementation. However, when considering effect sizes, the experimental group showed a large level of improvement, while the control group's improvement remained at a moderate level. This difference indicates that the AI-supported system creates additional learning that traditional learning alone cannot provide. The mechanism through which this contribution is achieved is supported by qualitative findings. Instant feedback was identified as the most practical aspect of the system for students. Feedback that both identified and explained the source and type of errors helped students perceive their own recurring error patterns. The hint-based method, which guided students towards solutions rather than directly offering solutions, promoted cognitive autonomy and learning. Progress tracking emerged as a feature that strengthened students' self-regulation skills. At the same time, technical problems, lack of content in advanced topics, and unmet expectations regarding adaptive content suggestions were identified as the current limitations of the system. In interpreting these results, it is important to distinguish what the study does and does not demonstrate: it shows that the AI-supported LMS, as an integrated system, was associated with higher achievement and positively experienced feedback, but it does not establish that AI-generated feedback alone

produced these gains, nor does it address their long-term durability.

When the quantitative and qualitative findings are considered together, the achievement gains are seen to align consistently with students' reported experiences. Instant feedback, error awareness, and the sense of control over the learning process provide qualitative support for the improvements observed at the quantitative level. The absence of these mechanisms in traditional instruction, in turn, contextually accounts for the control group's more limited gains. Meanwhile, the unmet expectations regarding adaptive content recommendations draw attention to a component of the system that remains open to development, suggesting that improvements in this area could contribute positively to the student experience.

Based on these findings, several recommendations are put forward. Researchers and designers planning to develop AI-supported systems in similar educational contexts are advised to place feedback quality and student progress tracking at the center of their design, in addition to content delivery. Higher education institutions are recommended to address technical reliability and mobile accessibility as fundamental infrastructure requirements, and to prioritize the enrichment of advanced subject matter content and the addition of adaptive content suggestion features, which the current system does not yet provide. Student expectations indicate that error-focused feedback, subject suggestions, individualized learning paths, and performance-based content guidance can improve the learning experience in future systems. It is believed that integrating personalized feedback and adaptive learning features can increase the effectiveness of AI-supported learning environments. Therefore, addressing these two features together can be considered a significant area of development and research in future learning systems and ongoing research in this field.

Studies examining the experiences of both groups will qualitatively address the lack of data on student experiences in traditional teaching settings and allow for a more comprehensive comparison of the findings. Experimental designs aimed at measuring the specific contribution of adaptive content suggestions to student achievement independently of other system components will answer the most critical question left unanswered in this study. Finally, replicating this study in different programming languages and teaching levels will contribute to testing the generalizability of the findings.

Author Contributions

The author contributed to all stages of the study, including conceptualization, methodology, data collection, analysis, interpretation, and manuscript preparation.

Ethical Declaration

This study was conducted with the approval decision numbered 09, dated 29.02.2024, by the Scientific Research and Publication Ethics Board of İskenderun Technical University (16.02.2026-03 E-22398675-050.04-222765).

Conflict of Interest

The authors declare that there is no conflict of interest with any institution or person within the scope of this study.

Declaration of Generative AI Use

The author confirms that no generative artificial intelligence (AI) tools were used in the drafting, writing, or substantive preparation of this manuscript.

References

- Cavalcanti, A. P., Barbosa, A., Carvalho, R., Freitas, F., Tsai, Y.-S., Gašević, D., & Mello, R. F. (2021). Automatic feedback in online learning environments: A systematic literature review. *Computers and Education: Artificial Intelligence*, 2, 100027. <https://doi.org/10.1016/j.caeai.2021.100027>
- Chen, A., Wei, Y., Le, H., & Zhang, Y. (2025). Learning by teaching with Chat GPT: The effect of teachable Chat GPT agent on programming education. *British Journal of Educational Technology*, 57, 163–184. <https://doi.org/10.1111/bjet.70001>
- Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). Lawrence Erlbaum Associates. <https://doi.org/10.4324/9780203771587>
- Creswell, J. W., & Plano Clark, V. L. (2022). *Designing and conducting mixed methods research* (4th ed.). SAGE Publications.
- Creswell, J. W., & Poth, C. N. (2018). *Qualitative inquiry and research design: Choosing among five approaches* (4th ed.). SAGE Publications.
- De Boer, H., Donker, A. S., Kostons, D. D., & Van der Werf, G. P. C. (2018). Long-term effects of metacognitive strategy instruction on student academic performance: A meta-analysis. *Educational Research Review*, 24, 98–115. <https://doi.org/10.1016/j.edurev.2018.03.002>
- Ezzaim, A., Dahbi, A., Haidine, A., & Aqqal, A. (2023). AI-based adaptive learning: A systematic mapping of the literature. *Journal of Universal Computer Science*, 29(10), 1283–1312. <https://doi.org/10.3897/jucs.90528>
- Flavell, J. H. (1979). Metacognition and cognitive monitoring: A new area of cognitive-developmental inquiry. *American Psychologist*, 34(10), 906–911. <https://doi.org/10.1037/0003-066X.34.10.906>
- Garzón, J., Patiño, E., & Marulanda, C. (2025). Systematic review of artificial intelligence in education: Trends, benefits, and challenges. *Multimodal Technologies and Interaction*, 9(8), 84. <https://doi.org/10.3390/mti9080084>
- Hao, Q., Smith IV, D. H., Ding, L., Ko, A., Ottaway, C., Wilson, J., Arakawa, K. H., Turcan, A., Poehlman, T., & Greer, T. (2022). Towards understanding the effective design of automated formative feedback for programming assignments. *Computer Science Education*, 32(1), 105–127. <https://doi.org/10.1080/08993408.2020.1860408>
- Hattie, J., & Timperley, H. (2007). The power of feedback. *Review of Educational Research*, 77(1), 81–112. <https://doi.org/10.3102/003465430298487>
- Heikkinen, S., Saqr, M., Malmberg, J., & Tedre, M. (2023). Supporting self-regulated learning with learning analytics interventions: A systematic literature review. *Education and Information Technologies*, 28(3), 3059–3088. <https://doi.org/10.1007/s10639-022-11281-4>
- Hooshyar, D., Weng, X., Sillat, P. J., Tammets, K., Wang, M., & Hämäläinen, R. (2024). The effectiveness of personalized technology-enhanced learning in higher education: A meta-analysis with association rule mining. *Computers & Education*, 223, 105169. <https://doi.org/10.1016/j.compedu.2024.105169>
- Hsieh, H.-F., & Shannon, S. E. (2005). Three approaches to qualitative content analysis. *Qualitative Health Research*, 15(9), 1277–1288. <https://doi.org/10.1177/1049732305276687>

- Hu, S. (2024). The effect of artificial intelligence-assisted personalized learning on student learning outcomes: A meta-analysis based on 31 empirical research papers. *Science Insights Education Frontiers*, 24(1), 3873–3894. <https://doi.org/10.15354/sief.24.re395>
- Jing, Y., Wang, H., Chen, X., & Wang, C. (2024). What factors will affect the effectiveness of using Chat GPT to solve programming problems? *Humanities and Social Sciences Communications*, 11(1), Article 319. <https://doi.org/10.1057/s41599-024-02736-9>
- Keuning, H., Jeuring, J., & Heeren, B. (2018). A systematic literature review of automated feedback generation for programming exercises. *ACM Transactions on Computing Education*, 19(1), 1–43. <https://doi.org/10.1145/3231711>
- Kölemen, C. Ş., & Şahin, E. (2026). Yapay zekâ destekli anında geri bildirimin lise öğrencilerinin başarı motivasyonuna etkisi [The effect of AI-supported instant feedback on high school students' achievement motivation]. *Eğitim Teknolojisi Kuram ve Uygulama*, 16(1), 131–158.
- Lo, C. K., Hew, K. F., & Jong, M. S. Y. (2024). The influence of Chat GPT on student engagement: A systematic review and future research agenda. *Computers & Education*, 219, Article 105100. <https://doi.org/10.1016/j.compedu.2024.105100>
- Ma, W., Ma, W., Hu, Y., & Bi, X. (2025). The who, why, and how of AI-based chatbots for learning and teaching in higher education: A systematic review. *Education and Information Technologies*, 30(6), 7781–7805. <https://doi.org/10.1007/s10639-024-13128-6>
- Merriam, S. B. (2002). *Qualitative research in practice: Examples for discussion and analysis*. Jossey-Bass.
- Miles, M. B., & Huberman, A. M. (1994). *Qualitative data analysis: An expanded sourcebook* (2nd ed.). Sage Publications.
- Nicol, D. J., & Macfarlane-Dick, D. (2006). Formative assessment and self-regulated learning: A model and seven principles of good feedback practice. *Studies in Higher Education*, 31(2), 199–218. <https://doi.org/10.1080/03075070600572090>
- Nunnally, J. C., & Bernstein, I. H. (1994). *Psychometric theory* (3rd ed.). Mc Graw-Hill.
- Sezer, C., & Korucu, A. T. (2019). Bilişim teknolojileri ve yazılım dersinde öğrenme yönetim sistemi kullanımının öğrenci akademik başarısına etkisi [The effect of using a learning management system in the information technologies and software course on students' academic achievement]. *Gazi Eğitim Bilimleri Dergisi*, 5, 157–176.
- Shute, V. J. (2008). Focus on formative feedback. *Review of Educational Research*, 78(1), 153–189. <https://doi.org/10.3102/0034654307313795>
- Tabachnick, B. G., & Fidell, L. S. (2019). *Using multivariate statistics* (7th ed.). Pearson.
- Theobald, M. (2021). Self-regulated learning training programs enhance university students' academic performance, self-regulated learning strategies, and motivation: A meta-analysis. *Contemporary Educational Psychology*, 66, 101976. <https://doi.org/10.1016/j.cedpsych.2021.101976>
- Tlili, A., Shehata, B., Adarkwah, M. A., Bozkurt, A., Hickey, D. T., Huang, R., & Agyemang, B. (2023). What if the devil is my guardian angel: Chat GPT as a case study of using chatbots in education. *Smart Learning Environments*, 10(1), 15. <https://doi.org/10.1186/s40561-023-00237-x>
- Topham, L., Atherton, P., Reynolds, T., Hussain, Y., Hussain, A., Kolivand, H., & Khan, W. (2025). Artificial intelligence in educational technology: A systematic review of datasets and applications. *ACM Computing Surveys*. Advance online publication. <https://doi.org/10.1145/3768312>
- Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological processes*. Harvard University Press.
- Wang, L., & Luo, J. (2023). The relationship between student interaction with generative artificial intelligence and learning achievement: Serial mediating roles of self-efficacy and cognitive engagement. *Frontiers in Psychology*, 14, Article 1285392. <https://doi.org/10.3389/fpsyg.2023.1285392>
- Wisniewski, B., Zierer, K., & Hattie, J. (2019). The power of feedback revisited: A meta-analysis of educational feedback research. *Frontiers in Psychology*, 10, 3087. <https://doi.org/10.3389/fpsyg.2019.03087>
- Wood, D., Bruner, J. S., & Ross, G. (1976). The role of tutoring in problem solving. *Journal of Child Psychology and Psychiatry*, 17(2), 89–100. <https://doi.org/10.1111/j.1469-7610.1976.tb00381.x>
- Xu, W., & Ouyang, F. (2021). A systematic review of AI role in the educational system based on a proposed conceptual framework. *Education and Information Technologies*, 27, 9455–9478. <https://doi.org/10.1007/s10639-021-10774-y>
- Yıldırım, A., & Şimşek, H. (2021). *Sosyal bilimlerde nitel araştırma yöntemleri [Qualitative research methods in the social sciences]* (12. baskı). Seçkin Yayıncılık.
- Zhang, C. X., Rice, R. E., & Wang, L. H. (2025). College students' literacy, Chat GPT activities, educational outcomes, and trust from a digital divide perspective. *New Media & Society*. <https://doi.org/10.1177/14614448241301741>
- Zhang, L., Jiang, Q., Xiong, W., & Zhao, W. (2025). Effects of Chat GPT-based human–computer dialogic interaction programming activities on student engagement. *Journal of Educational Computing Research*, 63(4), 988–1023. <https://doi.org/10.1177/07356331251333874>
- Zheng, L., Long, M., Zhong, L., & Gyasi, J. F. (2022). The effectiveness of technology-facilitated personalized learning on learning achievements and learning perceptions: A meta-analysis. *Education and Information Technologies*, 27(8), 11807–11830. <https://doi.org/10.1007/s10639-022-11092-7>
- Zimmerman, B. J. (2002). Becoming a self-regulated learner: An overview. *Theory Into Practice*, 41(2), 64–70. https://doi.org/10.1207/S15430421TIP4102_2