Online-Journals.org

iJET International Journal of Emerging Technologies in Learning

iJET | eISSN: 1863-0383 | Vol. 21 No. 3 (2026) | OPEN ACCESS

https://doi.org/10.3991/ijet.v21i03.61437

PAPER

Designing for Self-Regulated Learning in AI-Assisted Quizzing: A Classroom Pilot Study

Michael Pin-Chuan Lin[1,2], Daniel H. Chang[2,3], Vasudevan Janarthanan[4], Michael S. Hsiao[5], Marco Ho[6], Jeeho Ryoo[4] (✉)

[1]Mount Saint Vincent University, Halifax, NS, Canada

[2]Simon Fraser University, Burnaby, BC, Canada

[3]Athabasca University, Athabasca, AB, Canada

[4]Fairleigh Dickinson University, Vancouver, BC, Canada

[5]Virginia Tech, Blacksburg, VA, USA

[6]British Columbia Institute of Technology, Burnaby, BC, Canada

j.ryoo@fdu.edu

ABSTRACT

AI support is increasingly embedded in online quizzes, yet instructors often lack clear, actionable signals about where students struggle during those assessments. We present a classroom-deployed quiz system that combines integrity-preserving hinting (TA-AI), trace summarization (Analytics-AI), and feedback-driven refinement (AI-Improver) to generate instructor diagnostics from routine interaction logs. The system was used in a graduate assembly programming course over five quiz weeks (N = 18). We report deployment evidence focused on reliability and instructional usefulness for monitoring: prompt-intent coding reached substantial agreement (Cohen's kappa [κ] = 0.81); fixed-effects models (with student and item controls) showed a negative association for one-hint interactions (odds ratio [OR] = 0.231, indicating approximately 77% lower odds of a correct response for single-hint interactions relative to 0-hint interactions); and item-level demand spikes were operationalized via a demand × success prioritization process for weekly review. Rather than producing automated judgments or claims of causal learning gains, the analytics are designed as practical prioritization cues that direct instructor attention toward high-need items during AI-assisted quizzes.

KEYWORDS

self-regulated learning (SRL), AI in education, learning analytics, computer science education, assessment

1 INTRODUCTION

Generative AI is increasingly part of programming assessments in computer science courses, but it introduces a practical visibility problem for instructors [1], [2], [20]. In a conventional quiz workflow, struggle is often visible through failed attempts, syntax errors, and incremental debugging behavior. In AI-assisted workflows, many of those cues are compressed into short prompt–response exchanges, so a correct submission no longer clearly indicates whether a student resolves a concept through guided reasoning or simply requested direct help.

Lin, M. P. C., Chang, D. H., Janarthanan, V., Hsiao, M. S., Ho, M., Ryoo, J. (2026). Designing for Self-Regulated Learning in AI-Assisted Quizzing: A Classroom Pilot Study. *International Journal of Emerging Technologies in Learning (iJET)*, 21(3), pp. 30–43. https://doi.org/10.3991/ijet.v21i03.61437

Article submitted 2026-03-09. Revision uploaded 2026-04-15. Final acceptance 2026-04-18.

© 2026 by the authors of this article. Published under CC-BY.

This loss of visibility has direct instructional consequences. Instructors still need to decide which items are producing friction, which misconceptions are recurring, and where to spend limited review time between weeks. Most courses lack high-cost instrumentation (e.g., eye tracking or think-aloud protocols), so routine interaction traces are often the only scalable evidence source [3], [4]. The core challenge is not data scarcity, but whether ordinary logs can be interpreted in ways that are both useful and instructionally responsible.

In this paper, we address this challenge by treating AI-assisted quiz traces as instructor-facing diagnostics, not automated judgments. The system design combines bounded hinting, structured logging, and weekly analytics so that instructors can review where demand accumulates and how that demand relates to student success. This framing keeps the analysis grounded in classroom action: the goal is to help instructors prioritize review and revise targeted items, not to replace pedagogical decision-making.

Our deployment contributes to three applied elements that are intended to be reusable in similar courses. First, we present a role-separated system design with integrity-preserving hint guardrails and auditable analytics outputs. Second, we report classroom evidence from a five-week in-situ deployment (N = 18), showing that routine logs can be transformed into interpretable indicators, including reliable intent coding ($\kappa = 0.81$), demand spike identification, demand × success prioritization, and fixed-effects associations between hint use and correctness. Third, we translate these indicators into a weekly instructor workflow that clarifies what to inspect first and what to address next.

To organize this evidence, we use three research questions that move from association patterns to temporal stability to instructor utility. This progression matters in applied educational technology because instructors need signals that are statistically defensible, interpretable across weeks, and usable within routine course constraints.

1.1 Research questions

- RQ1: Relationship of hint use and correctness. How are hint-use categories associated with item correctness under student- and item-level controls?
- RQ2: Stability of help-seeking over time. Are help-seeking patterns stable across quiz weeks, or do they vary by weekly context?
- RQ3: Utility of item-level demand diagnostics. How effectively do demand spikes and the demand × success review matrix localize items for instructor review?

Together, these questions set the paper's scope: we present and evaluate a practical diagnostic workflow for AI-assisted quizzing, and we interpret the resulting signals to support instructor review in an authentic classroom deployment.

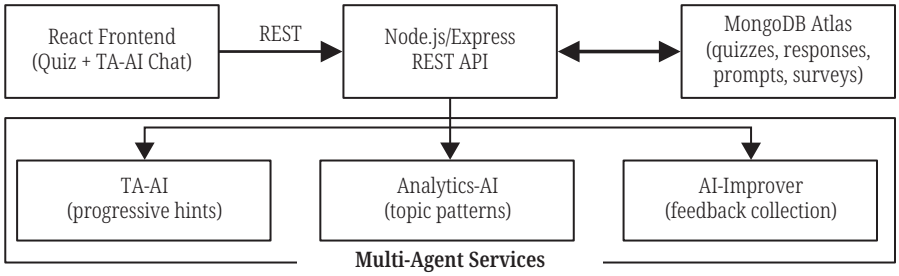

Fig. 1. System architecture of the role-separated AI quiz application

2 RELATED WORK AND CONCEPTUAL FRAMING

Research on AI-supported learning has established a consistent tension between access and effort [2], [5]. On one hand, AI assistance can reduce immediate friction and help learners continue working when they encounter impasses. On the other hand, if assistance is unconstrained or overly directive, it can shorten the reasoning process that assessment tasks are meant to surface. This tension is especially visible when learners take programming quizzes, where correct outputs can be produced even when underlying understanding remains partial [1].

From self-regulated learning (SRL) perspectives, researchers clarify learners' intention to seek assistance by separating adaptive help-seeking from executive help-seeking [6], [7]. Practically, this suggests quiz system designs should not remove help; instead, they should structure it so requests reveal information needs rather than turning into unlimited answer retrieval. Our deployment adopts this practical interpretation by using bounded hint access and by combining hint volume with prompt-intent evidence.

In terms of quiz items, Cognitive Load Theory (CLT) has been used as a guiding, complementary framework to conceptualize whether students' elevated demand for help seeking can signal either intrinsic conceptual complexity or extraneous cognitive load, such as prompt wording and task structure [8]. In classroom settings, this motivates demand monitoring at the item-week level, because aggregated demand patterns can reveal bottlenecks that are not visible in total score summaries alone. Cognitive apprenticeship reinforces this design choice by emphasizing scaffolded guidance that supports reasoning without displacing it [9], [10].

A motivational perspective from Self-Determination Theory (SDT) further supports these constraints. When students are given support that preserves autonomy and competence, help can remain instructional rather than purely substitutive [11], [12], [13]. In AI-assisted quiz practice, this translates into explicit hint budgets, progressive responses, and transparent reporting so instructors can review how assistance is being used [18], [19].

Despite these theoretical advances, many validated SRL and cognitive-load approaches still rely on intensive instrumentation that is difficult to sustain in ordinary classroom operation [3]. The practical gap is therefore not a lack of theory but a lack of portable workflows that convert routine traces into instructional decisions [4]. Our work addresses that gap through an in-situ deployment-centered approach that prioritizes interpretable analytics, weekly instructor review, and bounded claims suitable for day-to-day teaching.

3 SYSTEM AND TOOL DESIGN

The system was designed for a practical instructional setting in which quizzes must remain assessable while AI support is available. Instead of bundling assistance and analytics into a single opaque pipeline, the implementation separates student-facing support from instructor-facing interpretation so each step can be inspected. The design supports quizzes while preserving evidence instructors can use in weekly review. In use, students request bounded hints while solving items, the platform records traces in routine operation, and instructors receive weekly demand and success summaries. The system is organized around three interlocking components: a role-separated architecture, an SRL interaction model, and a prompt-type taxonomy; each is described in turn.

3.1 Role-separated architecture

The application separates three roles that work together during deployment: TA-AI for student hinting, Analytics-AI for instructor diagnostics, and AI-Improver for post-quiz refinement. This separation keeps instructional support, reporting, and maintenance distinct enough that instructors can inspect each stage without relying on a single opaque assistant [17].

As shown in Figure 1, students interact through a React interface connected to a stateless Node.js/Express backend and MongoDB logging layer (see Figure 2 for the student-facing chat interface). Before any hint is generated, the backend enforces policy constraints (maximum 3 hints per item and 5 per quiz), preserving both assessment integrity and interpretability of demand traces. The same trace stream is then aggregated for instructor review and reused by AI-Improver for post-quiz prompt refinement.

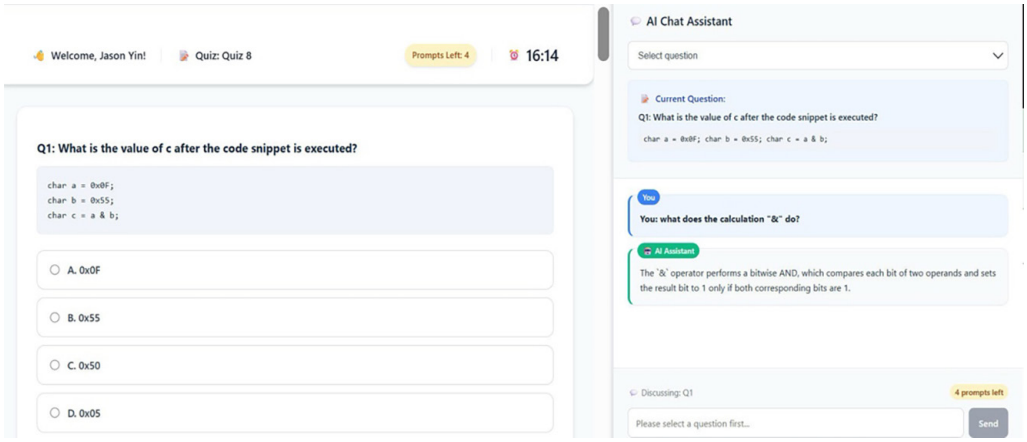

Fig. 2. Student-facing chat interface with progressive hints and remaining hint budget

3.2 Prompting protocols and guardrails

TA-AI and Analytics-AI are governed by fixed prompt templates that enforce role boundaries. TA-AI is constrained to hints only (no answer reveal, no option elimination, and no final solution) and must enforce the item/quiz caps at request time. Analytics-AI produces a structured weekly report from item-week aggregates, including a data-quality gate, spike list, demand × success mapping, and interpretation boundaries. Across both services, prompts enforce the same policy boundary: support and summaries are allowed; direct answers and causal claims are not.

3.3 Implementation stack

The deployment used a lightweight web stack on the Chameleon Testbed, so the workflow could run inside ordinary course infrastructure rather than a bespoke research platform [21]. In the pilot, we deployed on Chameleon Cloud using an m1.medium (4 vCPUs, 8 GB RAM) instance, with a Node.js/Express v5.1.0 backend, a React 18.2.0 + TypeScript 5.7.2 frontend, MongoDB for data storage, and the OpenAI API gpt-4o SDK for model service integration. This implementation profile matters for transfer: the architecture depends on standard web components and explicit

policy enforcement rather than specialized hardware or lab-only tooling. This allows similar assessment settings to adopt the pattern while retaining the same guardrail and reporting logic described in this paper.

4 METHODOLOGY

This study was an in-situ classroom deployment of AI-assisted quizzing in an existing course workflow. Methods prioritize operational transparency: how traces were recorded, how indicators were defined, and how those indicators were analyzed for instructor-facing use.

4.1 Study context and participants

The deployment took place in Summer 2025 in an Assembly Programming course (a 14-week graduate course) at a university in western Canada. The course was designed and overseen by the course instructor, who also deployed the AI-hint system. The course enrolled 18 graduate students (N = 18) with varied prior general programming experience, and the system was used during five quiz weeks (Weeks 2, 3, 7, 8, and 9) distributed across the term within routine graded course operation; quiz attempts counted toward final grades. This context was appropriate for studying AI-assisted help-seeking because assembly programming tasks are often conceptually dense and error-sensitive, which makes support demand visible when item design or prior knowledge is misaligned. At the same time, the graded setting preserved authentic stakes, so the recorded interactions reflect real assessment behavior rather than laboratory-only task engagement. This study received Institutional Review Board exemption. Students were informed that interaction logs would be used for instructional analytics research, and all records were pseudonymized prior to analysis.

4.2 Procedure

The weekly procedure followed the same sequence across the five deployment weeks. Students attempted quiz items in the web interface, requested hints when needed, and submitted item responses under normal course timing and grading constraints. This repeated structure allowed cross-week comparisons without introducing separate experimental sessions. At the interaction level, hint access was explicitly bounded to preserve constrained help-seeking: each item allowed up to 3 hints, and each quiz allowed up to 5 hints total. When students requested help, the system returned progressive non-answer guidance through the chat interface, then logged the request and response context before students proceeded with the item. All interaction events were recorded with pseudonymized identifiers and linked metadata, including week, item, hint count, prompt text (when provided), timestamps, and correctness. To verify baseline policy fidelity, we performed an automated phrase scan of all 179 generated hints for explicit answer-disclosure patterns (e.g., “the answer is,” “correct answer,” and “choose A/B/C”); this scan returned 0 matches. Note that the scan terms are not exhaustive and may not capture all near-answer phrasings.

4.3 Data and units of analysis

The analysis uses three aligned units so that student behavior, item demand, and week-level stability can be examined without mixing levels of inference. The primary unit is the student-item interaction (n = 820), which captures hint use and correctness at the most granular level used in the fixed-effects models. To support instructor-facing monitoring, student-item traces were aggregated into item-week observations (n = 50) for demand-spike and review-prioritization analysis and into student-week observations (n = 90) for stability analysis. Using these three units allows the results to answer different practical questions: where students request help, which items concentrate demand, and whether observed help-seeking patterns persist or shift across weeks.

4.4 Measures and operationalization

This subsection translates the conceptual framing into explicit measurement rules that can be audited from routine logs.

Hint usage categories. Hint use was binned into four categories (0, 1, 2, and 3+ hints per student-item interaction). In interpretation, these bins are treated as observable demand-intensity levels (independent, minimal, moderate, and heavy). The same bins support direct comparison across items and weeks.

Demand spike rules. Demand at the item-week level was defined as mean hints per item. An item-week was flagged as a demand spike only when both criteria were satisfied. The mean exceeded the item-week mean plus two standard deviations (threshold > 1.16 in this dataset), and demand was in the top 10% of item-week values (threshold > 0.776). Thresholds were derived from the distribution of hint-demand ratios across all 50 item-week observations (grand mean $\mu = 0.22$, SD = 0.47): the mean + 2SD criterion gives $\mu + 2\sigma = 1.16$, and the top-10% criterion corresponds to the 90th percentile of the distribution (p90 = 0.776). The dual-criterion design, which requires both conditions simultaneously, was chosen to reduce false positives: the mean + 2 SD rule alone flags statistical outliers that may reflect a single unusual week, while the top-10% rule alone flags high-volume items that may still fall within normal variance. Requiring both conditions together targets items that are simultaneously extreme relative to their own baseline and globally elevated in the dataset, yielding a more conservative and actionable spike list (N = 3 item-weeks flagged in this pilot: Week 7 Q1, Week 8 Q1, Week 9 Q1). As a sensitivity check, applying only the top-10% threshold (>0.776) would flag 5 item-weeks; applying only the mean + 2 SD threshold (>1.16) would flag 3 item-weeks; the intersection used here flagged 3.

Demand × Success matrix bands. For review prioritization, each item-week was mapped to demand bands (High/Low) and success bands (<40%, 40–70%, and >70%). This cross-classification combines demand and performance context in a single frame. It was used to prioritize review actions such as redesign, clarification, monitoring, or routine maintenance.

Prompt intent codebook and coding protocol. Prompt-intent coding was applied only to interactions that contained student prompts. Categories were direct answer, concept clarification, procedure/strategy, task interpretation, and other. Two independent raters coded a stratified 25% sample by week and hint-volume strata, with categories as defined in Table 1, producing Cohen's $\kappa = 0.81$. We use this

channel to interpret whether high-demand interactions are concept-oriented, procedural, or direct-answer seeking, while recognizing that prompt coding captures only prompted interactions and not 0-hint behavior.

Table 1. Codebook for student prompt intent classification

Category	Definition & Examples
Direct Answer	Requests for the solution, specific values, or confirmation of a final answer. "Is the answer C?" "What is the value of x?"
Concept Clarification	Questions about underlying concepts, definitions, or language syntax. "What does big-endian mean?" "Explain the difference between *ptr and&ptr."
Procedure/Strategy	Requests for next steps, debugging help, or problem-solving strategies. "How do I start calculating the offset?" "Why is my loop not terminating?"
Task Interpretation	Questions clarifying the problem statement or constraints. "Do we assume a 32-bit or 64-bit system?" "Does 'word' mean 2 bytes or 4 bytes here?"
Other	Off-topic, greetings, or incoherent inputs. "Hello" "This is hard."

Hint quality rubric and audit protocol. Hint quality was assessed with a three-level rubric: Q1 (non-actionable), Q2 (helpful scaffold), and Q3 (near-answer). Two coders independently labeled a stratified sample of 100 prompt–response exchanges to stress-test boundary clarity. Agreement in this independent stress test was $\kappa = 0.29$, which falls in the 'fair' range under standard interpretation benchmarks and indicates that Q2/Q3 boundary distinctions were not reliably separable by independent coders in this pilot [14], [15]. Because this level of agreement does not meet acceptable thresholds for classification, the Q1/Q2/Q3 rubric is not used as an analytic measure in this study. It is retained here as a design description of the intended quality boundaries; all interpretive weight rests on demand-volume and intent-coding indicators, which achieved substantially higher reliability.

4.5 Analysis

The analysis follows the three research questions across aligned units of inference. For RQ1, we fit a fixed-effects logistic regression model with student and item fixed effects to estimate associations between hint category and correctness under repeated measures. For RQ2, we compute the intraclass correlation coefficient (ICC) on student-week hints-per-item; for RQ3, we apply the item-week spike rule and Demand × Success mapping to produce auditable instructor-facing diagnostics. In this RQ1 model, the dependent variable is whether a student answered an item correctly (1/0). Hint use category is entered as a categorical predictor with 0 hints as the reference level, and student and item fixed effects are included to absorb baseline differences across students and items. The reported coefficients and odds ratios, therefore, represent within-student, within-item associations between hint-use category and correctness, rather than between-student or between-item comparisons. This specification reduces confounding from unobserved baseline differences in student ability and item difficulty. Together, these procedures keep assumptions auditable and keep outputs aligned with the instructor's weekly review workflow.

Table 2. Confound-aware association between hint category and correctness (Student and Item Fixed Effects)

Contrast (vs. 0 hints)	Coef (β)	OR	95% CI OR	p
1 hint	–1.467	0.231	[0.093, 0.574]	0.0016
2 hints	–0.534	0.586	[0.128, 2.675]	0.4905
3+ hints	0.007	1.007	[0.157, 6.445]	0.9937

Notes: Average marginal predicted probability of a correct response with all other covariates held at their observed means: 0 hints (0.803), 1 hint (0.614), 2 hints (0.743), and 3+ hints (0.804). β = log-odds coefficient; OR = exp(β). OR below 1 indicates lower odds of a correct response vs. the 0-hint reference group. Estimated in R using logistic regression with student and item fixed effects (lme4 package). No correction for multiple comparisons was applied; findings should be interpreted as exploratory.

5 RESULTS

This section presents results in the same order as the research questions so that evidence, interpretation, and instructional use remain aligned. We first report the relationship between hint use and correctness (RQ1), then examine cross-week stability in help-seeking behavior (RQ2), and finally evaluate whether item-level demand diagnostics support practical weekly review decisions (RQ3).

5.1 RQ1: Relationship of hint use and correctness

The central RQ1 takeaway is that hint demand behaves as a meaningful difficulty signal, but not in a simple linear way. In this pilot, single-hint interactions are associated with lower correctness than 0-hint interactions, while higher hint bins are sparse and therefore less precise for inference. Hint usage was highly skewed: most interactions used 0 hints (742/820), and higher-hint bins were sparse (1 hint: 49; 2 hints: 19; 3+ hints: 10), which bounds inference for the upper bins. Table 2 reports the fixed-effects estimates for hint categories relative to 0 hints.

The fixed-effects estimates indicate that 1-hint interactions are associated with lower correctness than 0-hint interactions, consistent with help requests clustering around moments of difficulty rather than implying that hints themselves reduce performance [6], [7]. In plain terms: students who used one hint were more likely to have been working on a harder item, not penalized by the hint itself. The direction of the association reflects when help is sought, not what the hint does to learning.

The wider intervals for higher hint categories primarily constrain inference rather than contradict the diagnostic interpretation. The 3+ hint estimate (OR = 1.007, 95% CI [0.157, 6.445], p = 0.994) is based on only 10 observations and carries wide uncertainty; it is not interpreted substantively and is reported for completeness. Overall, this distribution is most informative as a diagnostic pattern rather than an evaluative ranking of students [6], [8], [9], [10].

To triangulate whether hint demand reflects conceptual clarification versus answer-seeking, we examined prompt text only in cases where students submitted a prompt alongside a hint request. Because intents are observable only when a prompt is present, this qualitative channel is inherently conditional on prompting and should be read as interpretive context rather than a global prevalence estimate. Prompt content was dominated by conceptual clarification and procedural strategy requests, while explicitly answer-seeking prompts were comparatively infrequent.

Across the N = 122 coded prompts, conceptual understanding requests accounted for 75.4% (n = 92), procedural and debugging queries for 19.7% (n = 24), and explicit answer-seeking for 4.9% (n = 6). This intent distribution operationalizes the SRL distinction between adaptive help-seeking (conceptual or procedural requests) and executive help-seeking (direct-answer requests) described earlier in the framework mapping [6], [7].

5.2 RQ2: Stability of help-seeking over weeks

RQ2 examines whether hint use is mostly consistent across weeks or mainly shaped by topic and item context. The main RQ2 result is that help-seeking is neither purely fixed nor purely random across weeks. The student-week ICC (2, 1) for hints-per-item was 0.426 (95% CI [0.20, 0.65]), indicating moderate between-student and between-week consistency; ICC values in the 0.50–0.75 range reflect moderate reliability, so 0.426 falls at the lower end of moderate [16]. This means that while some individual and topical consistency exists, a substantial portion of help-seeking variance reflects contextual factors such as weekly topic difficulty and item design.

Figure 3 shows frequent category shifts across weeks, including 58% of student-week observations changing category from the prior week. In practice, this supports weekly monitoring rather than fixed student labels: instructors should read help demand in relation to the current topic and item context. From an SRL perspective, this week-to-week movement is consistent with context-sensitive regulation rather than static help-seeking profiles [6], [7].

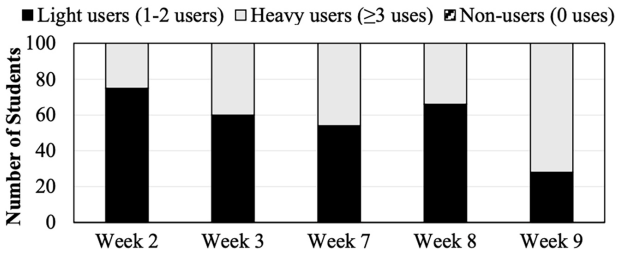

Fig. 3. Distribution of students by AI usage category each week

5.3 RQ3: Item-level demand spikes and practical usefulness for weekly review

RQ3 asks whether demand diagnostics can localize where instructors should look first. Using the predefined intersection rule from Section IV-D2, the stable spike set included Week 7 Q1 (1.47 hints), Week 8 Q1 (1.19), and Week 9 Q1 (2.53), indicating recurring concentration of support demand. CLT motivates treating repeated high-demand signals as candidates for item inspection rather than latent-difficulty claims [8].

Once spikes are identified, the Demand × Success matrix translates aggregate behavior into instructional prioritization. High-demand/low-success cases suggest redesign needs; high-demand/moderate-success cases often suggest clarification. This staged reading is consistent with cognitive apprenticeship and SDT [9], [10], [11], [12], [13]. Flagged items remain decision-support signals; instructors review wording and representative prompts before acting. Therefore, these indicators support a

weekly review process: identify high-demand items, interpret demand with success context, and then decide whether to clarify, revise, or monitor.

Table 3. Detailed question analysis with demand/success bands and instructor actions (Wk 2)

Question	Hint Requests	Success Rate	Review Category	Instructor Action
1	2	88.2%	Low Demand/High Success (Trivial)	—
3	3	23.5%	High Demand/Low Success (Redesign)	Added memory visualization live-coding segment.
7	3	29.4%	High Demand/Low Success (Redesign)	—
8	2	41.2%	High Demand/Moderate Success (Clarify)	—
9	7	64.7%	High Demand/Moderate Success (Clarify)	No action (monitor).

The review matrix maps demand jointly with success to distinguish redesign vs. clarification vs. monitoring actions. Table 3 makes this mapping concrete at the question level for Week 2, showing why similarly high-demand items may still receive different instructional responses depending on success context. Because prompt text is observable only when a prompt is present, we treat intent coding as the primary qualitative context channel and avoid further thematic aggregation in this pilot.

5.4 Integrity/discriminant checks

Intent audit reliability was substantial. Prompt-intent coding agreement on a stratified sample was substantial ($\kappa = 0.81$), which supports using intent categories as a discriminant channel when interpreting demand traces. In construct-validity terms, if high-demand interactions were dominated by direct-answer requests, hint counts could reflect executive help-seeking rather than adaptive regulation [6], [7]. Hint-quality coding was not reliable. The independent stress test produced $\kappa = 0.29$, which falls in the 'fair' range and does not support automated classification of hints as Q1, Q2, or Q3 [14], [15]. Accordingly, no analytic claims in this paper depend on hint-quality categories. The rubric is retained as a system design artifact describing the intended policy boundaries, and near-answer boundary judgments are left to instructor review rather than treated as a coded variable.

6 DISCUSSION

Cognitive Load Theory provides an item-level explanation for this pattern: elevated demand may reflect intrinsic complexity or extraneous friction [8]. Cognitive apprenticeship and SDT also motivate bounded scaffolding as support rather than penalty [9], [11], [12], [13], and [10].

Routine AI-quiz traces supported a usable instructor workflow without additional instrumentation. The most useful signal was the combination of demand intensity, success context, and prompt intent, which helped instructors prioritize where to inspect first. These diagnostics are best treated as prioritization tools, not adjudication tools: they localize likely friction, while instructors determine causes and actions.

RQ2 adds an operational nuance. With ICC (2, 1) = 0.426, 95% CI [0.20, 0.65], and frequent week-to-week category shifts, help-seeking appears partly stable but strongly context-sensitive [16]. This supports weekly monitoring over static student labels. In practice, instructors can review flagged high-demand items, compare demand against success bands, and choose redesign, clarification, or monitoring actions. Table 3 illustrates this with concrete Week 2 actions. (Note: Week 2 is a non-spike week included to illustrate the full range of demand × success combinations; the spike weeks are Weeks 7, 8, and 9.)

Interpretation remains bounded. The study is observational, help-seeking is endogenous, and there is no comparison condition without AI support. The hint-quality stress test ($\kappa = 0.29$ for Q2/Q3) also shows that near-answer boundary judgments are difficult to code reliably. For these reasons, the system is positioned as decision support with human review, not autonomous instructional decision-making.

7 LIMITATIONS

This pilot covers one graduate course (N = 18) and uses an observational design without a no-AI control group. Demand thresholds (mean plus two standard deviations and top-10%) are local operational defaults that may require recalibration in other courses. Finally, the independent hint-quality stress test produced $\kappa = 0.29$, which falls below acceptable reliability thresholds for classification. The Q1/Q2/Q3 rubric is therefore not used as an analytic measure, and quality interpretation is reserved for instructor review. Future work should revise boundary definitions and re-establish inter-rater reliability before using hint quality as a coded variable. Intent coding was applied to a stratified 25% sample; the remaining interactions are assumed to reflect a similar distribution, though this assumption is unverified and should be tested in future work with full coding coverage.

8 CONCLUSION

This paper addresses a practical assessment problem: AI assistance can reduce visible struggle cues during quizzes. We presented a role-separated workflow that combines integrity-preserving hinting, routine trace logging, and weekly diagnostics for instructor review. Across five quiz weeks (N = 18), the deployment produced usable monitoring evidence, including reliable intent coding ($\kappa = 0.81$), fixed-effects associations between hint use and correctness, moderate week-level stability (ICC (2, 1) = 0.426), and item-level demand-based prioritization. Future work should test portability across additional courses, expand the intent coding to the full dataset to verify the distribution assumption, and strengthen triangulation of demand and quality interpretations. Establishing a control-group comparison and increasing sample size would permit stronger causal inference about the impact of AI-assisted hint delivery on learning outcomes.

9 ACKNOWLEDGEMENTS

Results presented in this paper were obtained using the Chameleon testbed supported by the National Science Foundation [21]. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the Chameleon project or its sponsors. This research

was supported by the Social Sciences and Humanities Research Council of Canada (SSHRC), grant numbers 430-2024-00269 and 435-2026-1299 (PI: Michael Lin).

10 REFERENCES

[1] S. Lau and P. J. Guo, "From 'ban it till we understand it' to 'resistance is futile': How university programming instructors plan to adapt as more students use AI code generation and explanation tools such as chatgpt and github copilot," in *Proceedings of the 2023 ACM Conference on International Computing Education Research* V.1. New York, NY, USA: ACM, 2023, pp. 106–121. https://doi.org/10.1145/3568813.3600138

[2] M. Kazemitabaar, L. Hou, A. Z. Henley, B. J. Ericson, P. Denny, and T. Grossman, "Studying the effect of AI code generators on supporting novice learners in introductory programming," in *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: ACM, 2023, pp. 2877–2893. https://doi.org/10.1145/3544548.3580919

[3] R. Azevedo and D. Gasevic, "Analyzing multimodal multichannel data about self-regulated learning with advanced learning technologies: Issues and challenges," *Computers in Human Behavior*, vol. 96, pp. 207–210, 2019. https://doi.org/10.1016/j.chb.2019.03.025

[4] M. Siadaty, D. Gasevic, and M. Hatala, "Trace-based micro-analytic measurement of self-regulated learning processes," *Journal of Learning Analytics*, vol. 3, no. 1, pp. 183–214, 2016. https://doi.org/10.18608/jla.2016.31.11

[5] K. R. Koedinger and V. Aleven, "Exploring the assistance dilemma in experiments with cognitive tutors," *Educational Psychology Review*, vol. 19, no. 3, pp. 239–264, 2007.

[6] R. S. Newman, "Adaptive help seeking: A strategy of self-regulated learning," in *Self-Regulation of Learning and Performance: Issues and Educational Applications*, D. H. Schunk and B. J. Zimmerman, Eds., Hillsdale, NJ: Lawrence Erlbaum, 1994, pp. 283–301.

[7] B. J. Zimmerman, "Becoming a self-regulated learner: An overview," *Theory Into Practice*, vol. 41, no. 2, pp. 64–70, 2002.

[8] J. Sweller, P. Ayres, and S. Kalyuga, *Cognitive Load Theory*. New York, NY: Springer, 2011. https://doi.org/10.1007/978-1-4419-8126-4

[9] A. Collins, J. S. Brown, and S. E. Newman, "Cognitive apprenticeship: Teaching the craft of reading, writing and mathematics," in *Knowing, Learning, and Instruction: Essays in Honor of Robert Glaser*, L. B. Resnick, Ed., Hillsdale, NJ: Lawrence Erlbaum, 1989, pp. 453–494.

[10] D. Wood, J. S. Bruner, and G. Ross, "The role of tutoring in problem solving," *Journal of Child Psychology and Psychiatry*, vol. 17, no. 2, pp. 89–100, 1976.

[11] E. L. Deci, R. J. Vallerand, L. G. Pelletier, and R. M. Ryan, "Motivation and education: The self-determination perspective," *Educational Psychologist*, vol. 26, nos. 3–4, pp. 325–346, 1991.

[12] R. M. Ryan and E. L. Deci, "Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being," *American Psychologist*, vol. 55, no. 1, pp. 68–78, 2000.

[13] C. P. Niemiec and R. M. Ryan, "Autonomy, competence, and relatedness in the classroom: Applying self-determination theory to educational practice," *Theory and Research in Education*, vol. 7, no. 2, pp. 133–144, 2009.

[14] J. R. Landis and G. G. Koch, "The measurement of observer agreement for categorical data," *Biometrics*, vol. 33, no. 1, pp. 159–174, 1977.

[15] M. L. McHugh, "Interrater reliability: The kappa statistic," *Biochemia Medica*, vol. 22, no. 3, pp. 276–282, 2012. https://doi.org/10.11613/BM.2012.031

[16] T. K. Koo and M. Y. Li, "A guideline of selecting and reporting intraclass correlation coefficients for reliability research," *Journal of Chiropractic Medicine*, vol. 15, no. 2, pp. 155–163, 2016.

[17] J. Ryoo, M. P.-C. Lin, S. Rai, W. He, S. M. Park, and M. Ho, "WIP: Multi-agent artificial intelligence model to enhance self-regulated learning and conceptual understanding in computer science education," in *2025 IEEE Frontiers in Education Conference (FIE)*, Nashville, TN, USA, 2025, pp. 1–5. https://doi.org/10.1109/FIE63693.2025.11328198

[18] M. P.-C. Lin and D. Chang, "CHAT-ACTS: A pedagogical framework for personalized chatbot to enhance active learning and self-regulated learning," *Comput. Educ. Artif. Intell.*, vol. 5, p. 100167, 2023. https://doi.org/10.1016/j.caeai.2023.100167

[19] D. H. Chang, M. P.-C. Lin, S. Hajian, and Q. Q. Wang, "Educational design principles of using AI chatbot that supports self-regulated learning in education: Goal setting, feedback, and personalization," *Sustainability*, vol. 15, no. 17, p. 12921, 2023. https://doi.org/10.3390/su151712921

[20] S. M. Park, M. Ho, M. P. C. Lin, and J. Ryoo, "Evaluating the impact of assistive AI tools on learning outcomes and ethical considerations in programming education," in *2025 IEEE Global Engineering Education Conference (EDUCON)*, IEEE, 2025, pp. 1–10. https://doi.org/10.1109/EDUCON62633.2025.11016517

[21] K. Keahey *et al.*, "Lessons learned from the Chameleon testbed," in *Proceedings of the 2020 USENIX Annual Technical Conference (USENIX ATC '20)*, 2020, pp. 219–233. https://www.usenix.org/system/files/atc20-keahey.pdf

11 AUTHORS

Dr. Michael Pin-Chuan Lin is an Assistant Professor in the Faculty of Education at Mount Saint Vincent University, Canada. His research examines how generative AI, self-regulated learning, learning analytics, and AI-supported feedback can improve teaching and learning in K–12 and higher education. He has published on chatbot-assisted learning, online and blended learning, and the ethical and pedagogical implications of AI in education. His current work focuses on open-source large language models, human–AI collaboration, and the responsible integration of AI in education (E-mail: michael.lin@msvu.ca).

Dr. Daniel H. Chang is an Assistant Professor at Athabasca University and a lecturer in the Faculty of Education at Simon Fraser University. His teaching and research focus on the ethical and pedagogical integration of artificial intelligence in postsecondary education, with particular interest in supporting learners' self-regulated learning, metacognition, and inclusive teaching practices (E-mail: dchang@athabascau.ca).

Dr. Vasudevan Janarthanan is an Associate Professor of Computer Science at Fairleigh Dickinson University, Vancouver. He received his PhD in real-time embedded systems from Concordia University in 2007. His current research interests are learning-based query optimization, automated database configuration, and generative AI for database design. He has previously published in areas related to real-time scheduling and supervisory control of real-time systems.

Dr. Michael S. Hsiao is a Professor in the Department of Electrical and Computer Engineering at Virginia Tech. His research interests include design automation, natural language based design and artificial intelligence. He has developed a natural-language-based design platform, Game Changineer, that assists students and teachers design video games in natural language. This platform has been recognized

as State Winner by the Virginia Cooperative Extension Educational Technology in 2022. He is a Fellow of IEEE (E-mail: hsiao@vt.edu).

Dr. Marco Ho is a faculty of School of Computing and Academic Studies at the British Columbia Institute of Technology, Canada, and currently serves as the Option Head of Digital Processing. His current research interests include engineering education, computing pedagogy, digital transformation in education, embedded systems, and applied machine intelligence (E-mail: marco_ho@bcit.ca).

Dr. Jeeho Ryoo is an Assistant Professor in the Olsen College of Engineering and Science at Fairleigh Dickinson University, where he leads the Computer Systems Laboratory at FDU-Vancouver. His research focuses on computer architecture, memory systems, and the use of generative AI in computer science education (E-mail: j.ryoo@fdu.edu).