Issues in Information Systems
Volume 27, Issue 1, pp. 401-418, 2026

DOI: https://doi.org/10.48009/2_iis_2026_128

Development and Evaluation of an AI-Powered Adaptive Learning Platform for Introductory Programming Education

Reema Bhaskar, *Middle Georgia State University, reema.bhaskar@mga.edu*
Doyeon Lee, *Middle Georgia State University, doyeon.lee@mga.edu*
Joobum Kim, *Middle Georgia State University, joobum.kim@mga.edu*
Myungjae Kwak, *Middle Georgia State University, myungjae.kwak@mga.edu*
Jongho Seol, *Middle Georgia State University, jongho.seol@mga.edu*
Jonathan Jenkins, *Middle Georgia State University, jonathan.jenkins2@mga.edu*

Abstract

Introductory computer science courses face persistently high failure rates, with traditional pedagogical approaches unable to accommodate the diverse learning needs of undergraduate students. While intelligent tutoring systems have been developed to address this challenge, most lack empirical evidence of effectiveness and remain constrained to specific institutional contexts. This paper presents the design, implementation, and preliminary empirical evaluation of an AI-powered adaptive learning platform built with Django and the OpenAI API for introductory Python programming education. The platform integrates a 14-chapter mastery-based curriculum with a 5-gate progression system requiring 80% proficiency thresholds, supported by a comprehensive suite of large language model-driven tools including a debug lab, code review engine, and context-aware study assistant. An empirical pilot study was conducted using this platform with CS I students at a state university in the southeastern United States. Although the small sample size limited the statistical power of the formal hypothesis tests, the observed effect sizes were medium-to-large for quiz scores (d = 0.63) and large for open-ended Q&A items (d = 1.02), and exploratory structural equation modeling of survey data (N = 11) revealed theoretically coherent patterns of association among perceived usefulness, self-efficacy, engagement, learning effectiveness, and satisfaction. These findings provide preliminary, non-causal evidence that the platform is a promising approach warranting larger-scale evaluation, while the platform's free, open-access design offers a model for democratizing AI-augmented instruction.

Keywords: adaptive learning, generative AI, computer science education, intelligent tutoring systems, large language models, programming education

Introduction

Introductory computer science courses, commonly designated CS I, serve as the gateway to the computing discipline for hundreds of thousands of undergraduate students each year. Despite their foundational importance, these courses are characterized by high failure and withdrawal rates. A meta-analysis by Watson and Li (2014) found average failure rates of approximately 33% across CS I offerings internationally, a figure that has remained stubbornly consistent over decades of pedagogical reform. Contributing factors include insufficient prior programming experience, inadequate feedback cycles, the abstract nature of programming concepts and syntax, and the inability of large lecture formats to accommodate diverse learning needs (Gomes & Mendes, 2007; Luxton-Reilly et al., 2018).

Traditional CS I pedagogy relies primarily on lectures, textbook readings, and static programming assignments graded on correctness. While these methods establish curricular structure, they offer limited capacity for personalization: a student who struggles with loop logic receives the same instructional sequence as one who finds it intuitive. The result underserves both ends of the ability spectrum, as struggling students fall behind without targeted support while advanced students disengage from material they have already mastered (Kasneci et al., 2024). These pedagogical challenges are compounded by systemic access barriers, as the cost of supplementary resources creates an uneven playing field in which students with greater financial resources receive personalized support their peers cannot access.

To address these challenges, intelligent tutoring systems have been developed to teach computer programming, providing automated feedback, reference materials, and adaptive task sequencing. However, most of these systems lack rigorous research on their effectiveness and remain constrained to specific institutional contexts (Kabudi et al., 2021). Recent advances in generative artificial intelligence, particularly large language models (LLMs) such as OpenAI's GPT series, have created new possibilities for adaptive learning. LLMs can generate explanations, provide feedback on code, answer conceptual questions, and adapt their responses based on context, capabilities that align closely with the functions of a human tutor (Kasneci et al., 2024; Lim et al., 2023).

This paper presents the design, implementation, and preliminary empirical evaluation of an AI-powered adaptive learning platform for introductory Python programming education. The platform was built with Django and the OpenAI API and integrates a 14-chapter mastery-based CS I curriculum with a comprehensive suite of LLM-driven educational tools. Following a prior conference paper that proposed the architectural framework for a generative AI-based adaptive learning system (Kwak et al., 2023), this work advances the research program by reporting the development of a fully functional platform and the results of a within-subjects crossover pilot study conducted with CS I students at a state university in the southeastern United States during the Spring 2026 semester.

Literature Review

Adaptive Learning Systems

Adaptive learning systems dynamically adjust educational content, pacing, and assessment based on individual learner performance and behavior. Unlike static curricula, adaptive platforms construct personalized learning pathways that respond to each student's demonstrated strengths and weaknesses (Bommasani et al., 2021). Studies on adaptive learning focus on creating platforms that account for learners' needs, motivations, and favored learning styles, underscoring the concept of adaptive learning as an alternative to the traditional one-size-fits-all approach to instructional design (Ennouamani & Mahani, 2017). Jonassen et al. (2008) define adaptation as the system's ability to modify its behavior based on the learners' knowledge levels and other characteristics. Li et al. (2021) identified three core procedures for implementing adaptive learning: adapting material choice, providing adaptive navigation assistance, and optimizing content delivery. Essential technologies for personalized learning include data mining, learning assessment, artificial intelligence, knowledge mapping, cognitive advising, guided studying, edge computing, and virtual reality (Li et al., 2021).

A systematic review by Garcia-Martinez et al. (2023), examining 25 studies on adaptive learning technologies, concluded that these systems consistently enhance student engagement and academic performance across diverse educational contexts. A subsequent systematic review of Scopus-indexed articles documented a rapid shift from early rule-based adaptive systems to sophisticated models incorporating machine learning, natural language processing, and LLMs, with demonstrated improvements in engagement, motivation, and performance (Alhajri et al., 2025). However, the literature also identifies persistent challenges, including the risk of over-reliance on AI, concerns about algorithmic bias, and the difficulty of ensuring that AI-generated feedback is pedagogically sound rather than merely technically correct (Kasneci et al., 2024).

Intelligent Programming Language Learning Systems

Numerous researchers have designed intelligent learning systems for CS and programming language education. Müller et al. (2018) demonstrated the viability of IBM Watson as a cognitive software for creating a virtual teacher to answer common questions from students within an introductory Java programming class, and an evaluation conducted through a qualitative study with chosen members of the course confirmed the prototype's viability for its intended purpose. Marouf et al. (2018) designed an intelligent tutoring system called Computer Science Course Tutor, which dynamically adapts to the progress of CS students. While these systems represent meaningful advances, recent surveys of AI-assisted learning systems demonstrate system benefits in terms of adaptability and technical measures but remain limited in practical application in real learning scenarios and in evidence of student learning benefits (Kabudi et al., 2021). This gap between system capability and demonstrated educational impact motivates the present work.

Generative Artificial Intelligence

Generative AI is a branch of artificial intelligence that focuses on creating new content or information. It involves training models to generate original and realistic outputs such as images, music, text, or even entire video sequences. These models use complex algorithms and large datasets to learn patterns and generate digital content that resembles human-created data (Dwivedi et al., 2023). Generative AI can be used for learning by creating interactive and engaging educational materials, generating customized learning content tailored to individual students' needs and preferences, and creating virtual tutors or digital assistants that provide personalized guidance and support (Lim et al., 2023). The deployment of LLMs in CS education is a rapidly evolving research area: LLMs can generate code, explain programming concepts, identify errors, and provide natural language feedback, closely paralleling the functions of a human tutor (Kasneci et al., 2024). Several recent studies specifically examine generative AI tools such as GitHub Copilot and ChatGPT in introductory programming courses, identifying both opportunities for enhanced learning and pedagogical concerns about over-reliance, academic integrity, and the impact on novice programmers' conceptual development (Becker et al., 2023; Denny et al., 2023; Prather et al., 2023). However, studies have documented the risk of students developing dependency on AI-generated solutions, bypassing the productive struggle essential to learning (Fan et al., 2025), and LLMs are susceptible to hallucination, the generation of plausible but incorrect information, which poses particular risks in instructional settings where students may lack the expertise to identify errors (Isaeva et al., 2025).

Pedagogical Frameworks

The platform described in this paper draws on three established pedagogical frameworks. Bloom's (1968) mastery learning model posits that most students can achieve high levels of competence given sufficient time, appropriate instruction, and formative feedback, and that advancement should be contingent on demonstrated mastery rather than time spent. Vygotsky's (1978) zone of proximal development (ZPD) describes the space between what a learner can accomplish independently and what they can achieve with guidance. Sweller's (1988) cognitive load theory distinguishes between intrinsic, extraneous, and germane cognitive load and has been identified as a key framework in adaptive learning research (Kang et al., 2025). These frameworks collectively inform the platform's progression system, AI-driven scaffolding, and interface design.

Technology Acceptance and Learning Outcomes

The Technology Acceptance Model (TAM), introduced by Davis (1989), provides a theoretical framework for evaluating user adoption of educational technologies. TAM posits that perceived ease of use and perceived usefulness are primary determinants of technology acceptance. In the context of AI-assisted learning, research has extended TAM to include constructs such as self-efficacy (Compeau & Higgins, 1995), student engagement (Fredricks et al., 2004), learning effectiveness (Al-Rahmi et al., 2015), and

satisfaction (Oliver, 1980). These constructs form the basis of the survey instrument used in the present study.

Research Questions

Building on the literature reviewed above and the architectural framework proposed by Kwak et al. (2023), this study addresses the following research questions:

RQ1: Does an AI-assisted adaptive learning platform produce greater learning gains than traditional instruction in an introductory programming course, as measured by quiz scores and open-ended Q&A assessments?

RQ2: How do students perceive the AI-assisted adaptive learning system in terms of ease of use, usefulness, self-efficacy, engagement, learning effectiveness, and satisfaction?

RQ3: What structural relationships exist among perceived usefulness, perceived ease of use, self-efficacy, engagement, learning effectiveness, and satisfaction in the context of an AI-assisted adaptive learning system?

System Architecture and Design

Architectural Framework

The platform's architecture extends the generative AI-based adaptive learning system framework proposed by Kwak et al. (2023), which consists of three modules: a Learning Module that continuously collects and analyzes various data about learners including their learning styles, preferences, interactions, and history to provide them with a personalized adaptive learning experience; a Domain Knowledge Module that organizes domain knowledge such as learning materials, topics, and fields into an abstract form; and an Adaptation Module powered by generative AI that continuously combines data from the Learning Module and Domain Knowledge Module to create adaptive learning materials tailored to the individual learner's needs and pace. The prototype system described by Kwak et al. (2023) demonstrated the feasibility of using LLMs for both task generation and response evaluation in programming education, utilizing OpenAI's GPT-3.5-turbo model. Building on this foundation, one of the authors designed and developed the full-scale production platform presented in this paper, extending the prototype with a mastery-based progression system, a comprehensive AI tool suite, and student and instructor dashboards. Figure 1 presents the overall architecture of the platform, showing how the three core modules are integrated with the AI-powered tool suite, the 5-gate mastery progression system, and the supporting data and external services.

Technical Stack

The platform is built on the Django web framework (Python), selected for its robust authentication system, object-relational mapping, and scalability. The platform integrates the OpenAI API to power its suite of AI-driven educational tools. User authentication, progress tracking, and curriculum state management are handled through Django's built-in models and session management. The platform includes both a student dashboard and an instructor dashboard. The student dashboard provides learners with a centralized view of their progress, gate completion status, and access to the full AI tool suite, with direct and immediate feedback on submitted work that identifies strengths while targeting specific weaknesses. The instructor dashboard provides professors with data visualizations of student performance, gate completion rates, tool usage patterns, time on task, and assessment score distributions, enabling proactive intervention for struggling students. The platform is hosted as part of the university's Center for AI infrastructure and was additionally incorporated into the School of Computing's web application, expanding its visibility to the broader student population. The platform is publicly accessible at https://learning.mgacenterforai.org/.

Curriculum Design and 5-Gate Mastery Progression

The platform's core curriculum comprises 14 chapters covering the fundamentals of Python programming and introductory CS concepts, aligned with a free, open-access textbook authored by School of Computing faculty and published through the Affordable Learning Georgia (ALG) initiative. The full set of chapters and learning modules is available at https://comp.mga.edu/learning/python/. This alignment ensures curricular consistency with classroom instruction while eliminating textbook costs. The curriculum progresses from basic data types, variables, and input/output operations through control flow, functions, data structures, file handling, and introductory object-oriented programming.

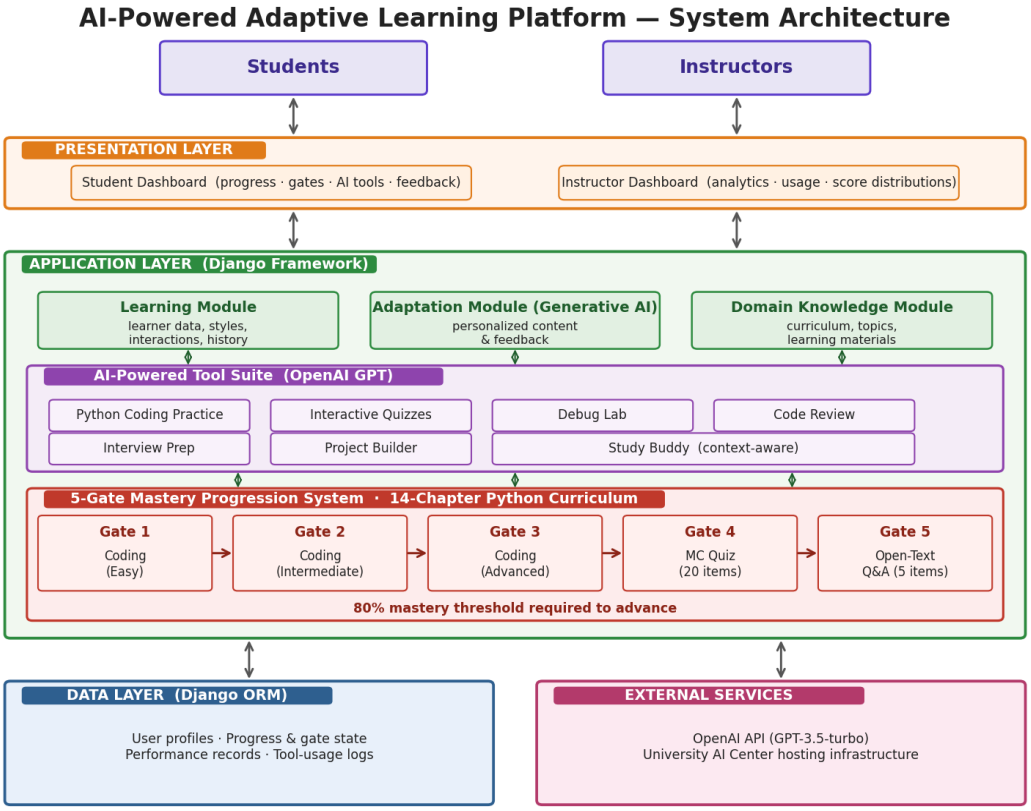


Figure 1. Overall architecture of the AI-powered adaptive learning platform

Central to the platform's pedagogical approach is a 5-gate progression system enforcing mastery-based advancement. Each chapter's gate structure consists of three components. First, students complete three coding exercises at escalating difficulty levels: easy, intermediate, and advanced. The platform applies a tiered scoring system: students who score 80% or above advance, students scoring between 60% and 80% remain at their current level for additional practice, and students scoring below 60% revisit prerequisite material. This mechanism prevents students from accumulating unresolved gaps in understanding. Second, each chapter includes a 20-question multiple-choice quiz assessing conceptual understanding. Third, students complete a 5-question open-text response assessment requiring them to articulate their understanding in their own words, demonstrating not just recognition of correct answers but the ability to reason about the material. Students must score 80% or higher on all components to advance. This structure operationalizes Bloom's (1968) mastery learning model and provides comprehensive evaluation addressing procedural skill, conceptual knowledge, and explanatory ability. Figure 2 illustrates the three components of each chapter's gate structure as they appear to students in the platform: the Practice Exercise (coding), the Topic Quiz (multiple-choice), and the Topic Q&A Session (open-text).

AI-Powered Tool Suite

Beyond the structured curriculum, the platform provides a comprehensive suite of AI-driven tools supporting different dimensions of the CS I learning experience. The Python Coding Practice module presents progressively challenging exercises with constructive, encouraging AI feedback that identifies strengths before addressing errors. Interactive Quizzes provide AI-generated explanations for incorrect answers targeting the specific misconception rather than restating the correct answer. The Debug Lab presents students with intentionally buggy code and guides them through identifying, diagnosing, and correcting errors, grounded in the principle that debugging is one of the most transferable and underemphasized skills in CS I education (Luxton-Reilly et al., 2018). The Code Review tool provides feedback on style, efficiency, and best practices. The Interview Preparation module offers practice with technical interview questions in a simulated format. The Project Builder guides students through designing and implementing small-scale projects with AI-assisted scaffolding. Finally, the Study Buddy is a context-aware conversational assistant that maintains awareness of the student's current position in the curriculum, recent performance, and previously asked questions, enabling it to reference covered material and tailor explanations to the student's demonstrated level of understanding.

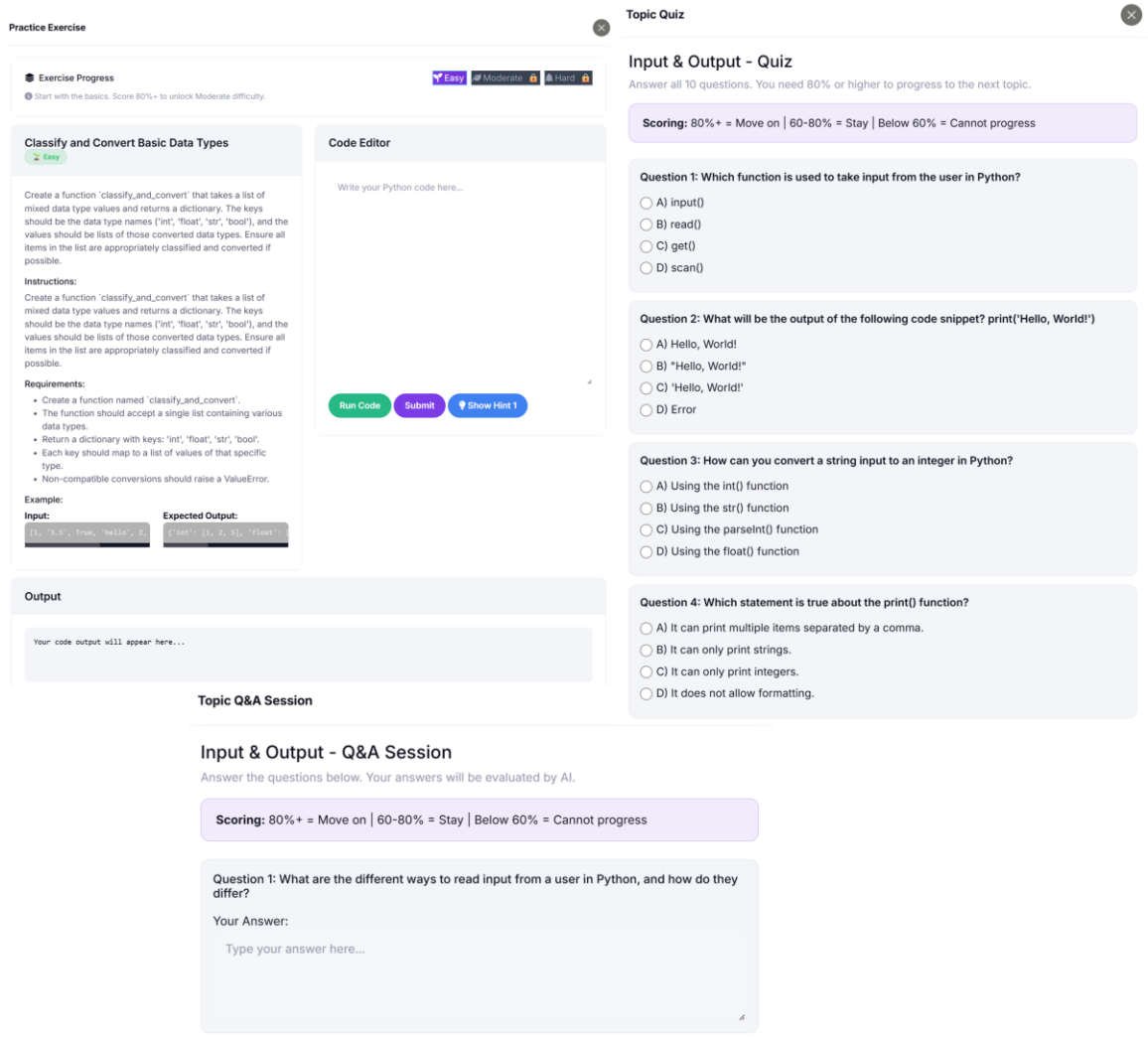


Figure 2. Three components of each chapter's gate structure: Practice Exercise, Topic Quiz, and Topic Q&A Session

Design Rationale and Safeguards

An unconstrained LLM asked to help with a programming problem will typically provide a complete solution, satisfying the student's immediate need while undermining the learning process. The platform addresses this through structured prompts that instruct the model to provide hints, ask guiding questions, and explain concepts rather than generate complete answers, aligning with the Socratic method and with Vygotsky's (1978) principle of scaffolding within the zone of proximal development. Hallucination remains a persistent concern; in an instructional context, a confidently stated but incorrect explanation can be more harmful than no explanation at all for novice learners who lack the domain knowledge to identify errors. The platform mitigates this by constraining model responses to the scope of the current chapter's content and implementing verification checks for code-related outputs. The platform's interface was designed according to Nielsen's (1994) usability heuristics, including clear progress indicators at each gate, user control allowing students to revisit completed material freely, error prevention through confirmation prompts, and recognition over recall through descriptively labeled tool functions. Sweller's (1988) cognitive load theory guided the progressive disclosure of information within each chapter, ensuring students encounter new concepts incrementally.

Methodology

Participants and Setting

The pilot study was conducted to evaluate the preliminary efficacy and pedagogical potential of the AI-assisted adaptive learning system. The participants consisted of 20 undergraduate students majoring in Computer Science at a state university in the southeastern United States, all of whom were enrolled in a "Computer Science I" course focusing on Python programming. The experiment took place during Week 10 (covering Dictionaries & Sets) and Week 12 (covering Classes & Objects) of the Spring 2026 semester, with an intervening spring break in Week 11. Of the 20 enrolled students, 13 provided validated datasets across the academic performance assessments and 11 completed the post-experimental survey; subsequent statistical analyses were conducted on these validated subsets, with sample sizes varying by analysis type as detailed in the Results section.

Experimental Procedure and Design

To ensure internal validity and mitigate potential order effects, a within-subjects crossover design was employed. Students were randomly assigned to Group A or Group B, with their roles as experimental or control subjects reversing between the two instructional sessions. Specifically, in Week 10, Group A utilized the AI-assisted platform while Group B followed traditional classroom procedures, a configuration that was reversed in Week 12. Each experimental session followed a rigorous four-step instructional cycle: pre-test, lecture, laboratory session, and post-test. The two conditions were held equivalent on every dimension except the tool used during the laboratory session: both groups attended the same lecture delivered by the same instructor and completed pre- and post-tests composed of identical items within the same session duration. The sole procedural difference was that AI-assisted students used the platform in place of the traditional paper-based lab worksheet during the laboratory portion of the cycle, while control students completed the equivalent exercises without AI assistance.

Beginning with a pre-test to assess baseline knowledge, the experimental group used the AI-assisted system as a continuous autonomous learning environment. Students utilized the AI system as a personalized scaffold throughout the lecture, self-study phases, and hands-on laboratory sessions. The cycle concluded with a post-test to quantify learning gains.

Measurement Instruments and Reliability

The AI system provided Quiz and Q&A Sessions to measure academic performance. Multiple-choice quiz items were scored automatically against a fixed answer key. Open-ended Q&A responses were scored independently by two human raters using a rubric evaluating the correctness and completeness of each

student's reasoning; raters were blind to condition assignment. Inter-rater agreement was approximately 90%, and any discrepancies were resolved through discussion to reach a final score. Following the completion of both modules, a comprehensive survey was administered via Google Forms to assess the psychological constructs underlying the learning experience. The survey instrument was adapted from the Technology Acceptance Model (TAM) and validated measures of self-efficacy and student engagement.

Table 1 summarizes the measurement sources and reliability coefficients for the six constructs measured in this study: Perceived Ease of Use, Perceived Usefulness, Self-Efficacy, Engagement, Learning Effectiveness, and Satisfaction. All constructs demonstrated high internal consistency, with Cronbach's Alpha values all exceeding 0.80, ranging from 0.83 to 0.96, confirming that the measurement instrument is highly robust and dependable for this study.

Table 1. Measurement Instruments and Reliability

Variable	Measurement	Cronbach's α	Reference
Perceived Ease of Use	Learning to operate this AI-Assisted Learning Application is easy for me. My interaction with this AI-Assisted Learning Application is clear and understandable. It is easy to get this AI-Assisted Learning Application to do what I want it to do.	0.86	Davis (1989)
Perceived Usefulness	Using this AI-Assisted Learning Application improves my academic performance in this course. This AI-Assisted Learning Application makes it easier to study the course materials. I find this AI-Assisted Learning Application useful for my learning and exam preparation.	0.95	Davis (1989)
Self-Efficacy	I believe I can master the skills being taught in this class with the help of this AI-Assisted Learning Application. I feel confident that I can understand the material presented by the instructor using this AI-Assisted Learning Application.	0.93	Compeau & Higgins (1995)
Engagement	I feel focused and deeply involved when using this AI-Assisted Learning Application for my studies. Using this AI-Assisted Learning Application makes me want to spend more time learning the course content. I actively interact with the AI-Assisted Learning Application to explore topics beyond the basic requirements.	0.94	Fredricks et al. (2004)
Learning Effectiveness	This AI-Assisted Learning Application helped me achieve the learning objectives of this course. I have a better understanding of the key concepts thanks to this AI-Assisted Learning Application. This AI-Assisted Learning Application provided high-quality support that enhanced my overall learning outcome.	0.95	Al-Rahmi et al. (2015)
Satisfaction	Overall, I am very satisfied with my experience using this AI-Assisted Learning Application. I am pleased with the decision to use this AI-Assisted Learning Application for this course. I would recommend this AI-Assisted Learning Application to other students taking this course.	0.95	Oliver (1980)

Results

Academic Performance (Quiz)

To ensure the integrity of the pilot test, the analysis was conducted using validated datasets from 13 unique students who successfully completed the pre- and post-tests.

Due to the crossover nature of the study, the available data were utilized differently for each statistical test. For the two-way repeated measures ANOVA, the analysis was restricted to the 7 students who completed the full sequence of both instructional conditions. This approach allowed for a rigorous within-subjects comparison while maximizing the use of all available pilot data for trend analysis. For the

independent t-test, all valid pre-post pairs were included, resulting in 11 observations for the AI-assisted condition and 9 observations for the control condition.

To evaluate the impact of the AI-assisted adaptive learning system on academic achievement, a two-way repeated measures ANOVA was conducted with Time (Pre-test vs. Post-test) as a within-subjects factor and Condition (AI-assisted vs. Control) as a between-subjects factor. The analysis revealed a significant main effect for Time ($F(1, 6) = 6.65$, $p = 0.04$), confirming that students across both conditions achieved significant learning gains during the instructional cycle. The main effect for Condition was not statistically significant ($F(1, 6) = 0.07$, $p = 0.80$). Regarding the interaction effect (Time x Condition), although the AI-assisted group showed a more pronounced upward trend in scores, the effect did not reach statistical significance within this pilot sample ($F(1, 6) = 2.98$, $p = 0.13$) (see Figure 3).

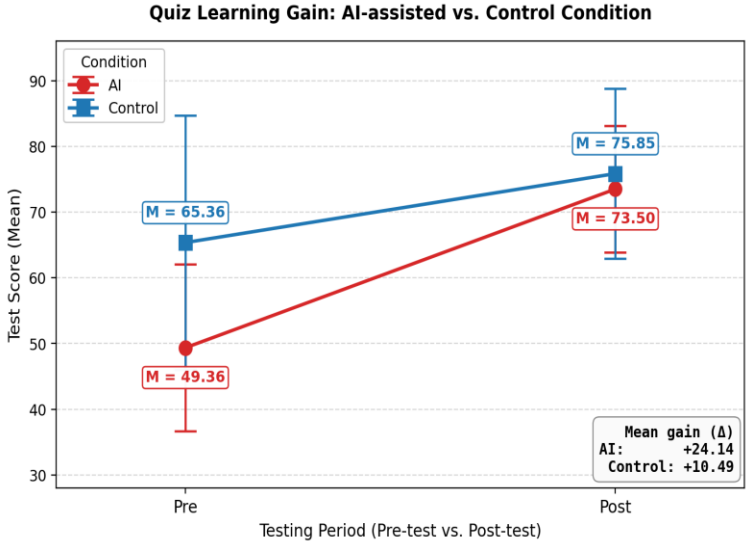


Figure 3. Comparison of pre- and post-test Quiz scores between AI-assisted and control conditions

To further examine the practical significance of the intervention, an independent t-test was performed on the gain scores (Post-test minus Pre-test). The AI-assisted condition yielded a substantially higher average gain ($M = 24.14$) compared to the control condition ($M = 10.49$). While the difference was not statistically significant ($t(14.25) = 1.37$, $p = 0.19$) due to the limited sample size of the pilot test, the calculated Cohen's d was 0.63 (see Figure 4). According to Cohen's conventions, a value of 0.63 represents a medium-to-large effect size, suggesting that the AI-assisted system has a meaningful impact on accelerating student learning. These findings indicate that while the current pilot test was underpowered to detect a significant interaction at the $p < 0.05$ level, the observed trends and effect size provide strong preliminary evidence for the efficacy of the AI-assisted approach in introductory programming education.

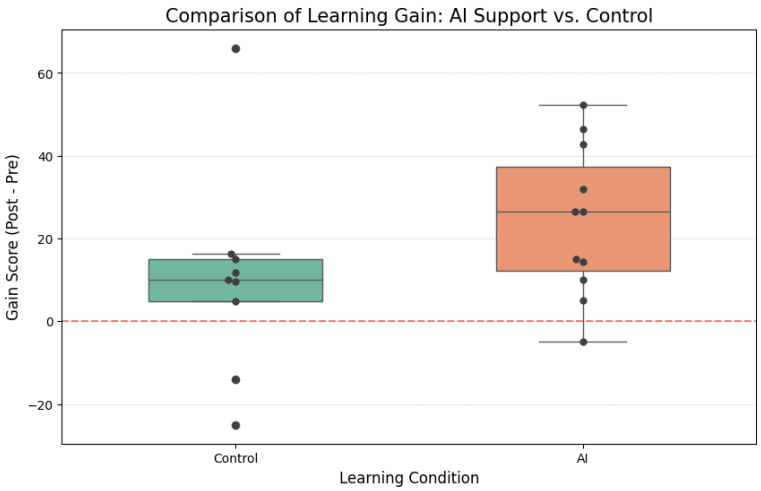


Figure 4. Distribution of Quiz gain scores by condition

Academic Performance (Q&A Sessions)

To further evaluate the impact of the AI-assisted system on higher-order thinking, the analysis was conducted on the Q&A items included in the pre- and post-tests. Similar to the quiz analysis, only validated datasets from students who completed all Q&A components were utilized. Due to the crossover nature of the study and the strict data cleaning process, the available data were utilized as follows. For the two-way repeated measures ANOVA, the analysis was restricted to the 3 students who successfully completed the entire sequence of all four Q&A measurement points across both conditions. For the independent t-test, all valid pre-post pairs were included, resulting in 14 observations (n = 7 for the AI-assisted condition and n = 7 for the control condition).

The ANOVA revealed a significant main effect for Time ($F(1, 2) = 20.60$, $p = 0.045$), confirming that the instructional cycle significantly improved students' programming abilities. The main effect for Condition was not statistically significant ($F(1, 2) = 0.35$, $p = 0.61$). Additionally, the interaction effect (Time × Condition) did not reach statistical significance ($F(1, 2) = 0.89$, $p = 0.446$).

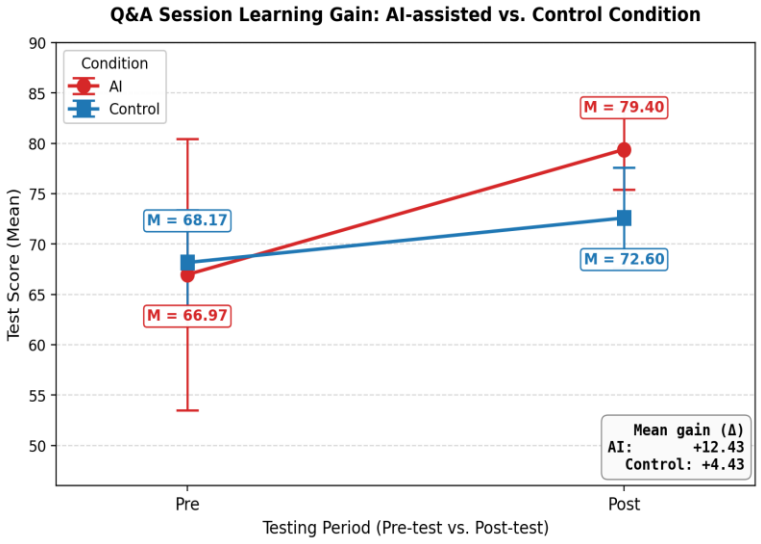


Figure 5. Comparison of pre- and post-test Q&A scores between AI-assisted and control conditions

To further examine the practical significance of the intervention, an independent t-test was performed on the gain scores of the Q&A items. The analysis revealed that the AI-assisted condition yielded

a substantially higher mean gain (M = 12.43) compared to the control condition (M = 4.43). Although the p-value was slightly above the threshold (t(12) = 1.86, p = 0.081) due to the limited sample size of the pilot test, the calculated Cohen's d was 1.02. According to Cohen's conventions, a value of 1.02 represents a large effect size, providing strong evidence that the AI-assisted adaptive system has a high potential for enhancing practical programming capabilities. These findings, particularly the large effect size, suggest that the AI intervention had a practically superior impact on learning outcomes compared to the control condition.

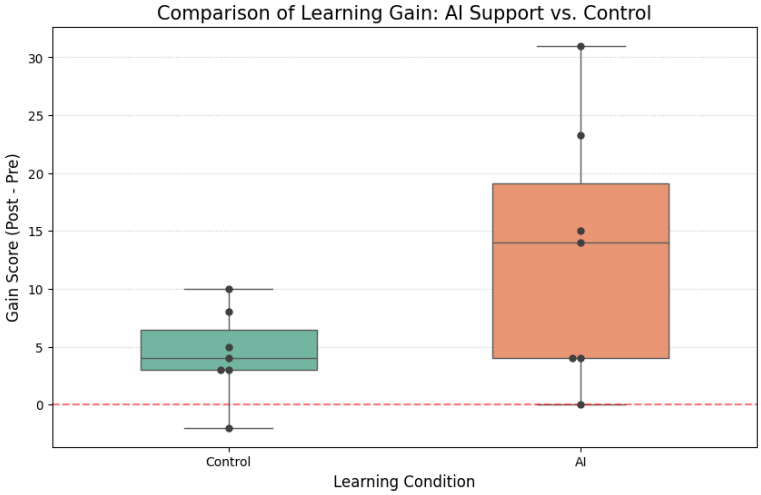


Figure 6. Distribution of Q&A session gain scores by condition

Table 2 provides a consolidated summary of the academic performance results across both assessment types.

Table 2. Summary of Academic Performance Results

Assessment	Analysis	n	Source	F	p-value	AI Gain	Ctrl Gain	Cohen's d	Effect
Quiz	ANOVA	7	Time	6.65	0.04*	M = 24.14	M = 10.49	0.63	Med-Lg
	ANOVA	7	Condition	0.07	0.80				
	ANOVA	7	Time × Cond.	2.98	0.13				
Q&A	ANOVA	3	Time	20.60	0.045*	M = 12.43	M = 4.43	1.02	Large
	ANOVA	3	Condition	0.35	0.61				
	ANOVA	3	Time × Cond.	0.89	0.446				

Note. *p < 0.05. ANOVA df: Quiz F(1, 6); Q&A F(1, 2). Gain scores from independent t-tests (Quiz: n = 11 AI, n = 9 control; Q&A: n = 7 per condition).

Perception of the AI-Assisted Adaptive Learning System (Survey)

To evaluate students' subjective experiences with the AI-assisted system, a post-experimental survey was administered (N = 11). Descriptive statistics showed that students perceived the system to be relatively easy to navigate (M = 3.67, SD = 0.99). While the mean scores for other psychological constructs were moderate, ranging from 2.33 (Engagement) to 2.77 (Self-Efficacy), the subsequent path analysis revealed significant structural relationships between these variables. The remaining five constructs measured alongside Perceived Ease of Use—Perceived Usefulness, Self-Efficacy, Engagement, Learning Effectiveness, and Satisfaction—all fell within this moderate range.

Structural Equation Modeling (SEM) was performed to examine the hypothesized structural relationships among the constructs; because this is a pilot sample, these results are presented as exploratory

associations rather than evidence of causation. The results of the path analysis are summarized in Table 3. The analysis revealed that Perceived Usefulness was significantly associated with both Self-Efficacy ($\beta = 0.81$, $p < 0.001$) and Engagement ($\beta = 0.73$, $p < 0.001$). Interestingly, Perceived Ease of Use did not show a statistically significant association with these constructs within this pilot sample. Given that the sample size ($N = 11$) falls well below conventional minima for structural equation modeling, all path estimates in this section should be interpreted as exploratory rather than confirmatory, and no causal claims should be drawn from them.

Both Self-Efficacy ($\beta = 0.40$, $p < 0.05$) and Engagement ($\beta = 0.57$, $p < 0.001$) were significantly associated with the Perceived Learning Effect. Finally, students' overall Satisfaction was significantly associated with three factors: Engagement ($\beta = 0.21$, $p < 0.05$), Self-Efficacy ($\beta = 0.27$, $p < 0.001$), and, most strongly, the Learning Effect ($\beta = 0.48$, $p < 0.001$). Although these path estimates should not be read as causal given the small pilot sample, the pattern is consistent with a model in which satisfaction with the AI-assisted Adaptive Learning System is associated with students' sense of competence and active involvement, which is in turn associated with perceived academic gains.

Table 3. SEM Analysis Results

Path	β	z-value	p-value
Perceived Ease of Use → Self-Efficacy	-0.097	-0.377	0.705
Perceived Ease of Use → Engagement	-0.009	-0.030	0.975
Perceived Usefulness → Self-Efficacy	0.813	4.669	0.000
Perceived Usefulness → Engagement	0.731	3.637	0.000
Self-Efficacy → Learning Effectiveness	0.396	2.444	0.014
Engagement → Learning Effectiveness	0.567	3.494	0.000
Self-Efficacy → Satisfaction	0.269	4.877	0.000
Engagement → Satisfaction	0.207	3.209	0.001
Learning Effectiveness → Satisfaction	0.483	5.853	0.000

Discussion of Findings

Interpretation of Academic Performance Results

The pilot study results provide encouraging preliminary evidence for the efficacy of the AI-assisted adaptive learning platform in introductory programming education (RQ1). The significant main effect for Time across both quiz and Q&A assessments confirms that the instructional cycle produced genuine learning gains regardless of condition. More importantly, the effect size analyses reveal a consistent pattern: the AI-assisted condition produced meaningfully larger gains than the control condition for both assessment types. The medium-to-large effect size for quiz scores ($d = 0.63$) and the large effect size for Q&A items ($d = 1.02$) suggest that the platform's impact may be particularly pronounced for higher-order thinking tasks requiring students to articulate and apply programming concepts in open-ended formats, rather than simply recognizing correct answers in structured assessments. It should be noted that the formal Time × Condition interaction tests did not reach statistical significance for either assessment (quiz $p = 0.13$; Q&A $p = 0.446$), reflecting the limited statistical power of the pilot sample; the effect-size analyses are therefore presented as the primary basis for this preliminary RQ1 finding.

That the Q&A assessments yielded a larger effect size than the quizzes is pedagogically significant. Open-ended programming tasks require students to construct solutions, explain reasoning, and demonstrate applied understanding, skills that align closely with the type of scaffolded, personalized feedback the AI-assisted system provides. This pattern is consistent with the platform's design philosophy, grounded in Vygotsky's (1978) ZPD, of providing guided support within the learner's developmental range rather than

direct answers. The fact that interaction effects did not reach statistical significance is attributable to the limited sample size inherent in a pilot study and should not be interpreted as evidence of absence; the effect sizes provide a more meaningful indicator of practical impact at this stage of the research program.

Interpretation of Survey Results

The exploratory SEM analysis revealed a coherent and theoretically consistent pattern of associations (RQ2, RQ3), though, given the small sample, these patterns should be viewed as hypothesis-generating rather than confirmatory. The strong association between Perceived Usefulness and both Self-Efficacy and Engagement is consistent with TAM research and suggests that students who perceived the AI-assisted system as functionally valuable for their coursework were more likely to report confidence in their abilities and sustained attention in the learning process. The non-significant association involving Perceived Ease of Use is a noteworthy finding: it suggests that for this population of CS students, the functional utility of the platform may have been more consequential than its surface-level usability, a pattern consistent with research showing that perceived usefulness becomes dominant as users gain experience with a technology (Davis, 1989).

The pathway from Engagement and Self-Efficacy through Learning Effectiveness to Satisfaction is consistent with a cascading pattern of association, in which satisfaction with the AI-assisted system appears linked primarily to students' sense of competence and active involvement, which is in turn associated with perceived academic gains; replication with a larger sample would be needed to test this pattern as a causal model. This exploratory finding nonetheless has potential implications for the design of AI-assisted educational tools, tentatively suggesting that developers may wish to prioritize features that enhance students' sense of capability and promote active, sustained interaction with the learning material.

A particularly noteworthy aspect of these results is what may be termed an engagement paradox. While Engagement registered the lowest mean score among all surveyed constructs (M = 2.33), the AI-assisted condition produced the largest learning gains observed in the study, with a Cohen's d of 1.02 on the open-ended Q&A assessments. This apparent dissociation between subjective engagement and objective learning outcomes suggests that students may have benefited substantially from the platform without subjectively experiencing the kind of sustained enthusiasm or deep involvement that the engagement scale was designed to capture. Several explanations are plausible. First, students may have perceived the platform primarily as an instrumental tool rather than as an inherently engaging activity, valuing it for its results rather than for the experience of using it. Second, the cognitive demands of mastery-based programming exercises and AI-mediated Socratic feedback may produce productive struggle that does not register as enjoyable engagement in the moment but nonetheless promotes deep learning, consistent with research on desirable difficulties. Third, the relatively brief two-week intervention may have been insufficient for engagement to develop in the way it might over a full semester. Whatever the underlying mechanism, this finding cautions against treating self-reported engagement as a proxy for learning effectiveness in AI-assisted learning environments and suggests that future evaluations should rely on multiple indicators rather than perception measures alone.

Relationship to Prior Work

The present findings advance the research program initiated by Kwak et al. (2023), who proposed the architectural framework for a generative AI-based adaptive learning system and demonstrated its feasibility through prototype testing. That earlier work established that generative AI could successfully generate programming tasks and evaluate learner responses. The current study extends this contribution by reporting results from a fully implemented platform deployed in an authentic classroom setting with a controlled experimental design. The results confirm and strengthen the initial feasibility findings by demonstrating not only that the system functions as intended but that it produces educationally meaningful effects on student learning outcomes.

Comparison with Existing Platforms

The platform occupies a distinct position in the landscape of CS education technology. Commercially available platforms such as Codecademy and freeCodeCamp provide interactive coding exercises but lack mastery-based progression systems and AI-powered personalized feedback. Khan Academy's integration of GPT-4 through Khanmigo represents the closest parallel, but it targets a broad range of subjects rather than providing a purpose-built CS I experience, and its AI tutoring features require a paid subscription. The present platform's combination of a structured mastery curriculum, a comprehensive AI tool suite, and its focus specifically on introductory CS, paired with entirely free access and alignment with a free open-access textbook through the ALG initiative, distinguishes it from existing offerings and embodies a model for equitable AI-augmented education. To our knowledge, this combination of mastery-gated progression, a multi-tool AI suite spanning debugging, code review, and Socratic Q&A, and fully free public access, evaluated together within an authentic classroom deployment, has not been previously reported in the CS I literature.

Implications for Research and Practice

This study contributes to the growing body of research on AI-assisted education by providing empirical evidence, through a controlled crossover design, that generative AI-based adaptive learning systems can produce meaningful learning gains in introductory programming courses. For researchers, the medium-to-large and large effect sizes observed in this pilot provide a basis for power analysis in designing larger-scale studies. The SEM findings contribute to the theoretical understanding of how AI-assisted learning tools influence student outcomes through the interplay of perceived usefulness, self-efficacy, engagement, and satisfaction.

For practitioners, the findings suggest that AI-assisted adaptive learning systems can serve as effective supplements to traditional instruction, particularly for tasks requiring applied problem-solving and code construction. The platform's open-access design demonstrates that meaningful AI-augmented instruction need not be locked behind subscription paywalls, a consideration of particular importance for institutions serving financially constrained student populations. The architectural framework, originally proposed by Kwak et al. (2023) and implemented in this work, provides a replicable model that other institutions can adapt for their own CS curricula.

AI Risk Mitigation and Responsible Deployment

Beyond demonstrating learning gains, the platform's design directly confronts several risks associated with deploying generative AI in instructional settings. Unconstrained LLMs are prone to two failure modes with particular relevance to novice programmers: they readily supply complete solutions that short-circuit the productive struggle central to skill acquisition, and they hallucinate plausible but incorrect explanations that learners lack the expertise to detect (Fan et al., 2025; Isaeva et al., 2025). The structured-prompt scaffolding, content-scope constraints, and output verification checks described earlier (see Design Rationale and Safeguards) represent concrete engineering responses to these risks rather than incidental design choices; every AI-driven tool in the suite was built on the premise that an unconstrained model would, by default, undermine the pedagogical goal it was meant to serve. Framed this way, the present work engages with AI-related risk not as a downstream ethical add-on but as a first-order design constraint, situating the platform's contribution within the broader discourse on responsible AI deployment, in which mitigating over-reliance, hallucination, and misuse is inseparable from questions of educational effectiveness.

Limitations

Several limitations should be considered when interpreting these results. First, the pilot study was conducted with a small sample (N = 20, with validated datasets from 13 students for academic analyses and

11 for survey analyses), limiting statistical power and generalizability. The crossover design, while methodologically rigorous, further reduced the number of students completing all measurement points for the repeated measures analyses. Second, the study was conducted within a single CS I course at one institution, and results may not generalize to other programming languages, course levels, or institutional contexts. Third, the two experimental modules (Dictionaries & Sets, Classes & Objects) represent only a subset of the CS I curriculum; it is unknown whether the observed effects would hold across different topic areas with varying levels of conceptual difficulty. Fourth, the platform's dependence on the OpenAI API introduces reliance on a proprietary external service whose behavior and pricing may change. Fifth, the moderate mean scores observed for engagement and other psychological constructs in the survey suggest room for platform refinement, and the non-significant effect of perceived ease of use may indicate opportunities for interface improvement. Sixth, one of the co-authors designed and developed the platform evaluated in this study; although data collection, scoring, and analysis were conducted independently of that author's development decisions, this author-developer proximity is a potential source of bias that a pilot study of this scale cannot fully rule out, and future work should include evaluation by research teams without a direct role in system development. Finally, as a pilot study, these findings are preliminary and require replication with larger samples and across multiple semesters. More specifically, the ANOVA Time × Condition interaction—the direct test for RQ1—did not reach statistical significance for either the quiz ($p = 0.13$) or Q&A ($p = 0.446$) assessments, and the survey-based path analysis ($N = 11$) is well below conventional sample-size minima for structural equation modeling, so the path coefficients reported in Table 3 should be regarded as exploratory.

Conclusions

This paper presented the design, implementation, and preliminary empirical evaluation of an AI-powered adaptive learning platform for introductory Python programming education. Building on the architectural framework proposed by Kwak et al. (2023), the platform integrates a 14-chapter mastery-based curriculum with a 5-gate progression system and a comprehensive suite of LLM-driven educational tools. A within-subjects crossover pilot study conducted during the Spring 2026 semester provided encouraging results: significant learning gains were observed across conditions, and effect size analyses revealed medium-to-large ($d = 0.63$) and large ($d = 1.02$) practical advantages for the AI-assisted condition on quiz and Q&A assessments, respectively. SEM analysis of survey data identified perceived usefulness as the primary driver of self-efficacy and engagement, which cascaded through learning effectiveness to overall satisfaction. Although the small pilot sample constrained the statistical power of the formal interaction tests and the SEM path estimates, the consistent direction and magnitude of these effects indicate that the platform is a promising approach worthy of larger-scale evaluation.

As large language models become more widely integrated into educational settings, platforms like the one presented in this study will play an increasingly important role in personalizing the learning experience for students who might otherwise be underserved by traditional instruction. The challenge ahead is ensuring that AI serves as a scaffold for deeper learning rather than a shortcut around it and ensuring that the most powerful educational tools are not reserved for those who can afford them. By providing free, open-access, AI-powered adaptive instruction aligned with a free textbook and embedded within a public university's academic infrastructure, this work offers a model for how generative AI can be deployed not just to enhance education, but to democratize it.

Future Work

Several directions for future work are planned. A larger-scale comparative study spanning a full semester will provide the statistical power necessary to detect interaction effects and enable more robust claims about the platform's impact. Investigation into which specific platform features (e.g., Debug Lab, Study Buddy, Code Review) contribute most to learning gains will inform prioritized development efforts.

Work is also underway to replace the OpenAI API with a custom, fine-tuned open-source language model trained specifically on programming and CS fundamentals, which would eliminate ongoing API costs, provide full control over model behavior, and ensure long-term sustainability. The instructor dashboard will be expanded with more advanced learning analytics capabilities, including predictive indicators for at-risk students, longitudinal performance trends, and exportable reports aligned with institutional assessment requirements. Integration with institutional learning management systems could enable seamless incorporation into existing course workflows.

Acknowledgement

This research was supported by Affordable Learning Georgia under Research Grant RG025. The authors gratefully acknowledge this funding, which made the development and evaluation of the platform possible. Any opinions, findings, and conclusions expressed in this paper are those of the authors and do not necessarily reflect the views of the funding agency.

References

Alhajri, R., Aldajani, S., Alaskar, B., & Al-Shehri, M. (2025). Artificial intelligence in personalized learning: A global systematic review. *Heliyon, 11*(22). https://doi.org/10.1016/j.heliyon.2025.e41234

Al-Rahmi, W. M., Othman, M. S., & Yusuf, L. M. (2015). The role of social media for collaborative learning to improve academic performance of students and researchers in Malaysian higher education. *International Review of Research in Open and Distributed Learning, 16*(4), 177–204.

Becker, B. A., Denny, P., Finnie-Ansley, J., Luxton-Reilly, A., Prather, J., & Santos, E. A. (2023). Programming is hard — or at least it used to be: Educational opportunities and challenges of AI code generation. *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1*, 500–506. https://doi.org/10.1145/3545945.3569759

Bloom, B. S. (1968). Learning for mastery. *Evaluation Comment, 1*(2), 1–12.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, A., Creel, K., Davis, J. Q., Demszky, D., Donahue, C., . . . Liang, P. (2021). On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258.*

Compeau, D. R., & Higgins, C. A. (1995). Computer self-efficacy: Development of a measure and initial test. *MIS Quarterly, 19*(2), 189–211.

Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly, 13*(3), 319–340. https://doi.org/10.2307/249008

Denny, P., Kumar, V., & Giacaman, N. (2023). Conversing with Copilot: Exploring prompt engineering for solving CS1 problems using natural language. *Proceedings of the 54th ACM Technical*

Symposium on Computer Science Education V. 1, 1136–1142. https://doi.org/10.1145/3545945.3569823

Dwivedi, Y. K., Kshetri, N., Hughes, L., Slade, E. L., Jeyaraj, A., Kar, A. K., Baabdullah, A. M., Koohang, A., Raghavan, V., Ahuja, M., Albanna, H., Albashrawi, M. A., Al-Busaidi, A. S., Balakrishnan, J., Barlette, Y., Basu, S., Bose, I., Brooks, L., Buhalis, D., Carter, L., . . . Wright, R. (2023). "So what if ChatGPT wrote it?" Multidisciplinary perspectives on generative conversational AI. *International Journal of Information Management, 71*, 102642.

Ennouamani, S., & Mahani, Z. (2017). An overview of adaptive e-learning systems. *Proceedings of the 2017 Eighth International Conference on Intelligent Computing and Information Systems* (pp. 342–347). IEEE.

Fan, Y., Tang, L., Le, H., Shen, K., Tan, S., Zhao, Y., & Shen, Y. (2025). The potential for AI to undermine learner agency and metacognition. In *Mapping the Evolution of AI in Education.* ScienceDirect.

Fredricks, J. A., Blumenfeld, P. C., & Paris, A. H. (2004). School engagement: Potential of the concept, state of the evidence. *Review of Educational Research, 74*(1), 59–109.

Garcia-Martinez, I., Fernandez-Batanero, J. M., Fernandez-Cerero, J., & Leon, S. P. (2023). A meta-analysis of adaptive learning technologies in education. *Computers & Education, 198*, 104764.

Gomes, A., & Mendes, A. (2007). An environment to improve programming education. *Proceedings of the 2007 International Conference on Computer Systems and Technologies* (pp. 1–6).

Isaeva, R., Bekova, A., & Asilbekova, G. (2025). Enhancing learning effectiveness through adaptive learning platforms and emerging computer technologies. *Jurnal Ilmiah Ilmu Terapan Universitas Jambi, 9*, 144–160.

Jonassen, D., Howland, J., Marra, R. M., & Crismond, D. (2008). *Handbook of Research on Educational Communications and Technology.* Routledge.

Kabudi, T., Pappas, I., & Olsen, D. H. (2021). AI-enabled adaptive learning systems: A systematic mapping of the literature. *Computers and Education: Artificial Intelligence, 2*, 100017.

Kang, N., Liu, J., Wang, X., & Zhang, Y. (2025). The research hotspots and future trends of adaptive learning in the age of AI: A bibliometric analysis. *Journal of Nursing Management.* https://doi.org/10.1155/jonm/5765893

Kasneci, E., Sessler, K., Küchemann, S., Bannert, M., Dementieva, D., Fischer, F., Gasser, U., Groh, G., Günnemann, S., Hüllermeier, E., Krusche, S., Kutyniok, G., Michaeli, T., Nerdel, C., Pfeffer, J., Poquet, O., Sailer, M., Schmidt, A., Seidel, T., . . . Kasneci, G. (2024). ChatGPT for good? On the implications of generative AI for education and society. *Learning and Individual Differences, 103*, 102274.

Kwak, M., Jenkins, J., & Kim, J. (2023). Adaptive programming language learning system based on generative AI. *Issues in Information Systems, 24*(3), 222–231. https://doi.org/10.48009/3_iis_2023_119

Li, F., He, Y., & Xue, Q. (2021). Progress, challenges and countermeasures of adaptive learning. *Educational Technology & Society, 24*(3), 238–255.

Lim, W., Gunasekara, A., Pallant, J., Pallant, J., & Pechenkina, E. (2023). Generative AI and the future of education. *The International Journal of Management Education, 21*(2).

Luxton-Reilly, A., Simon, Albluwi, I., Becker, B. A., Giannakos, M., Kumar, A. N., Ott, L., Paterson, J., Scott, M. J., Sheard, J., & Szabo, C. (2018). Introductory programming: A systematic literature review. *Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*, 55–106.

Marouf, A., Yousef, M., Mukhaimer, M., & Abu-Naser, S. (2018). An intelligent tutoring system for learning introduction to computer science. *International Journal of Academic Multidisciplinary Research,* 1–8.

Müller, S., Bergande, B., & Brune, P. (2018). Robot tutoring: On the feasibility of using cognitive systems as tutors. *Proceedings of the 3rd European Conference of Software Engineering Education* (pp. 45–49).

Nielsen, J. (1994). Usability heuristics for user interface design. Nielsen Norman Group. https://www.nngroup.com/articles/ten-usability-heuristics/

Oliver, R. L. (1980). A cognitive model of the antecedents and consequences of satisfaction decisions. *Journal of Marketing Research, 17*(4), 460–469.

Prather, J., Denny, P., Leinonen, J., Becker, B. A., Albluwi, I., Craig, M., Keuning, H., Kiesler, N., Kohn, T., Luxton-Reilly, A., MacNeil, S., Petersen, A., Pettit, R., Reeves, B. N., & Savelka, J. (2023). The robots are here: Navigating the generative AI revolution in computing education. *Proceedings of the 2023 Working Group Reports on Innovation and Technology in Computer Science Education*, 108–159. https://doi.org/10.1145/3623762.3633499

Sweller, J. (1988). Cognitive load during problem solving: Effects on learning. *Cognitive Science, 12*(2), 257–285.

Vygotsky, L. S. (1978). *Mind in Society: The Development of Higher Psychological Processes.* Harvard University Press.

Watson, C., & Li, F. W. (2014). Failure rates in introductory programming revisited. *Proceedings of the 2014 Conference on Innovation and Technology in Computer Science Education*, 39–44.