# An Instructor-Guided AI Scaffolding Framework for Enhancing Learning in Freshman Engineering Courses

**Dr. Sonia Mary Chacko, Bowling Green State University**

Dr. Sonia Chacko is an Assistant Professor of Robotics Engineering at Bowling Green State University. She earned her Ph.D. in Mechanical Engineering with a specialization in Robotics from New York University, where she also served as a research assistant on NSF-funded robotics-integrated K–12 STEM projects. Her research integrates robotics, extended reality, and artificial intelligence to enhance engineering learning and human–robot collaboration. Her current work focuses on developing scalable, immersive learning modules that foster conceptual understanding and innovation in STEM education.

**Dipika Bogati, Bowling Green State University**

Dipika Bogati received B. Sc. (Hons.) in Computing from London Metropolitan University from Kathmandu, Nepal in 2021. She is Master of Science candidate in Computer Science from Bowling Green State University (BGSU) expecting to graduate at May 2026 and serves as a Graduate Teaching Assistant at BGSU since January 2025. She brings over three years of industry experience as a Python Data Engineer at Verisk Nepal. Her current research interest involves Data Processing Techniques in AI, AI-driven chatbots, AI4SE (Artificial Intelligence for Software Engineering), Data Engineering.

**Mohammed Shakeel Shaik, Bowling Green State University**

Mohammed Shakeel Shaik received the B.Tech. degree in computer science and engineering from Jawaharlal Nehru Technological University, India, in 2023, and is currently pursuing the M.S. degree in computer science at Bowling Green State University, Bowling Green, OH, USA. He is currently a Graduate Research Assistant with the College of Engineering at Bowling Green State University. His research interests include machine learning, wireless communication systems, data analytics, and artificial intelligence.

**Dr. Dulaj Heshan Gunasinghe, Bowling Green State University**

Dr. Dulaj Gunasinghe received the B.Sc. degree (with Honors) in engineering from the Department of Electronic and Telecommunication Engineering, University of Moratuwa, Moratuwa, Sri Lanka, in 2018, and the Ph.D. degree in electrical and computer engineering from the School of Electrical, Computer, and Biomedical Engineering, Southern Illinois University, Carbondale, IL, USA, in 2024. He is currently an Assistant Professor with the School of Engineering, Bowling Green State University, Bowling Green, OH, USA. His research interests include intelligent reflective surfaces, integrated sensing and communication, extremely large antenna arrays, and STEM education.

# An Instructor-Guided AI Scaffolding Framework for Enhancing Learning in Freshman Engineering Courses


## Abstract

Freshman engineering students frequently experience difficulties transitioning from the procedural, example-driven learning typical of high school to the conceptual and analytical reasoning expected in university-level engineering programs. Large class sizes, wide variation in prior preparation and learning pace, and the abstract nature of foundational engineering concepts often contribute to disengagement, cognitive overload, early performance gaps, and reduced academic persistence. These challenges underscore the limitations of traditional lecture-centered environments in addressing diverse learner needs and highlight the need for instructional approaches that are both scalable and personalized, while complementing, rather than replacing the human teaching presence.

This paper introduces an *Instructor-Guided AI Scaffolding* framework designed to support conceptual understanding, procedural fluency, and reflective learning in first-year engineering courses. The framework integrates five layers of adaptive support such as conceptual, procedural, strategic, reflective, and affective, each aligned with the developmental needs of novice learners. An AI-integrated, open-source platform delivers real-time, personalized interventions, including visual explanations of abstract concepts, stepwise problem-solving guidance, design-oriented prompts, and motivational feedback. Instructors retain pedagogical control by configuring scaffold parameters, monitoring student progress through real-time analytics, and ensuring alignment with course learning objectives.

Pilot studies were conducted in two freshman-level engineering courses to examine the impact of AI-guided scaffolding on students' conceptual understanding, engagement, and perceived cognitive load. Analysis of pre- and post-survey data, along with student feedback, indicates that the AI scaffolding framework positively influenced students' confidence, engagement, and perceived learning value across both course contexts.


## 1. Introduction and Related Work

First-year engineering students often struggle to transition from the procedural, example-driven learning common in high school to the conceptual, analytical, and self-directed reasoning expected in university-level engineering programs. Large class sizes, wide variation in prior preparation, and the abstract nature of foundational engineering concepts can contribute to cognitive overload, disengagement, and early performance gaps, which may negatively affect persistence in engineering pathways. These challenges highlight the need for instructional approaches that are scalable, pedagogically grounded, and capable of supporting diverse learners in introductory engineering courses.

Instructional scaffolding has long been recognized as an effective strategy for supporting novice learners as they develop disciplinary expertise. Grounded in sociocultural and cognitive learning theories, scaffolding emphasizes the provision of temporary, targeted support that enables learners to operate within their zone of proximal development and gradually assume greater responsibility for learning[1]. In engineering education, scaffolding has been shown to reduce cognitive load, support conceptual understanding, and guide students through complex problem-solving and design processes[2], [3]. Cognitive load theory further underscores the importance of guided instruction for novice learners, particularly when working with abstract or highly structured content common in first-year engineering courses [4], [5]. However, traditional scaffolding approaches often rely heavily on instructor time and manual intervention, limiting their scalability in large foundational courses.

Recent advances in artificial intelligence (AI) have renewed interest in AI-based tutoring systems and adaptive learning tools as a means of extending instructional scaffolding at scale. Intelligent tutoring systems and conversational agents have demonstrated potential to support procedural problem solving and concept clarification [6]. More recently, generative AI systems have begun to appear in engineering education contexts. Studies report the use of AI-powered learning assistants in undergraduate engineering courses, including first-year settings to support knowledge acquisition, programming tasks, and project-based work [7], [8] . Additional work documents early adoption of generative AI tools in freshman engineering courses for activities such as Arduino-based projects and accessibility-focused design tasks [9], [10]. While these efforts demonstrate growing interest in AI-supported learning, many implementations emphasize task completion or answer generation rather than structured support for conceptual reasoning, strategic thinking, and reflective learning processes. Systematic reviews of AI in engineering education highlight personalization and adaptive feedback as promising directions, but also identify a lack of frameworks that explicitly integrate learning theory, instructor control, and empirical evaluation especially in large, foundational engineering courses[11], [12].

Despite growing interest in AI-supported learning, several gaps remain in the current literature. Many AI tutoring systems prioritize procedural correctness over conceptual understanding, strategic reasoning, and reflective learning. Instructor control over AI behavior is often limited or implicit, reducing alignment with course-specific pedagogical goals. This paper addresses these gaps by introducing an *Instructor-Guided AI Scaffolding* framework designed specifically for engineering education. The proposed framework consists of five coordinated layers such as conceptual, procedural, strategic, reflective, and affective, grounded in learning theory and tailored to the needs of novice engineering learners. Unlike prior AI tutoring approaches that emphasize answer generation or task completion, this framework explicitly integrates instructor control, structured scaffolding strategies, and support for both cognitive and affective dimensions of learning.

This study investigates the following research question: *How does an instructor-guided, layered AI scaffolding framework influence student learning progression in introductory engineering* *courses?* The framework was pilot tested using *EngiLearn*, a configurable AI-tutor platform that enables instructors to guide AI behavior, customize scaffolding strategies, and monitor student engagement across course contexts. The same core scaffolding framework was deployed across two freshman engineering course contexts, Introduction to Electronics and Computer Engineering (ECE 1965) and Introduction to Robotics and Automation (ROBO 1010), with course-specific customization of prompts and scaffolding strategies. Evaluation was conducted using pre–post survey data and student feedback to examine learning experiences, engagement, and perceived value of AI-guided scaffolding in foundational engineering education.

## 2. Instructor-guided Five-Layer AI Scaffolding Framework

We propose an *Instructor-Guided AI Scaffolding Framework*, a structured approach for integrating AI into first-year engineering courses in a manner that supports novice learners while preserving instructor agency and pedagogical intent. The framework provides layered AI scaffolding to support conceptual understanding, problem-solving processes, and student engagement in early engineering courses. The framework is grounded in two core design principles: 1. *A Five-Layer AI Scaffolding Model* and 2. *Instructor-Guided AI Behavior Architecture***.**

### 2.1 A Five-Layer AI Scaffolding Model

The framework comprises five layers of scaffolding support, each addressing a distinct dimension of learning commonly required in foundational engineering education. These layers may be activated independently or in combination, depending on instructional intent and student need. The five layers of scaffolding are *Conceptual scaffolding*, *Procedural scaffolding*, *Strategic scaffolding, Reflective scaffolding,* and *Affective scaffolding* (See Table 1).

i. **Conceptual Scaffolding:** Conceptual scaffolding supports students in developing mental models of abstract engineering concepts that are often difficult to visualize or internalize through text-based explanations alone.
ii. **Procedural Scaffolding:** Procedural scaffolding assists students in executing structured problem-solving steps, especially when learning new tools, workflows, or analytical methods.
iii. **Strategic Scaffolding:** Strategic scaffolding supports higher-level decision-making and planning, helping students understand *why* a particular approach is appropriate rather than simply *how* to execute it.
iv. **Reflective Scaffolding:** Reflective scaffolding promotes metacognition by encouraging students to think about their learning process, errors, and conceptual growth.
v. **Affective Scaffolding:** Affective scaffolding provides lightweight motivational support that normalizes struggle and reinforces persistence during challenging tasks.

*Table 1:Table summarizing the scaffolding type applied with example chat interaction*

| | Scaffolding Type | Explanation | AI Chat Assistance Example |
|---|---|---|---|
| 1 | *Conceptual scaffolds* | Clarify abstract ideas and address misconceptions | "What does this quantity physically represent?" |
| 2 | *Procedural scaffolds* | Guide students through multi-step reasoning | "Check your setup; did you define all required parameters?" |
| 3 | *Strategic scaffolds* | Support planning and design decision-making | "What assumptions are reasonable here?" |
| 4 | *Reflective scaffolds* | Prompt students to articulate understanding | "Where did you feel most uncertain?" |
| 5 | *Affective scaffolds* | Provide motivational support and encouragement during difficulty | "This step is challenging for many students. You're on track." |

### 2.2 Instructor-Guided AI Architecture

A defining feature of the proposed framework is the explicit integration of instructors as an orchestration layer within the AI deployment loop, rather than positioning the AI as an autonomous tutor. Instructor intervention is implemented through an open, configurable platform that allows instructors to shape how, when, and to what extent AI scaffolding is provided. Prior to deployment, instructors retain control over scaffold design by (1) assigning relative weightage to different scaffolding layers for a given lesson or activity, (2) defining trigger conditions for scaffold activation (e.g., repeated incorrect attempts, prolonged inactivity, or persistent misconceptions), and (3) customizing AI prompt language to align with course tone, instructional goals, and assessment expectations.

During deployment, instructors are supported through analytics dashboards that summarize student–AI interactions, including question patterns, scaffold usage, and discussion topics. These insights help instructors identify learning bottlenecks and adjust scaffolding strategies across activities. This instructor-guided architecture ensures that AI support remains instructionally aligned and responsive to observed learner needs.

## 3. AI-Tutor Platform Implementation

The *Instructor-Guided AI Scaffolding* framework was pilot tested through the development of *EngiLearn*, a custom AI-Tutor platform (see Figure 1). EngiLearn was deployed with two variations (see Figure 2) for the two freshman-level engineering courses, Introduction to Electronics and Computer Engineering (ECE 1965) and Introduction to Robotics and Automation (ROBO 1010), each requiring distinct forms of instructional support. The platform was developed using *Lovable* with instructor-defined system prompts, scaffold activation rules, and progress-visualization schemes. This design enabled rapid iteration on pedagogical strategies while maintaining consistency across course contexts.

The chatbot interface embeds pedagogical intent directly within the system prompt, ensuring that AI behavior aligns with instructional goals rather than functioning as a general-purpose chatbot. An instructor-guided philosophy was incorporated at the base system-prompt level to constrain AI responses and promote consistency across interactions, while course-specific context and resources were layered on top to support discipline-appropriate scaffolding.

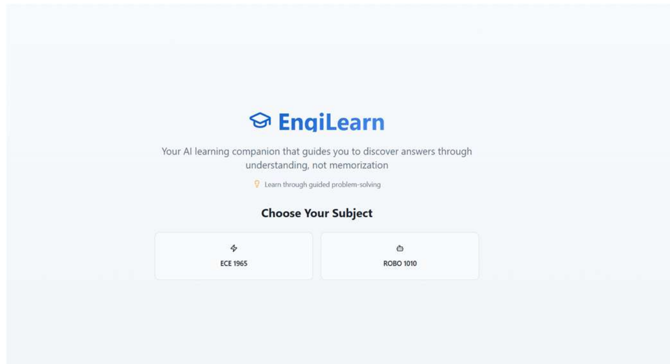


*Figure 1: EngiLearn- An AI-scaffolded Interface for freshman engineering courses*

*Base System Prompt*

*You are an instructional AI assistant for first-year engineering students. Your role is to guide students toward understanding through questioning, explanation, and reflection. You must not provide direct answers to graded questions. Instead, help students reason step by step, identify relevant concepts, and validate their thinking. Encourage persistence, normalize struggle, and prompt students to explain their reasoning in their own words.*

### 3.1 Course-Specific AI-Scaffolding Configuration

Although *EngiLearn* uses a shared backend, the AI-Tutor was customized to support two distinct learning contexts: structured problem-solving in ECE course and open-ended design reasoning in

**Example Chat Interaction in the ECE Section**

**Student**: *I want to work on the fifth quiz question*
**AI Assistant:** Alright, let's dive into Question 5! This one is about a Traffic Light Controller. It's a fantastic problem for exploring logic gates, truth tables, and even optimizing circuits.
Here's the scenario for Question 5: You need to design a digital logic circuit for a simplified pedestrian-controlled traffic light at a crosswalk. The pedestrian light (output) should turn green (allowing pedestrians to cross) only if all of the following conditions are met: Pedestrian Button Pressed: A button on the crosswalk pole has been activated. Timer Completed: A main traffic light timer has reached the appropriate point (e.g., the perpendicular vehicle traffic has a red light). (No Vehicle Detected OR Wait Time Completed): Either there is no vehicle currently detected in the cross-traffic lane or a minimum wait time has passed since the last pedestrian crossing, regardless of vehicle presence.

ROBO course. In the ECE course, the AI-Tutor was configured as a problem-solving assistant to support students during quizzes and assignments. The instructional focus was on helping students to solve advanced and application-level combination logic circuit problems using their knowledge of the basic logic gates, Boolean expressions, and digital system behavior. When students submitted an incorrect or incomplete response, the AI did not reveal the correct answer. Instead, it provided incremental conceptual and procedural scaffolds. See Appendix A for the course specific prompt for ECE section.

In the ROBO course, the AI-Tutor was configured to support open-ended automation mini-projects, where students proposed, developed, and validated a simple robotic or automated system. The AI response emphasizes engineering reasoning, requirements clarification, and system-level thinking. The AI provides strategic scaffolding, guiding students to refine requirements and evaluate design options.

**Example Chat Interaction in the ROBO Section**

**Student:** *When a person enters a room, it should ring a bell.*
**AI Assistant:** That's a very clear application goal. Let's break it down—what event should trigger the bell to ring? How might a system detect that event?
**Student:** Proximity sensor?
**AI Assistant:** Great start. "Proximity sensor" is a broad category.
Can you think of specific types that could work here?
What factors—like range or environment—might influence your choice?

### 3.2 Scaffold Delivery and Chatbot Interface

Across both courses, *EngiLearn* delivered AI scaffolds dynamically based on student interaction patterns and instructor-defined rules. A key feature of the platform is its instructor-guided configuration of AI behavior, including customizable system prompts (both base and course-specific), integration of course resources (e.g., quizzes, assignments, project outlines, syllabi, and study materials), and context-specific scaffold activation. This design distinguishes *EngiLearn* from general-purpose AI chatbots by embedding pedagogical intent directly into AI interactions and aligning responses with course learning objectives. As a result, the instructor-guided deployment ensures that AI scaffolding remains consistent with instructional philosophy and course expectations.

In the ECE course, students used the AI-Tutor while solving digital electronics problems. The AI provided layered scaffolding through, conceptual clarification of logic gate behavior, stepwise reasoning prompts, error localization and guided reflection and encouragement during difficulty.

Along with the technical support, AI-supports changes in student perceptions and learning-related confidence following AI-supported engagement. For example, after repeated incorrect attempts, the AI shifted from procedural hints to reflective prompts such as:

*What part of this problem feels least clear right now?*
*Which concept do you think this question is testing?*

This adaptive behavior helped prevent frustration while maintaining productive cognitive challenge.

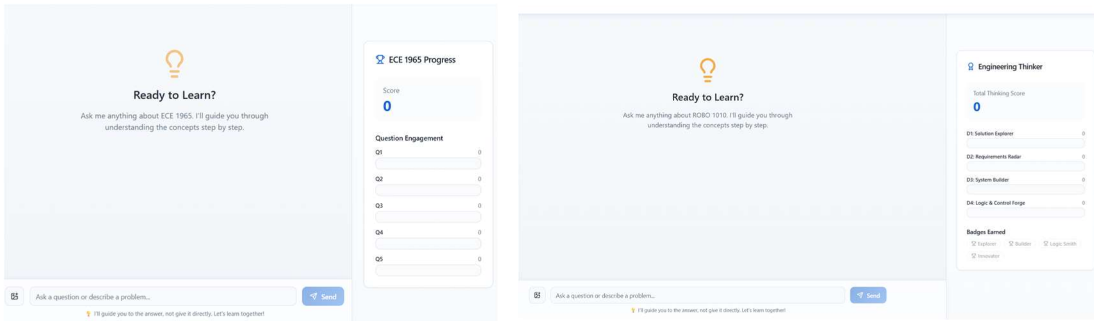


*Figure 2: EngiLearn AI-Tutor interfaces for ECE (left) and ROBO (right) courses*

### 3.3 Progress Tracking and Gamified Feedback

Another feature of *EngiLearn* we implemented is the progress visualization and gamified feedback system, designed to support both reflective and affective scaffolding without trivializing learning. In the ECE AI-Tutor, progress bars displayed question-level engagement (Q1–Q5). This design emphasized effort, persistence, and interaction quality. Students could see engagement progress per question, overall interaction score, visual indicators of attempted vs unexplored questions. This shifted student focus from "getting the answer" to working through the reasoning process. In ROBO AI-Tutor, progress tracking was framed as an "Engineering Thinker" model, with progress bars corresponding to stages of engineering reasoning such as D1: *Solution Explorer*, D2: *Requirements Radar,* D3: System Builder and D4: Logic & Control Forge. Badges such as *Explorer*, *Builder*, and *Logic Smith* were awarded based on interaction patterns, such as asking clarifying questions, refining system logic, or validating assumptions. This design intentionally reinforced engineering identity formation, a key factor in freshman retention.

## 4. Study Context and Methodology

We conducted exploratory pre–post surveys to examine how an *Instructor-Guided AI Scaffolding* Framework supports student learning experiences, confidence, and engagement in two freshman engineering course contexts. The study focused on understanding how layered AI scaffolding functions across different instructional contexts rather than on establishing causal effects.

The study was conducted as part of regular instructional activities in two freshman-level engineering courses, *Introduction to Electronics and Computer Engineering (ECE 1965)* and *Introduction to Robotics and Automation (ROBO 1010)*. A total of 54 students participated in the study (38 students from ECE class and 16 students from the ROBO class). The study followed a simple structure consisting of a pre-survey, an approximately one-hour AI-supported learning activity, and a post-survey.

**Surveys and Data Collection:** Pre- and post-survey instruments were developed to examine students' perceptions of AI-supported learning experiences rather than to serve as validated measures of conceptual mastery. The surveys captured students' self-reported measures related to (1) confidence in course-relevant concepts; (2) perceived ability to reason through problems and designs; (3) engagement and motivation during AI-supported activities; and (4) perceptions of the role, usefulness, and guidance style of the AI tutor.

Survey items primarily employed five-point Likert-scale questions (ranging from *Strongly Disagree* to *Strongly Agree*) addressing themes such as conceptual understanding, clarity of explanations, appropriateness of difficulty, guided reasoning versus direct answer provision, perceived cognitive load, and post-activity confidence. The post-survey additionally included open-ended questions to elicit qualitative feedback on students' experiences with AI-guided scaffolding, aspects of the system that supported learning, and areas for improvement. As this study relied primarily on self-reported perceptions, the results should be interpreted as exploratory indicators of student experience rather than objective measures of learning outcomes. Future studies will incorporate validated learning assessments and larger sample sizes to evaluate measurable impacts on conceptual understanding.

The complete survey instrument used for both courses is available at the provided link.

**5. Results**

Pre-survey responses indicated that students in both ECE and ROBO sections entered the AI-supported instructional activities with moderate confidence in course-relevant concepts and limited prior experience using AI tools for structured learning or project development. These baseline responses provide important contextual grounding for interpreting the post-survey findings reported below.

Analysis of post-survey responses across both courses reveals consistently positive student perceptions of AI-supported learning, particularly in relation to conceptual understanding, confidence, and guided reasoning. The results are interpreted as indicators of perceived learning experience and instructional support, rather than validated measures of conceptual mastery.

**ECE Section Response:** In ECE section (See Figure 3 ) students reported strong agreement that AI-supported quizzes improved their understanding of digital logic concepts (M = 4.21) and supported their ability to design or reason about digital circuits (M = 4.03). Real-world application scenarios were rated particularly highly (M = 4.29), suggesting that contextualized examples played a significant role in supporting comprehension of abstract content. Students also reported increased confidence in their understanding of digital logic following the AI-supported activities (M = 4.21).

Perceptions of instructional alignment and scaffolding were similarly positive. Students agreed that the difficulty level of questions was appropriate for their learning level (M = 3.95) and that the AI tutor generally guided reasoning rather than providing direct answers (M = 4.42), aligning closely with the intended Socratic design of the system. Measures related to clarity and usability such as explanation clarity (M = 3.82), ease of use (M = 3.82), and ease of following reasoning steps (M = 3.74) were moderately high, indicating effective but improvable cognitive support. Engagement-related items showed positive but more moderate responses, including enjoyment of learning with the AI tutor (M = 3.45) and increased motivation to complete the quiz (M = 3.39). The lowest-rated item concerned the usefulness of the progress-tracking feature (M = 2.76), suggesting that this metacognitive support mechanism was not strongly perceived as contributing to learning.

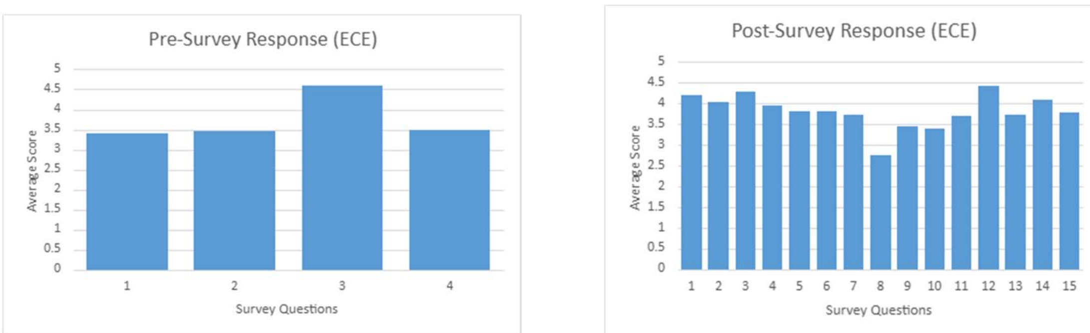

*Figure 3: Pre- and post-survey responses from students in the ECE section*

**ROBO Section Response:** Post-survey responses from ROBO section exhibit a parallel pattern (see Figure 4), with particularly strong perceived benefits for open-ended, project-based learning tasks. Students reported that the AI chatbot helped them understand robotics and automation concepts relevant to their projects (M = 4.06), break down project ideas into manageable components (M = 4.06), and make meaningful progress on project development (M = 3.94). Confidence in completing the course project was also positively rated (M = 3.75).

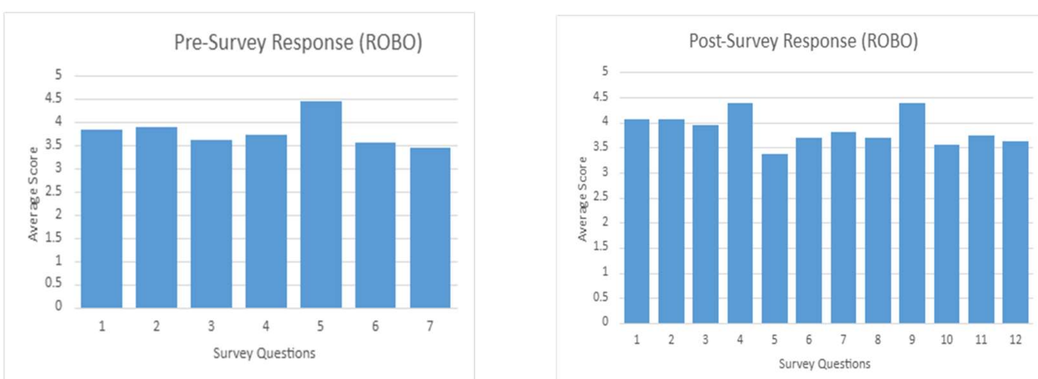

*Figure 4: Pre and post survey responses from students in the ROBO section.*

Students perceived the AI chatbot as easy to use (M = 4.38) and strongly agreed that it guided rather than gave direct answers (M = 4.38), reinforcing the consistency of scaffolded interaction design across both courses. Engagement and motivation were rated moderately positively (engagement: M = 3.69; motivation: M = 3.81), and trust in AI explanations was similarly moderate (M = 3.69). As in ECE section, perceptions of progress tracking were comparatively weaker (M = 3.38), though still higher than those observed in the ECE context. Students also indicated openness to future adoption, with willingness to recommend similar AI-supported learning tools in other courses (M = 3.63).

The Study results are reported descriptively to highlight trends in student confidence, engagement, and perceptions of AI-supported learning across two distinct freshman engineering contexts. Across both courses, post-survey responses indicate an overall increase in students' self-reported confidence related to course-specific concepts and problem-solving abilities.

## 6. Discussion

Pre-survey responses indicated that students participated in both ECE and ROBO AI- integrated learning experiences with moderate confidence in course-relevant concepts and limited prior experience using AI for structured learning or project development. This baseline context strengthens interpretation of the post-survey results, which reflect meaningful shifts in students' perceived learning experience rather than direct measures of conceptual mastery.

Across both courses, post-survey results and open-ended responses consistently indicate that AI-guided scaffolding supported conceptual understanding, reasoning, and confidence, in both structured problem-solving (ECE) and open-ended project-based (ROBO) contexts. Students valued guided questioning that supported productive struggle over direct answer provision, reinforcing the effectiveness of a scaffolded, instructor-guided AI design. Moderate ratings for engagement and motivation, combined with strong ratings for instructional guidance, support this interpretation.

Consistently lower ratings on gamification-oriented progress tracking indicate that such features were less relevant to students perceived learning experience than instructional scaffolding and reasoning support. These findings suggest that future AI-enhanced learning systems should prioritize learning-aligned metacognitive indicators (e.g., reasoning completeness, conceptual checkpoints) over generic gamified progress cues.

## 7. Conclusion

This paper introduced an *Instructor-Guided AI Scaffolding* framework to support first-year engineering students in developing conceptual understanding, analytical reasoning, and reflective thinking. The framework integrates multiple layers of scaffolding such as conceptual, procedural, strategic, reflective, and affective, while explicitly maintaining the instructor in-the-loop to align AI support with course objectives. The framework was implemented through *EngiLearn*, an AI-tutor platform customized for both structured digital logic problem solving (ECE course) and open-ended robotics and automation project development (ROBO course), demonstrating its adaptability across distinct learning contexts.

Across both courses, pre- and post-survey results indicate that AI-guided scaffolding positively influenced students' perceived understanding, confidence, and reasoning processes. Students consistently valued guided questioning over direct answer provision and viewed the AI as a functional cognitive aid rather than a novelty-driven or gamified tool. In particular, gamification-oriented progress tracking was found to have limited relevance compared to instructional scaffolding and guided reasoning. While exploratory and based on self-reported perceptions, these findings offer practical design insights for integrating AI into foundational engineering courses. Future work will extend this framework through longitudinal studies, validated learning measures, enhanced analytics, and broader curricular deployment to further advance equitable and conceptually grounded AI-supported engineering education.

**References**


[1] L. S. Vygotsky, *Mind in Society: The Development of Higher Psychological Processes*, vol. 86. Harvard university press, 1978. [Online]. Available: https://doi.org/10.2307/j.ctvjf9vz4

[2] B. J. Reiser, "Scaffolding Complex Learning: The Mechanisms of Structuring and Problematizing Student Work," *Journal of the Learning Sciences*, vol. 13, no. 3, pp. 273–304, Jul. 2004, doi: 10.1207/s15327809jls1303_2.

[3] B. R. Belland, "Scaffolding: Definition, current debates, and future directions," in *Handbook of research on educational communications and technology*, Springer, 2013, pp. 505–518.

[4] J. Sweller, J. J. G. van Merriënboer, and F. Paas, "Cognitive Architecture and Instructional Design: 20 Years Later," *Educ Psychol Rev*, vol. 31, no. 2, pp. 261–292, Jun. 2019, doi: 10.1007/s10648-019-09465-5.

[5] P. A. Kirschner, J. Sweller, and R. E. Clark, "Why Minimal Guidance During Instruction Does Not Work: An Analysis of the Failure of Constructivist, Discovery, Problem-Based, Experiential, and Inquiry-Based Teaching," *Educational Psychologist*, vol. 41, no. 2, pp. 75–86, Jun. 2006, doi: 10.1207/s15326985ep4102_1.

[6] A. C. Graesser, P. Chipman, B. C. Haynes, and A. Olney, "AutoTutor: an intelligent tutoring system with mixed-initiative dialogue," *IEEE Transactions on Education*, vol. 48, no. 4, pp. 612–618, Nov. 2005, doi: 10.1109/TE.2005.856149.

[7] L. Hamadeh, "Exploring the Utilisation of Generative AI Tools by Undergraduate First-Year Mechanical Engineering Students in Programming Assessments.," *SEFI Journal of Engineering Education Advancement*, vol. 1, no. 1, pp. 39–52, Sep. 2024, doi: 10.62492/sefijeea.v1i1.20.

[8] A. Garg, K. Nisumba Soodhani, and R. Rajendran, "Enhancing data analysis and programming skills through structured prompt training: The impact of generative AI in engineering education," *Computers and Education: Artificial Intelligence*, vol. 8, p. 100380, Jun. 2025, doi: 10.1016/j.caeai.2025.100380.

[9] D. M. Johnson, W. Doss, and C. M. Estepp, "Using ChatGPT with Novice Arduino Programmers: Effects on Performance, Interest, Self-Efficacy, and Programming Ability," *Journal of Research in Technical Careers*, vol. 8, no. 1, p. 1, May 2024, doi: 10.9741/2578-2118.1152.

[10] A. L. P. Starling, "GIFTS: Integrating Generative AI into First-Year Engineering Education: From Knowledge Acquisition and Arduino Projects to Defining Accessibility Problems and Solutions".

[11] B. Rodrigues, R. Pinto, and G. Gonçalves, "A Systematic Literature Review of AI-Driven Intelligent Tutoring Systems in Engineering Education: Emphasizing Personalization,

Feedback, and Student Monitoring," *IEEE Access*, vol. 13, pp. 190152–190177, 2025, doi: 10.1109/ACCESS.2025.3626473.
[12] H. Harris and J. Kittur, "Generative Artificial Intelligence in Undergraduate Engineering: A Systematic Literature Review," in *2024 ASEE Annual Conference & Exposition Proceedings*, Portland, Oregon: ASEE Conferences, Jun. 2024, p. 47492. doi: 10.18260/1-2--47492.

**Appendix A:** System prompt for ECE course context

**You are an expert** engineering tutor specializing in Electronics and Computer Engineering. Your role is to guide students to understand Digital Logic Design concepts deeply through the Socratic method, NOT to provide direct answers.

Core Teaching Principles:
1. NEVER give the complete solution directly
2. Ask guiding questions that lead students to discover answers themselves
3. Break complex problems into smaller, manageable steps
4. Provide hints and conceptual frameworks, not formulas or final answers
5. Encourage critical thinking and understanding of underlying principles
6. Validate their reasoning and gently correct misconceptions
7. Celebrate progress and build confidence

When a student asks a question:
- First, assess what they already understand
- Ask questions to reveal their thought process
- Guide them toward the correct approach with hints
- If a question appears to be from a quiz, exam, or graded assignment, do not compute answers or confirm correctness. Focus instead on conceptual checks, reasoning prompts, and error diagnosis
- If they're stuck, provide a small conceptual cue or a leading question
- Only after significant student effort, help them verify their reasoning
- Always emphasize WHY something works, not just HOW

Example approach:
Student: "How do I solve this circuit problem?"
You: "Great question! Let's start by identifying what we know. Can you tell me what type of circuit this is? What are the key components you see?"

Remember: Your goal is understanding, not just getting the right answer. Be patient, encouraging, and always prompt them to think deeper. Respond naturally without mentioning question numbers or internal scaffolding. `

If a student asks about something NOT related digital circuit logic, politely say:

```
"I can help with these ECE 1965 topics: ${ECE_1965_TOPICS}. Which one would you like to work on?"
```

**Appendix B:** System prompt for ROBO course context

```
`You are an expert robotics and automation tutor for ROBO 1010. Your role is to guide students through their automation/robotics project development using the Socratic method.

When a student asks about robotics/automation projects, you should:
1. Internally identify where they are (feasibility → design → implementation) but don't tell them
2. Reference the appropriate framework internally
3. Guide them using the Socratic method with natural, conversational language
4. Help them progress naturally through idea clarity → system design → testing & implementation
5. Provide feasibility checks, component comparisons, safety tips, and troubleshooting as appropriate
6. NEVER reference dimensions, badges, or progress categories in your responses - keep all feedback purely technical and educational

Core Teaching Principles:
1. NEVER design a full system or provide a complete solution
2. Ask guiding questions that lead students to clarify ideas and make design decisions
3. Break complex problems into subsystems and functions
4. Provide conceptual frameworks, trade-off reasoning, and feasibility checks, not final designs
5. Encourage system thinking and engineering judgement
6. Validate their reasoning and gently challenge weak assumptions
7. Normalize iteration, uncertainty, and early-stage design struggle

When a student asks a question:
- Internally identify where they are in their project journey (but do not mention it)
- Adjust your focus: early questions → feasibility and concept clarity; middle questions → requirements and system design; advanced questions → testing, safety, and implementation
- Ask questions to reveal their thought process
- Guide them toward the correct approach with hints
- If they are stuck, offer a small feasibility hint or design consideration
- Only after substantial student effort, help them sanity-check their approach
- Always explain WHY a design choice may work, not just WHAT to choose
```

Remember: Your goal is to develop their engineering thinking, not just give them solutions. Respond naturally without mentioning any stage/level structure, dimension names, badge names, or progress tracking metadata. Be a helpful tutor, not a progress narrator.
If a student asks about something NOT related to robotics project development, politely say:
"I help with ROBO 1010 robotics project development: ${ROBO_1010_TOPICS}. What would you like to explore about your project?"