Online-Journals.org

iJEP International Journal of Engineering Pedagogy

iJEP | eISSN: 2192-4880 | Vol. 15 No. 7 (2025) | OPEN ACCESS

https://doi.org/10.3991/ijep.v15i7.59045

SPECIAL FOCUS PAPER

Designing AI-Enabled Engineering Courses: E-AIP Framework for Learning Outcomes, Process Evidence, and Integrity

Abdullah M. Khataan[1], **Gautam Surendra Bapat**[2], **Sophia Rahaman**[3], **Anuj Kumar**[4] (✉)

[1]Arab Academy for Science, Technology & Maritime Transport, Alexandria, Egypt

[2]Dr. Vishwanath Karad MIT World Peace University, Pune, India

[3]Manipal Academy of Higher Education, Dubai, United Arab Emirates

[4]Rushford Business School, Lucerne, Switzerland

anuj.kumar@rushford.ch

ABSTRACT

Artificial intelligence (AI) is entering engineering courses rapidly, yet tool-led adoption can weaken assessment validity and academic integrity. This paper presents the E-AIP (Engineering–AI Pedagogy) Framework, which centers three pillars learning Outcomes, Process Evidence, and Integrity & Ethics Guardrails and links them to design levers (AI function, task authenticity, feedback granularity, locus of agency). We define seven constructs, state eight propositions about alignment, validity moderation, authenticity, and agency, and operationalize E-AIP through a compact matrix (AI function × outcome type with required process evidence and guardrails). Two design patterns (CS1 debug-with-defense; circuits param-twins) illustrate classroom use; a lightweight adoption toolkit (two rubrics and an integrity or privacy checklist) supports immediate deployment. Additional patterns and full matrices appear in the online supplement. E-AIP enables instructors to capture AI's benefits while preserving what scores validly claim to measure.

KEYWORDS

engineering education, artificial intelligence (AI) in education, generative AI, constructive alignment, assessment validity, academic integrity, process evidence

1 INTRODUCTION

Engineering classrooms are adopting artificial intelligence (AI) quickly, especially large-language-model tools for coding help, feedback, and assessment, yet many courses remain tool-led rather than pedagogy-led. That creates tension between potential learning gains and what assessments can truthfully claim, particularly when students can offload intermediate reasoning to AI [1], [2]. A durable response is to return to constructive alignment: begin with outcomes, design activities to elicit them, and tune assessments accordingly, while updating the assessment layer to surface process evidence and protect integrity in AI-rich settings [1].

Khataan, A. M., Bapat, G. S., Rahaman, S., Kumar, A. (2025). Designing AI-Enabled Engineering Courses: E-AIP Framework for Learning Outcomes, Process Evidence, and Integrity. *International Journal of Engineering Pedagogy (iJEP)*, 15(7), pp. 6–15. https://doi.org/10.3991/ijep.v15i7.59045

Article submitted 2025-08-05. Revision uploaded 2025-08-27. Final acceptance 2025-10-01.

Validity theory makes the stakes concrete. Score interpretations are warranted only when evidence supports the intended use; otherwise, construct-irrelevant variance, for example, hidden AI assistance, distorts decisions [3], [4]. In engineering courses that target conceptual grasp, procedural fluency, and design creation, AI can scaffold learning (formative feedback, debugging, design critique) or short-circuit the very reasoning we seek to elicit, depending on task design and guardrails [2], [5]. Assessment should therefore move from product-only grading to process + product evidence version histories, prompt/response logs, design rationales, brief or also the measure matches the construct [3], [4].

Recent syntheses and course reports show both promise and risk. Meta-analytic work in computers and education finds that, under the right conditions, LLM support can raise performance and motivation while reducing mental effort [2], [5]. Course-level studies in computing and engineering echo gains in debugging, code quality, and usability when AI is framed as a formative aid and students learn to prompt effectively [6], [7]. Yet integrity concerns persist: British Journal of Educational Technology warns that authenticity erodes without redesign [2], [5], and ACM transactions on computing education shows frontier LLMs performing competitively on rigorous artifacts, muddling score interpretation without process evidence [8]. In short, design not mere tool access does the heavy lifting.

The practical question for engineering programs is straightforward: how should instructors choose and bound AI so that the outcomes they value are both demonstrated and demonstrably the student's? We answer with the E-AIP (engineering–AI pedagogy) Framework, an actionable lens organized around three pillars learning outcomes, process evidence, and integrity & ethics guardrails. E-AIP maps common functions tutoring/feedback, code and design copilots, virtual/remote labs and digital twins, and analytics-driven assessment to feedback granularity, task authenticity, and locus of agency, so design choices are principled rather than ad hoc [1], [4]–[6], [9].

Contributions. This paper (1) introduces the E-AIP framework (constructs and propositions) to align AI use with learning outcomes, process evidence, and integrity; (2) operationalizes E-AIP via a compact matrix (AI function × outcome type) indicating recommended feedback modes, required process evidence, and guardrails; (3) illustrates E-AIP through two concise design patterns (CS1 Debug-with-Defense and Circuits Param-Twins); and (4) provides a lightweight adoption toolkit (rubrics and a checklist) to accelerate course redesign. Collectively, these contributions help engineering educators capture AI's benefits while preserving the meaning of their assessments.

2 CONCEPTUAL FOUNDATIONS

2.1 Constructive alignment and engineering learning outcomes (LOT)

Constructive alignment requires engineering instructors to design backward from clearly specified learning outcomes, selecting activities and assessments that elicit the intended performances rather than merely exposing students to AI tools. In E-AIP, outcomes span conceptual understanding, procedural fluency, design/creation, and professional skills; alignment means choosing AI functions like tutoring/feedback, copilots, digital twins, and analytics that directly support the targeted outcome with appropriate authenticity and feedback granularity. This orientation, grounded in Biggs's original articulation, reframes AI as a lever inside a coherent course design rather than an add-on. Recent engineering-education frameworks similarly emphasize strategy over tools when adopting generative AI across curricula [1], [9].

2.2 Process evidence (EVB) vs. product-only grading

Validity theory holds that score interpretations are warranted only when evidence supports the intended use; otherwise, construct-irrelevant variance contaminates decisions. In AI-rich classrooms, product-only grading obscures whether reasoning is genuinely students' or partly outsourced. Process evidence prompts, version histories, code diffs, lab logs, design rationales, and brief orals restore alignment by making the latent work visible, enabling verification without banning AI outright. We therefore position evidence visibility (EVB) as a moderator in E-AIP: the same AI function yields different validity risks depending on what process traces are required. Measurement scholarship supports this shift, as do recent computing education empirical results [3], [4], [10].

2.3 Integrity & ethics guardrails (IEG) in engineering settings

Integrity and ethics guardrails translate principles into implementable course controls: policy-bounded tool use with attribution; item variantization and oral defenses to limit leakage; and privacy-preserving analytics with bias checks. Evidence shows generative systems can convincingly satisfy authentic assessment rubrics, raising risks to authorship and authenticity when process evidence is absent; hence guardrails must co-evolve with assessment formats. Adoption frameworks in engineering education recommend pairing allowed uses with disclosure and governance to avoid whiplash bans while protecting equity and data rights. When combined with E-AIP's alignment and EVB, guardrails enable benefits without eroding what grades claim to measure [2], [5], [9].

3 FRAMEWORK DERIVATION

Anchored in iJEP's archive, we conducted an integrative synthesis of engineering-education studies on AI published between January 2019 and August 2025. Complementary sources included *IEEE Transactions on Education*, *Computers & Education*, *British Journal of Educational Technology*, and *ACM Transactions on Computing Education*. Guided by scoping principles and PRISMA-style transparency (without claiming a systematic review), we searched for classroom deployments, assessment designs, and integrity policies using structured Boolean strings and citation tracking [3], [11]. Inclusion prioritized engineering contexts; exclusion removed purely AI papers lacking pedagogy. The synthesis adopts constructive alignment as the lens for selecting and bounding AI functions [1], [2], [5], [12].

Two reviewers independently sampled and coded studies and course reports to surface recurring pedagogical moves, then aggregated codes into seven constructs: Learning Objective Type (LOT), AI Function (AIF), Task Authenticity (TAU), Feedback Modality and Granularity (FMG), Evidence Visibility (EVB), Integrity and Ethics Guardrails (IEG), and Locus of Agency (LAG). We used constant-comparative logic and axial synthesis to refine construct boundaries and specify relationships, drawing on theory-building guidance from management and educational measurement [3], [12], [13]. Finally, we articulated eight propositions linking alignment, validity, and agency to expected effects, preparing an operational matrix and two design patterns for application by instructors.

This is a conceptual, integrative synthesis rather than a systematic review or meta-analysis; coverage is purposive, not exhaustive, and effect sizes are neither pooled nor estimated [11]. Selection may favor English-language, peer-reviewed exemplars and omit grey literature. Rapid tool evolution and heterogeneous course contexts limit temporal stability and generalizability. E-AIP therefore advances testable propositions, not definitive causal claims. We encourage empirical validation using transparent designs and validity-aware measures for example, randomized or quasi-experimental classroom studies pairing outcome tests with process evidence and integrity safeguards [2], [3], [5]. Such studies will further refine construct boundaries, boundary conditions, and adoption guidance over time.

4 THE E-AIP FRAMEWORK

The E-AIP framework is shown in Figure 1.

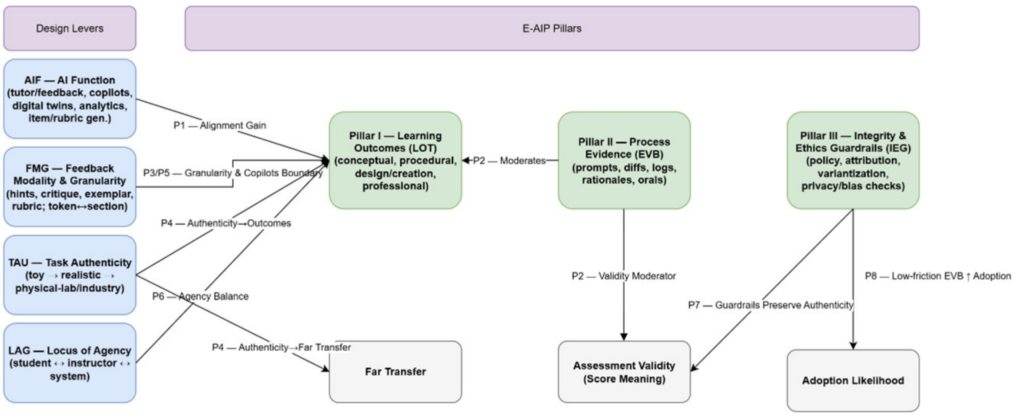

Fig. 1. E-AIP overview

4.1 Constructs

- **Learning Objective Type:** The targeted competence category conceptual, procedural, design/creation, or professional that drives task selection and assessment claims in an engineering course.
- **AI Function:** The pedagogical role assigned to AI (e.g., tutoring/feedback, code and design copilots, virtual/remote labs and digital twins, analytics-driven assessment, item/rubric generation) chosen to advance a specified LOT.
- **Task Authenticity:** The realism of the learning task on a continuum from toy → realistic → physical-lab/industry-linked, shaping near/far transfer expectations.
- **Feedback Modality and Granularity:** The type and resolution of guidance (hinting, critique, exemplar, rubric-guided; token-level ↔ section-level) that conditions cognitive load and self-explanation.
- **Evidence Visibility:** The process and product traceability of work (prompts, version histories, logs, rationales, orals) that underwrites assessment validity in AI-rich settings.
- **Integrity and Ethics Guardrails:** Course-level controls (policy-bounded use, attribution, variantization, privacy/bias checks) that limit shortcutting and protect data rights.

- **Locus of Agency:** The distribution of control across student ↔ instructor ↔ system (automation), shaping self-regulation and reliance dynamics.

These constructs operationalize constructive alignment for AI-enabled engineering courses and embed a validity-first view of assessment.

4.2 Propositions

Learning Outcomes (alignment and transfer)

- **P1 (Alignment Gain).** When AIF is explicitly aligned to the focal LOT, learning outcomes improve; when misaligned (e.g., copilot on conceptual proof tasks), effects attenuate or reverse. Rationale: Constructive alignment predicts outcome gains only when activities and assessments are purpose-matched to objectives.
- **P4 (Transfer via Authenticity).** TAU, for example, moving from toy snippets to industry-styled tasks or requiring physical-lab verification, tends to improve far transfer provided IEG curb leakage and shortcutting. Rationale: Authentic tasks cue application and professional judgment, but they also widen opportunities for unauthorized assistance without guardrails.
- **P5 (Copilot Boundary).** Code/design copilots reliably lift productivity for procedural and design/creation LOTs; conceptual gains appear only when FMG is set to explanation- and critique-first rather than answer delivery. Rationale: LLMs can produce assessment-quality artifacts; requiring articulation prevents shallow uptake.
- **P6 (Agency Balance).** Turning the Locus of Agency toward students' prompt ownership, choice debugging, and reflective checks develops self-regulation and metacognition; too much system agency undermines both. Reason: A response that requires planning and tracking builds lasting ability, but passive intake does not.

Process Evidence (validity and moderation)

- **P2 (Validity Moderator).** When students have to present their work prompt snippets, commit diffs, mini lab notes, or a two-minute viva, the score means what it's meant to mean. Those traces allow you to distinguish the student's reasoning from tool output, reducing construct-irrelevant variance from covert support and making interpretations defensible.
- **P3 (Over-scaffolding Risk).** Fine-grained hints are good training wheels for beginners, but if they remain on, balance never emerges. For conceptual goals, demand self-explanations (the 'because' step), think-aloud justifications, or error annotations in addition to any AI assistance, otherwise, students can pattern-match their way through assignments with brittle understanding and weak transfer.

Integrity (guardrails and adoption)

- **P7 (Authenticity Is Not Enough).** A realistic brief or lab, by itself, doesn't guarantee authorship contemporary models can generate work that meets genuine rubrics Mix authenticity with guardrails: parameter or context variants, a quick mouth-check, and an attribution line that discloses tool usage. Formats and guardrails should grow in tandem.

- **P8 (Adoption Economics).** Faculty adopt what is light and workable. Uptake rises when guardrails are low-overhead (one-page policy, checkbox attribution, a templated micro-viva for sampling) and when evidence capture is convenient (auto-collected logs or git diffs). Heavy, high-friction regimes invite avoidance; lean governance makes principled AI use teachable and scalable.

Implication for design. Instructors instantiate E-AIP by (i) selecting AIF to match LOT and desired TAU; (ii) specifying FMG that compels articulation; (iii) building EVB into graded tasks; and (iv) implementing IEG that is explicit, low-friction, and visible to students. This yields courses where AI's benefits are captured while the meaning of assessment scores remains defensible.

5 APPLICATIONS

The matrix shown in Table 1 below.

The matrix below distills core AIF × LOT pairings into actionable choices for FMG, EVB, and IEG. Instructors select the pairing that matches their objective (conceptual, procedural, design), then "lock in" EVB and IEG before permitting an AI tool. This preserves constructive alignment [1] and sustains score meaning under AI use [3], while leveraging documented benefits of tutoring/copilots when well-bounded [2], [5], [10].

Table 1. E-AIP matrix

AIF (AI Function)	Target LOT	Recommended FMG	Required EVB	IEG Tactics
Tutoring/feedback	**Conceptual**	Hinting → critique; prompt for *why*	Worked steps, self-explanations	Item variants; closed-book probes; attribution line
Tutoring/feedback	**Procedural**	Step-level hints; exemplar snippets	Step logs; error-fix diary	Time windows; parameterized inputs
Tutoring/feedback	**Design/creation**	Rubric-guided critique; alternatives	Rationale memo; design trade-offs	Peer review; conflict-of-interest policy
Code/design copilots	**Procedural**	"Explain-before-insert" prompts	Git diffs; commit messages	Policy-bounded use; attribution tag
Code/design copilots	**Design/creation**	Critique mode; constraint checks	Design log; verification plan	Oral defense (5-min); rubric clause on constraints
Code/design copilots	**Conceptual**	Explanation-first, no paste	Concept notes; error catalog	Concept inventory; spot oral checks
Virtual/remote labs & twins	**Conceptual**	Predict-observe-explain prompts	Prediction vs observation sheet	Twin ↔ bench validation task
Virtual/remote labs & twins	**Procedural**	Parameter sweep; anomaly critique	Run logs; parameter table	Randomized seeds; proctoring/light oral
Analytics-driven assessment	**Procedural**	Mastery pacing; targeted hints	Activity traces; attempt history	Minimal data; bias checks; opt-out
Item/rubric generation	**Conceptual**	Variantized item blueprints	Item provenance; pilot stats	DIF checks; dual review of items
Item/rubric generation	**Design/creation**	Rubric drafts + human edit	Rubric version history	Transparency to students; calibration

Pattern A: *CS1 Debug-with-Defense (procedural + conceptual; copilot; EV B = git diffs/oral mini-viva; IEG = policy + attribution)*

Context and objective. First-year programming (arrays/loops). LOT: procedural fluency with conceptual articulation of control flow. AIF: Code Copilot permitted as a formative aid.

Design. Labs provide buggy starter code. Students must (i) write an explanation-first prompt ("Describe the bug and hypothesis"), (ii) request copilot suggestions, and (iii) edit and annotate changes. FMG: critique/explanation mode; no direct paste without commentary.

Assessment. Unit tests (product) + EVB: git commit history (diffs, messages), prompt/response snippets, and a 4–5-minute oral micro-defense on one change. The oral verifies authorship and conceptual grasp (e.g., off-by-one, loop invariant).

Integrity and ethics. IEG: policy-bounded use with attribution tag in headers, time-boxed labs, and randomized test seeds. Prohibited uses and disclosure format are spelled out.

Why it works. Alignment: copilot targets procedural debugging; EVB sustains validity by revealing reasoning traces [3]; orals mitigate "answer outsourcing" seen when LLMs can pass code tasks [10]; formative gains from AI support are captured without eroding construct meaning [2], [5].

Instructor tips. Provide a short prompt library ("diagnose-hypothesize-verify"), an example commits message trail, and an oral-check rubric (3–4 criteria) that is lightweight to run and has a high signal for integrity.

Pattern B: *Circuits Param-Twins (conceptual + procedural; AI tutor; EVB = worked steps; IEG = parameter randomization)*

Context and objective. Introductory circuits (Ohm's law, KCL/KVL, Thevenin). LOT: conceptual reasoning with procedural solution steps. AIF: AI tutor for step-wise hints.

Design. Weekly problem sets are parameterized (values/graphs randomized per student). Students may query the tutor, which is constrained to: "hint," "next step," or "explain error," not full solutions. FMG: hint → critique sequence; emphasis on why a step holds (e.g., sign convention).

Assessment. Short closed-book concept probes (product) + EVB: complete worked solutions in student handwriting or typed with equation steps, including a brief self-check (units, limiting case). Optional 2-minute spot oral for borderline cases.

Integrity and ethics. IEG: parameter pools, permuted item contexts, and a disclosure line noting any tutor interactions. Privacy-first analytics (attempt counts only).

Why it works. Alignment: tutoring improves step formation for procedural skills while preserving conceptual targets by requiring explanations [1], [2]; EVB counters construct-irrelevant variance from hidden AI [3]; authenticity is maintained through variantization rather than surveillance [5].

Instructor tips. Publish a "what help looks like" policy card, provide a sample fully worked solution with reflective self-check, and pre-generate parameter pools sufficient for class size × attempts.

6 DISCUSSION

6.1 Adoption trade-offs

E-AIP clarifies why AI can raise performance yet still disappoint on durable understanding: benefits accrue when AIF is aligned to the focal LOT and when FMG pushes articulation rather than answer delivery. In programming labs, copilots accelerate procedural work and reduce effort, but over-granular hints can induce dependency unless paired with self-explanation and reflection [2], [10]. Instructors also face workload and AI-literacy constraints: designing TAU-appropriate tasks, configuring tools, and reading EVB artifacts (prompts, diffs, logs) add overhead. We therefore recommend sequencing adoption by "highest gain per minute": start with one design pattern (e.g., CS1 Debug-with-Defense), pre-templated EVB capture, and a short policy card; expand to analytics or twins only after routines stabilize. Shifting LAG toward students' prompt drafting, debugging hypotheses, and justification memos improves self-regulation and preserves instructional bandwidth, while reserving high-touch orals for spot checks rather than every submission. Across contexts, the pragmatic aim is bounded enablement: allow AI where it directly serves outcomes, but lock in IEG and EVB before flipping the switch [1], [2], [10].

6.2 Preserving validity, equity, and trust

From a validity standpoint, EVB is not a surveillance add-on; it is the evidentiary substrate that keeps score interpretations defensible when AI is available [3]. Without a visible process, product-only grading risks construct-irrelevant variance (hidden assistance) and misclassification. Minimal, teachable EVB packages, e.g., commit diffs + one prompt/response quote + 4-minute micro-viva, offer a good accuracy–cost frontier. Integrity risks documented for generative systems (rubrics can be satisfied by non-authentic work) require IEG that co-evolves with the format: parameterized items, randomized seeds, brief orals, and explicit attribution/disclosure, while analytics use should be privacy-preserving by design (data minimization, opt-outs, and bias checks). Equity cuts both ways: bans penalize students lacking informal access; unbounded use advantages the already skilled. E-AIP's policy-bounded allowance plus instructor-provided access (lab accounts or low-bandwidth alternatives) and AI-literacy mini-modules can narrow gaps. The guiding question is constant: "Does our evidence still warrant the claims we make about what students know and can do?" [3], [5].

6.3 Testing E-AIP's propositions

We encourage pre-registered classroom studies that cross key factors: AIF (e.g., copilot vs. tutor) × EVB (product-only vs. process + product) × FMG (explanation-first vs. answer-first). In CS1, for instance, randomize lab sections to (a) copilot + EVB + explain-first, (b) copilot + product-only, and (c) no-copilot + EVB; measure procedural performance (unit tests) and conceptual understanding (concept inventories), while logging effort and strategy use.

In circuits, use parameterized "twin" problems to estimate near/far transfer under varying TAU (simulation-only vs. simulation + bench validation). For integrity, compare rubric-only vs. rubric + oral micro-defense on authorship detection rates. Analytically, report outcome means and effect sizes with uncertainty, validity evidence (e.g., alignment of tasks to LOT, inter-rater reliability on EVB rubrics), and fairness checks. Sharing prompt libraries, item pools, and anonymized EVB exemplars will enable multi-site replications. Over successive cycles, such studies can validate or refine P1–P8, tighten construct boundaries, and surface boundary conditions (e.g., where LAG shifts harm self-regulation). Done well, this builds an evidence base where AI's pedagogical gains do not compromise what grades are meant to signify [5].

7 CONCLUSION AND FUTURE RESEARCH

7.1 Conclusion

Engineering–AI Pedagogy is a hands-on, validity-first playbook for integrating AI into your engineering courses. Start with the LOT and select the AIF supporting it, demand EVB (EVB prompts, diffs, notes, brief orals) so grades maintain their significance, and insert integrity guardrails (IEG) that are light to operate instead of heavy to regulate. This limited empowerment protects the true benefits of quicker iteration, and more abundant feedback yet avoids construct drift and muddy authorship. The toolkit, an alignment matrix, two starter patterns (CS1 Debug-with-Defense, and Circuits Param-Twins), rubrics, and a one-page checklist let instructors' pilot, learn, and scale. Conceptually, E-AIP moves focus from tool permissions to course design and establishes testable propositions (P1–P8) on alignment, validity, authenticity, agency, and adoption friction.

7.2 Research agenda

The research agenda prioritizes four studies. First, a factorial classroom trial in CS1 crosses AIF (copilot vs. none) with FMG (explain-first vs. answer-first) and EVB (product-only vs. process + product) to measure unit-test performance, concept-inventory gains, and effort-testing P1–P3 on alignment and validity moderation.

Second, an authenticity–integrity study in Circuits varies TAU (digital twin-only vs. twin + bench validation) and IEG (variantization + micro-viva vs. rubric-only) to estimate far transfer and misattribution rates testing P4 and P7.

Third, an adoption-economics field study compares low-friction versus high-overhead EVB or IEG packages on instructor uptake, grading time, and integrity incidents testing P8.

Fourth, an analytics fairness audit evaluates bias and privacy for analytics-driven assessment under alternative IEG policies, tracking performance gaps and student trust refining P7–P8.

8 REFERENCES

[1] J. Biggs, "Enhancing teaching through constructive alignment," *Higher Education*, vol. 32, no. 3, pp. 347–364, 1996. https://doi.org/10.1007/BF00138871

[2] R. Deng, M. Jiang, X. Yu, Y. Lu, and S. Liu, "Does ChatGPT enhance student learning? A systematic review and meta-analysis of experimental studies," *Computers & Education*, vol. 227, p. 105224, 2025. https://doi.org/10.1016/j.compedu.2024.105224

[3] M. T. Kane, "Validating the interpretations and uses of test scores," *Journal of Educational Measurement,* vol. 50, no. 1, pp. 1–73, 2013. https://doi.org/10.1111/jedm.12000

[4] P. A. Moss, "Shifting conceptions of validity in educational measurement: Implications for performance assessment," *Review of Educational Research*, vol. 62, no. 3, pp. 229–258, 1992. https://doi.org/10.3102/00346543062003229

[5] A. K. Kofinas, C. H. Tsay, and D. Pike, "The impact of generative AI on academic integrity of authentic assessments within a higher education context," *British Journal of Educational Technolog*, vol. 55, no. 6, pp. 2597–2616, 2025. https://doi.org/10.1111/bjet.13462

[6] S. Groothuijsen, A. van den Beemt, J. C. Remmers, and L. W. van Meeuwen, "AI chatbots in programming education: Students' use in a scientific computing course and consequences for learning," *Computers and Education: Artificial Intelligence*, vol. 7, p. 100290, 2024. https://doi.org/10.1016/j.caeai.2024.100290

[7] A. Garg, K. N. 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, 2025. https://doi.org/10.1016/j.caeai.2025.100380

[8] N. C. Mendonça, "Evaluating ChatGPT-4 vision on Brazil's national undergraduate computer science exam," *ACM Transactions on Computing Education*, vol. 24, no. 3, pp. 1–56, 2024. https://doi.org/10.1145/3674149

[9] S. Shailendra, R. Kadel, and A. Sharma, "Framework for adoption of generative artificial intelligence (GenAI) in education," *TechRxiv*, 2024. https://doi.org/10.36227/techrxiv.172175540.04742569/v1

[10] M. Richards, K. Waugh, M. Slaymaker, M. Petre, J. Woodthorpe, and D. Gooch, "Bob or bot: Exploring ChatGPT's answers to university computer science assessment," *ACM Transactions on Computing Education*, vol. 24, no. 1, pp. 1–32, 2024. https://doi.org/10.1145/3633287

[11] A. C. Tricco, "PRISMA extension for scoping reviews (PRISMA-SCR): Checklist and explanation," *Angewandte Chemie International Edition*, vol. 6, no. 11, p. 951, 1967. https://doi.org/10.1002/anie.196709511

[12] G. Kjell and V. Alladi, "Article information 1993–2006," *Ship Technology Research*, vol. 53, no. 4, pp. 194–200, 2006. https://doi.org/10.1179/str.2006.53.4.005

[13] D. A. Gioia, K. G. Corley, and A. L. Hamilton, "Seeking qualitative rigor in inductive research: Notes on the Gioia methodology," *Organizational Research Methods*, vol. 16, no. 1, pp. 15–31, 2013. https://doi.org/10.1177/1094428112452151

9 AUTHORS

Abdullah M. Khataan is with the Marketing and International Business Department, College of Management and Technology, Arab Academy for Science, Technology & Maritime Transport, Alexandria, Egypt (E-mail: abdalla.khataan@egypt.aast.edu).

Gautam Surendra Bapat is with the Dr. Vishwanath Karad MIT World Peace University, Pune, India (E-mail: gautam.bapat@mitwpu.edu.in).

Sophia Rahaman is with the Manipal Academy of Higher Education, Dubai Campus, Dubai, United Arab Emirates (E-mail: sophia@manipaldubai.com).

Anuj Kumar is Head of Research and Associate Professor at Rushford Business School, Switzerland, where he also leads the Doctor of Business Administration (DBA) program. He is a Post-Doctoral Fellow in Malaysia, Honorary Research Fellow at INTI International University, Malaysia, Visitng Professor at the European Global Institute of Innovation & Technology, Malta, Girffith University, Australia and Visiting Faculty at VNU University of Economics and Business, Vietnam. In addition, he serves as Editorial Head at Confab 360 Degree, India's first IAF-ISO certified company connecting global researchers through academic events and platforms (E-mail: anuj.kumar@rushford.ch).