Human Factors in Design, Engineering, and Computing, Vol. 238, 2026, 1661–1671
https://doi.org/10.54941/ahfe1008394

A High-Availability Multi-Agent RAG Architecture for Automated Expert Assistance and Multimodal Technical Onboarding

Javier Izquierdo-Reyes, Miguel de Jesús Ramírez-Cadena, and Olivia Calderón-Valderrama

School of Engineering and Sciences, Tecnológico de Monterrey, Mexico City, 14380, Mexico

ABSTRACT

This paper presents the design, deployment and evaluation of a high-availability, multi-agent Retrieval-Augmented Generation (RAG) architecture acting as an expert assistant for technical training. In sectors such as cyber-physical systems and automated instrumentation, onboarding depends on synchronous supervision by senior engineers, which creates bottlenecks and increases technical downtime. Retrieval is restricted to a closed, curated corpus of manuals, datasheets and approved guidelines, keeping answers traceable and reducing large language model hallucinations. The interaction layer runs on WhatsApp through the Evolution API and accepts text, documents, circuit images and voice notes; the backend combines n8n orchestration, a dedicated virtual private server and a Supabase PostgreSQL instance with the pgvector extension, with inference served by paid-tier models through Google Vertex AI with provider fallback. Retrieval scope is bound to the caller's resolved identity rather than chosen by the model, enforcing need-to-know separation, and the interaction policy is configurable by user profile: a Socratic mode that withholds finished solutions, and a direct mode for experienced practitioners. An adoption survey administered to final-semester engineering students at the end of an intensive training cycle ($n = 24$) obtained a global Cronbach's alpha of 0.90. Perceived competence ($M = 4.38$, $d = 1.93$) and autonomy ($M = 4.14$, $d = 1.39$) were significantly above the scale midpoint, whereas relatedness was not ($M = 3.24$, $p = 0.125$): the assistant substitutes for the informational function of mentorship but not for its social function. Industrial deployment remains a projected application.

Keywords: Retrieval-augmented generation, Multi-agent systems, Engineering education, Technical onboarding, Self-determination theory, Human-machine co-agency

INTRODUCTION

Technical onboarding in advanced manufacturing has become a limiting factor rather than a routine administrative step. Personnel joining a plant that runs cyber-physical systems or automated instrumentation must absorb procedural knowledge distributed across equipment manuals, internal standards, wiring diagrams and the undocumented experience of senior staff.

Received September 24, 2026; Revised October 25, 2026; Accepted November 2, 2026; Available online December 1, 2026

© 2026 The Authors. This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

That knowledge is normally transferred through synchronous mentorship, which is effective but does not scale: it consumes expensive engineering hours and turns a small number of experts into a queue. The same structure appears earlier in the professional pipeline, where final-year engineering students on instrumented design projects depend on a single instructor for the review cycles that drive their progress.

Large language models (LLMs) appear to relieve this bottleneck, and the literature documents their capacity to act as virtual tutors and to give immediate feedback (Caccavale et al., 2024). General-purpose assistants are nevertheless poorly suited to either setting. They generate plausible but incorrect statements, unacceptable when the answer determines how a circuit is wired or an instrument calibrated; they are not bound to the documentation an organisation requires, and apply no separation between what different users may see; and they encourage passive consumption of finished answers, a behaviour associated with reduced critical thinking (Atchley et al., 2024; Walter, 2024).

This paper describes a multi-agent RAG assistant restricted to a closed corpus of verified documents, deployed on an enterprise cloud stack, and governed by two policies that are configured rather than inferred: an interaction policy attached to the user profile, and a retrieval scope bound to the caller's resolved identity. The scope of the evidence should be stated plainly. The architecture targets industrial onboarding but has so far been evaluated only with engineering students, and that evaluation is a single end-of-cycle measurement rather than a controlled comparison. The contributions are therefore an architecture with an explicit governance mechanism, an interaction policy, adoption results from an academic pilot, and an analysis of what would have to change before plant-floor deployment.

RELATED WORK

Holmes and Tuomi (2022) warn that expectations of AI in education exceed what the technology delivers, partly because the pedagogical assumptions behind each system are rarely explicit. Davar et al. (2025) identify virtual tutoring and real-time feedback as the most consistent gains, with hallucination, data privacy and integration complexity as the main obstacles. He et al. (2024) report that model performance in engineering degrades on project-oriented tasks, precisely those that dominate capstone projects and industrial troubleshooting. Caccavale et al. (2024) restricted a tutor to a regulated document base to audit manufacturing-practice compliance. Recent work moves from single chatbots to coordinated agents: Ba et al. (2026) find across 46 studies that agents with clearly separated functions produce more consistent cognitive outcomes than undifferentiated assistants and and Dever et al. (2022) argue that pedagogical agents carry a learner model too thin to track self-regulation. On the retrieval side, Lewis et al. (2020) established grounding in an external corpus as the remedy for unsupported generation, and later surveys (Gao et al., 2023) treat corpus curation and retrieval scope as design variables rather than implementation details.

Self-determination theory (SDT) frames how users relate to such tools. Ryan and Deci (2000) define autonomy, competence and relatedness as the needs that sustain intrinsic motivation, and Peters et al. (2018) extend the framework to technology design, arguing that need satisfaction must be assessed separately at the interface, the task and the user's wider life, because a tool can be pleasant to operate while contributing nothing to the activity it supports. Li et al. (2025a) validated the AI Motivation Scale and Li et al. (2025b) showed with 1465 students that introjected regulation is central to the AI motivational system. Lan and Zhou (2025) conclude that the decisive variable is whether agency remains human-centred or shifts to the system, and Mina et al. (2021) document that self-regulation in engineering populations develops through management of environmental resources rather than isolated study. Against this, Atchley et al. (2024) and Walter (2024) argue that unreflective use removes the effort that produces durable learning. An assistant used for training must therefore be evaluated not only by the correctness of its answers but by whether the user can still diagnose a fault without it—a criterion that does not apply uniformly to every user.

SYSTEM ARCHITECTURE

The system is organised in four decoupled layers, so that a failure or upgrade in one does not interrupt the others (Figure 1). The goal is resilience: the interface must stay responsive while the backend is under load, and the knowledge base must be updatable without redeploying the channel. The stack described below is deployed and in current use.

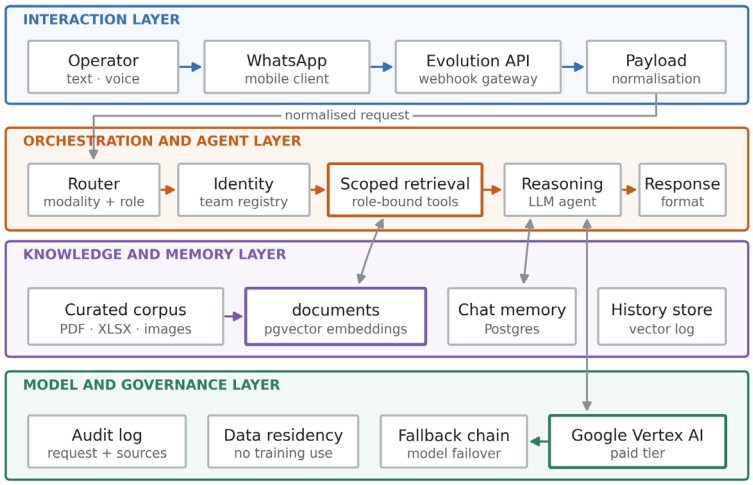


Figure 1: Layered architecture of the assistant. Solid arrows show the request path; double-headed arrows show service calls.

Not every function of the orchestration layer is an autonomous agent, a distinction that matters for reproducibility. Routing and identity resolution are deterministic workflow nodes; retrieval is exposed to the model as scoped tools whose filters it cannot alter; the interaction policy is carried by a single

generative agent; grounding is enforced by an instruction forbidding an answer before a retrieval call, with sources logged; and memory is written by the workflow. Reporting these as five independent agents would overstate the autonomy of the implementation.

Interaction Layer

The user-facing channel is WhatsApp, integrated through the Evolution API. The choice is pragmatic: users already carry the application, it requires no training, and it works where desktop access is limited. The channel accepts text, structured documents, diagnostic images such as circuit photographs, and voice notes, useful when the user's hands are occupied. Each message is normalised into a common representation before reaching the reasoning layer, so the agent receives a uniform payload regardless of modality. Group conversations and direct messages are distinguished here, because the two carry different entitlements.

Knowledge Base Construction

An ingestion workflow watches one storage folder per team plus a shared folder for official material, extracting, splitting, embedding and inserting each new file into a single pgvector table (Figure 2). Every chunk carries the identifier of the folder it came from, and that identifier is the mechanism on which the access rules below depend, so corpus curation and access control become the same operation. No external web content is retrieved at inference time, which makes every instruction traceable to an approved document (Lewis et al., 2020).

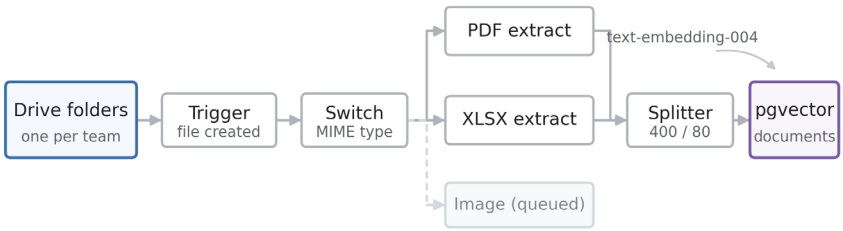


Figure 2: Ingestion pipeline. Every chunk is stored with its team_id, file_id and file name. Image ingestion is implemented but not enabled in the reported deployment.

Identity-Bound Retrieval Scope

Retrieval is not a single tool. The caller's messaging identifier is resolved against a registry table, and that resolved identity determines which retrieval tools the agent is given and what metadata filter each applies (Figure 3). Learners receive a tool restricted to the shared corpus and a tool restricted to their own team folder. Supervisors receive an unfiltered tool over team material and access to the interaction logs. The filter is written into the workflow, not into the prompt, so the model cannot widen its own scope

and a prompt-injection attempt inside an uploaded document cannot reach material the caller was not entitled to see.

This is the element most directly relevant to industrial transfer. Plant documentation is compartmentalised by line, by certification and by contractual obligation, and an assistant that cannot express those boundaries is not deployable regardless of how well it answers. Binding scope to identity at the workflow level converts a policy requirement into an enforced one.

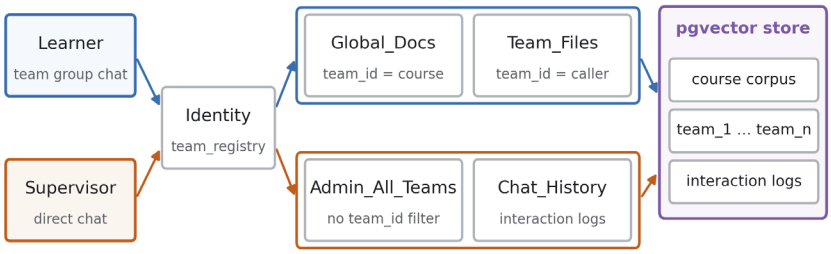


Figure 3: Retrieval scope bound to the resolved caller identity. Each role is given a different set of tools, and the metadata filter each tool applies is fixed by the workflow rather than chosen by the model.

Memory and Governance

Two stores serve different horizons. A relational conversation memory holds short-term context within a session; a vector store of past interactions, indexed by user and team, retrieves exchanges from previous weeks, so the assistant recognises a returning problem and supervisors obtain a searchable record of how a team's reasoning developed — the longitudinal learner model Dever et al. (2022) find missing in conventional pedagogical agents. Inference is served by paid-tier LLMs through Google Vertex AI, with fallback to a secondary provider so an incident degrades quality rather than causing an outage. The enterprise tier was selected for contractual guarantees that submitted data are not used for training, predictable latency and regional data residency; all requests and responses are logged with their retrieved sources (Table 1).

Table 1: Deployment parameters of the reported configuration.

Component	Configuration
Messaging channel	WhatsApp via Evolution API, webhook to the orchestrator
Orchestration	n8n on a dedicated virtual private server
Vector store	Supabase PostgreSQL with pgvector, single table, metadata filtering
Embeddings	768-dimension text embedding model
Chunking	400 tokens, 80-token overlap, recursive character splitter
Retrieved passages	15 for learner tools, 30 for supervisory tools
Inference	Paid-tier models via Google Vertex AI with secondary-provider fallback
Modalities	Text, PDF and spreadsheet documents, images, voice notes

INTERACTION POLICY: SOCRATIC AND DIRECT MODES

The interaction policy distinguishes this system from a documentation search engine, and is deliberately not the same for every user. The Socratic mode was applied in the pilot, whose users were students acquiring diagnostic competence rather than resolving incidents. The instruction set does not deliver complete solutions to problems the user is expected to solve. Dosed explanation supplies only the concept required for the immediate step; the guiding question narrows the search space, for example by asking which measurement would distinguish two candidate causes. A budget of at most two substantive observations per message prevents an exhaustive review the user could adopt without reading, and every response ends in a question that returns the decision to the user. Guiding questions support autonomy because the user retains control, and calibrated feedback supports competence, yielding the co-agency described by Lan and Zhou (2025).

The direct mode targets users who already hold the underlying competence. Withholding a solution from someone whose task is to restore a machine is not pedagogy but obstruction; here the assistant returns a concise procedural answer citing the manual, section and revision that supports it, and the metacognitive loop is replaced by acknowledgement of the referenced source. The grounding constraint operates identically in both modes; what changes is how much reasoning the system externalises. Mode assignment is a configuration attached to the user profile, not a runtime decision by the model. An escalation rule overrides the Socratic mode when a request involves a safety risk or falls outside the user's assigned scope, in which case the assistant answers directly and flags the exchange. Withholding a solution is a training decision, not a safety policy.

EVALUATION DESIGN

The evaluation was conducted with final-semester engineering students during an intensive training cycle between February and June 2026. Participants worked in teams on instrumented design projects requiring iterative hardware configuration, debugging and optimisation, a structure resembling the diagnostic work of industrial onboarding while remaining academic. Each team interacted with the assistant from its own group conversation, in Socratic mode throughout.

Adoption was measured with a survey administered once, at the end of the deployment. The instrument combines a 12-item Autonomy and Competence in Technology Adoption (ACTA) block, recording the regulation behind the decision to use the technology on a continuum from external to intrinsic, with a 39-item need-satisfaction block derived from the METUX framework (Peters et al., 2018), which measures competence, autonomy and relatedness separately at the interface, task and life spheres. All items use a five-point agreement scale; negatively worded items were reverse-coded. Twenty-four students completed the survey (mean age 21.8 years, range 21 to 26). Participation was voluntary and responses were analysed in aggregate.

Because the survey was administered once, the analysis tests whether need satisfaction differs from the scale midpoint of 3, using one-sample t-tests with Cohen's d as the effect size and Wilcoxon signed-rank tests as a non-parametric check; all contrasts agree across both. Relative autonomy

was summarised with the standard index RAI = 2(intrinsic) + identified – introjected – 2(external), positive when autonomous regulation dominates. The design supports claims about the level of need satisfaction reached, not causal claims, for which the paired control-group design of the original protocol remains mapped but not executed.

RESULTS

The global Cronbach's alpha was 0.90 across the 51 items and 0.92 for the 39 need-satisfaction items; the ACTA block alone reached 0.79. Subscale reliabilities were adequate for competence and autonomy at all three spheres (alpha 0.75 to 0.93) but weak for relatedness at the interface and task spheres (0.50 and 0.37), so those two means are reported with that caveat and support no conclusion on their own (Table 2).

Table 2: Need satisfaction and motivation (n = 24). Tests are against the scale midpoint of 3 for need satisfaction and against zero for the relative autonomy index.

Measure	Items	M	SD	alpha	d	p
Competence, interface	7	4.36	0.70	0.90	1.93	< 0.001
Competence, task	4	4.38	0.73	0.75	1.87	< 0.001
Competence, life	3	4.42	1.01	0.93	1.40	< 0.001
Autonomy, interface	5	4.21	0.76	0.79	1.59	< 0.001
Autonomy, task	4	3.99	1.01	0.86	0.98	< 0.001
Autonomy, life	4	4.21	1.00	0.93	1.21	< 0.001
Relatedness, interface	5	3.28	0.74	0.50	0.38	0.072
Relatedness, task	4	3.48	0.78	0.37	0.61	0.006
Relatedness, life	3	2.86	1.30	0.93	–0.11	0.605
Relative autonomy index	12	+3.12	2.17	0.79	1.44	< 0.001

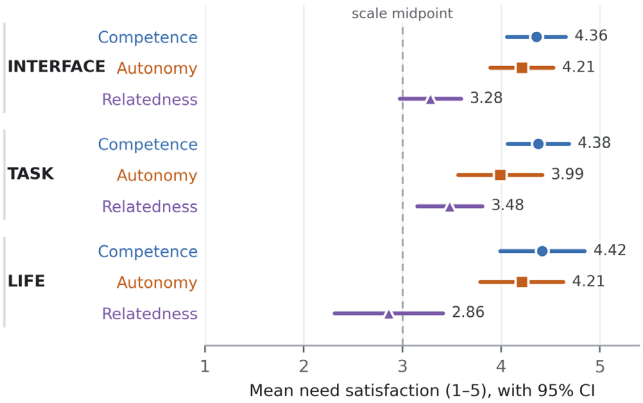


Figure 4: Need satisfaction by sphere, with 95% confidence intervals (n = 24).

The pattern is consistent and asymmetric (Figure 4). Perceived competence was high and homogeneous across spheres, from 4.36 at the interface to 4.42 at the level of the user's wider studies, far above the midpoint in every case.

Autonomy was also clearly satisfied, lowest at the task sphere ($M = 3.99$), where the Socratic constraint operates and friction is expected by design. Relatedness behaved differently: pooled, it did not differ from the midpoint ($M = 3.24$, $t(23) = 1.59$, $p = 0.125$), and at the life sphere it fell below it ($M = 2.86$).

That asymmetry is the most informative result in the dataset. The assistant succeeded at the two needs corresponding to the informational function of mentorship, making users feel capable and in control of their decisions, and did not substitute for its social function. Read against Ryan and Deci (2000), this bounds what the architecture can replace: resolving the queue of routine questions does not reproduce belonging to a team of practitioners, and the life-sphere value suggests heavy reliance on an individual channel may displace it. For industrial onboarding this argues for a blended arrangement in which the assistant absorbs routine queries and human mentorship is redirected to the relational and judgement-intensive parts of induction rather than eliminated.

Motivation for use was predominantly autonomous. Intrinsic ($M = 4.10$) and identified ($M = 4.06$) regulation exceeded introjected ($M = 2.58$) and external ($M = 3.32$), giving a relative autonomy index of +3.12 ($SD = 2.17$), significantly above zero, $t(23) = 7.03$, $p < 0.001$, $d = 1.44$. Students used the assistant because they valued it rather than because it was required, which contrasts with the centrality of introjected regulation reported by Li et al. (2025b).

Conversation logs confirmed the intended mechanism. Users did not receive finished solutions and had to diagnose their own errors. Iterative refinement was visible as successive design versions, V1 to V4, each revision addressing a fault the user identified through the assistant's guiding questions rather than through a corrected artefact supplied by the system. Queries concentrated on design refinement, component and bill-of-materials validation, and compliance with the evaluation criteria—the signature of co-agency rather than passive automation.

The pilot also produced a clear negative finding. The early open-source infrastructure could not sustain concurrent traffic peaks and suffered repeated instability when several teams interacted simultaneously, particularly close to deadlines. An assistant unavailable at the moment a user is blocked is equivalent to no assistant, and it erodes the trust adoption depends on. This motivated the enterprise architecture and provider fallback now in use.

PROJECTED TRANSFER TO INDUSTRIAL ONBOARDING

These results were obtained with students and do not demonstrate industrial effectiveness, but they indicate which parts of the design should transfer. The infrastructure is largely domain-independent: the messaging channel, the ingestion pipeline, the vector store and the identity-bound scope do not depend on whether the corpus holds laboratory manuals or plant documentation, and the compartmentalisation separating student teams is the mechanism that would separate production lines or certification levels. Four elements would change. The corpus would be replaced by official equipment manuals, calibration procedures and operating standards, with

the versioning discipline regulated environments require. The default mode would shift to direct for experienced personnel, reserving the Socratic mode for new hires. The governance layer would need to pass corporate data-protection review, which the enterprise tier supports but which has not been exercised in a plant. And grounding, now enforced by instruction, would become an explicit verification step whose block rate can be audited. The populations also differ: students work under assessment pressure and plant personnel under production pressure.

DISCUSSION

Three implications follow. First, correctness and pedagogy are separate problems. Restricting retrieval to a closed corpus improves traceability and reduces hallucination but does nothing to prevent overreliance: a system can be accurate and still produce users who cannot work without it. The Socratic constraint addresses that second risk and must be evaluated separately.

Second, the appropriate level of scaffolding is a property of the user, not of the tool. The same architecture supports opposite interaction policies, and treating the Socratic constraint as universally desirable would be a design error—as would treating retrieval scope as a property of the question rather than of the caller.

Third, infrastructure is a pedagogical variable. Research on AI-supported learning rarely treats uptime as a question, yet availability determines whether the interaction model can be enacted at all. Where users engage under external pressure rather than intrinsic interest (Li et al., 2025b), an unreliable tool is abandoned rather than tolerated.

LIMITATIONS AND FUTURE WORK

The results are preliminary and academic. The survey was administered once to 24 students from a single cohort, so the analysis establishes the level of need satisfaction reached but cannot attribute it to the intervention; the relatedness subscales at the interface and task spheres had weak internal consistency and need a better instrument. Grounding is enforced by instruction and logged, but no audit of citation support has yet been reported, so the hallucination claim rests on the design rather than on a measurement. Uploaded documents enter the same corpus the supervisory agent later reads, a data-poisoning and indirect prompt-injection surface (Greshake et al., 2023) that identity-bound scope narrows but does not close. No industrial deployment has been evaluated. Future work covers a controlled study with a larger cohort, a grounding audit using automated faithfulness measures (Es et al., 2024), longitudinal measurement of retention after the assistant is withdrawn, a comparison of the two modes with matched users, and a pilot with practising engineers.

CONCLUSION

This paper described a high-availability multi-agent RAG architecture intended to reduce the supervisory bottleneck in technical training without transferring diagnostic authority from the user to the machine. It combines

a multimodal messaging interface, workflow orchestration, a knowledge and memory layer restricted to a curated corpus, retrieval scope bound to the caller's resolved identity, and enterprise-tier inference with provider fallback, under an interaction policy configurable by user profile. Evaluated with 24 final-semester engineering students, it reached a Cronbach's alpha of 0.90, high perceived competence and autonomy, autonomous motivation for use, and logs showing iterative design refinement driven by the user. It did not satisfy relatedness, which marks the boundary of what the architecture replaces. Transfer to industrial onboarding remains a projected application until evaluated with practising personnel.

ACKNOWLEDGMENT

The authors acknowledge the support of Tecnológico de Monterrey and the students who participated in the training cycle.

REFERENCES

Atchley, P., Pannell, H., Wofford, K., Hopkins, M. and Atchley, R. A. (2024). Human and AI collaboration in the higher education environment: Opportunities and concerns. Cognitive Research: Principles and Implications, 9, 20.

Ba, S., Shi, X., Wu, S. and Lu, G. (2026). Artificial intelligence agents in computer-supported collaborative learning: A systematic literature review. Computers and Education: Artificial Intelligence, 10, 100579.

Caccavale, F., Gargalo, C. L., Gernaey, K. V. and Krühne, U. (2024). Towards Education 4.0: The role of Large Language Models as virtual tutors in chemical engineering. Education for Chemical Engineers, 49, pp. 1–11.

Davar, N. F., Dewan, M. A. A. and Zhang, X. (2025). AI chatbots in education: Challenges and opportunities. Information, 16(4), 235.

Dever, D., Wiedbusch, M. and Azevedo, R. (2022). Enhancing learner models for pedagogical agent scaffolding of self-regulated learning, in: Proceedings of the 30th International Conference on Computers in Education, Asia-Pacific Society for Computers in Education.

Es, S., James, J., Espinosa-Anke, L. and Schockaert, S. (2024). RAGAS: Automated evaluation of retrieval augmented generation, in: Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations. Association for Computational Linguistics, pp. 150–158.

Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M. and Wang, H. (2023). Retrieval-augmented generation for large language models: A survey. arXiv preprint arXiv:2312.10997.

Greshake, K., Abdelnabi, S., Mishra, S., Endres, C., Holz, T. and Fritz, M. (2023). Not what you've signed up for: Compromising real-world LLM-integrated applications with indirect prompt injection, in: Proceedings of the 16th ACM Workshop on Artificial Intelligence and Security. New York: ACM, pp. 79–90.

He, Z., Nguyen, T., Miari, T., Aliasgari, M., Rafatirad, S. and Sayadi, H. (2024). The AI companion in education: Analyzing the pedagogical potential of ChatGPT in computer science and engineering, in: 2024 IEEE Global Engineering Education Conference (EDUCON). IEEE.

Holmes, W. and Tuomi, I. (2022). State of the art and practice in AI in education. European Journal of Education, 57(4), pp. 542–570.

Lan, M. and Zhou, X. (2025). A qualitative systematic review on AI empowered self-regulated learning in higher education. npj Science of Learning. doi:10.1038/s41539-025-00319-0

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S. and Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks, in: Advances in Neural Information Processing Systems, 33, pp. 9459–9474.

Li, J., King, R. B., Chai, C. S., Zhai, X. and Lee, V. W. Y. (2025a). The AI Motivation Scale (AIMS): A self-determination theory perspective. Journal of Research on Technology in Education. doi:10.1080/15391523.2025.2478424

Li, J., Zhang, J., Chai, C. S., Lee, V. W. Y., Zhai, X., Wang, X. and King, R. B. (2025b). Analyzing the network structure of students' motivation to learn AI: A self-determination theory perspective. npj Science of Learning. doi:10.1038/s41539-025-00339-w

Mina, V., Silvestre, M. and Otero, L. (2021). Aprendizaje autorregulado en estudiantes de ingeniería: Estrategias de gestión de recursos pedagógicos. Anales AFA, 32(1), pp. 32–38.

Peters, D., Calvo, R. A. and Ryan, R. M. (2018). Designing for motivation, engagement and wellbeing in digital experience. Frontiers in Psychology, 9, 797.

Ryan, R. M. and Deci, E. L. (2000). Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. American Psychologist, 55(1), pp. 68–78.

Walter, Y. (2024). Embracing the future of artificial intelligence in the classroom: The relevance of AI literacy, prompt engineering, and critical thinking in modern education. International Journal of Educational Technology in Higher Education, 21, 15.