

Goal Conditioned Multi-Agent Architecture for Human-in-the-Loop Continual Learning in Education: The PSIS Framework

Adam Zahi^{1,*}, Diego H. Peluffo-Ordóñez²

¹SDAS Research Group, Rabat, Morocco

²Yachay Tech University, San Miguel de Urcuquí, Ecuador

Abstract

Most deployed educational AI systems are reactive by design: they answer questions, generate content, or recommend materials in response to immediate learner requests, but they do not maintain a persistent representation of what the learner is ultimately trying to achieve, and they do not autonomously select actions to close the distance between the learner's current state and that objective. This paper introduces the Personal Success Intelligence System (PSIS), a conceptual architecture that treats a learner-declared objective as a persistent coordination condition across agents and sessions. The central contribution is the Goal-Locked Coordination Protocol (GLCP), a closed-loop mechanism that maps a declared goal onto a target competency profile, estimates the learner's current state, computes a structured gap vector, reorients four role-specialised agents Student, Teacher, Expert, and Orchestrator around that gap, generates a constrained weekly plan aligned with their life, academic, and career goals, monitors execution, and incorporates multi-stakeholder human feedback continuously. We distinguish PSIS explicitly from GenMentor [17], the closest prior work, along four dimensions that the present architecture directly addresses: the absence of a cross-session accountability loop, the absence of structured HITL input from teachers and experts as distinct coordinating agents, the absence of formal goal-drift detection, and the absence of explicit learner autonomy mechanisms. PSIS is presented as a conceptual architecture supported by a falsifiable evaluation framework. No components have been implemented or empirically validated; the contribution is architectural and theoretical.

Keywords

Multi-agent systems, goal-conditioned coordination, human-in-the-loop AI, intelligent tutoring systems, continual learner modelling, knowledge tracing, educational AI

1. Introduction

One-to-one human tutoring remains, after more than four decades of research, the most effective instructional format known [3]. Intelligent Tutoring Systems were designed to approximate that effectiveness at scale [9, 16], and more recent systems have incorporated large language models, planning modules, memory, and multi-agent coordination [5, 7, 18]. Furthermore, autonomous agents can learn from interactions with their environment and adapt their behavior over time [19, 20]. Despite these advances, a structural limitation persists across almost all deployed systems: they are organised around the immediate request rather than a persistent, explicitly declared learner objective.

A student preparing for an algorithms examination can ask a reactive tutoring system to explain recursion, and a capable system will do so. What it cannot do is maintain, across multiple sessions, a formal representation of the student's declared target scoring above 90% on a specific examination by a specific date, compare that target with the student's current estimated competency state, select interventions specifically to reduce the gap between them, and follow up the following week to assess whether the plan was executed and also make sure to make the plans or the goals match the end future goals, like being a scientist or engineer or doctor, etc with searching, recommending and even applying for opportunities which will help him to achieve his goals, develop the needed skills without

ICAIW 2026: Workshops at the 9th International Conference on Applied Informatics 2026, October 13–17, 2026, Orlando, United States

*Corresponding author.

✉ collaborationadam@gmail.com (A. Zahi); diego.peluffo@sdas-group.com (D. H. Peluffo-Ordóñez)



© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

wasting time and effort on some unnecessary resources or learning something not useful. Those four capabilities together constitute what we call goal-conditioned coordination, and none of the deployed systems we are aware of provides all four simultaneously.

The closest prior work is GenMentor [17], an LLM-powered multi-agent framework for goal-oriented learning that maps learner goals to required skills and schedules a learning path. GenMentor establishes that goal-conditioned planning is both feasible and valuable in educational AI. The present work builds on this insight while addressing four limitations. First, GenMentor does not implement a cross-session accountability loop; planning and execution monitoring are not coupled through an autonomous follow-up mechanism. Second, GenMentor does not incorporate structured HITL input from teachers and domain experts as distinct coordinating agents with defined feedback channels; the system is effectively learner-facing only. Third, GenMentor does not include a formal goal-drift detection protocol; it does not distinguish between a learner who has temporarily paused and one who has implicitly abandoned their goal. Fourth, GenMentor does not provide explicit learner autonomy mechanisms such as plan rejection, pause, or goal reassessment. PSIS is specifically designed to address all four gaps.

The present paper makes four contributions, each of which is directly motivated by the gap analysis above:

1. The Goal-Locked Coordination Protocol (GLCP): a formal closed-loop mechanism converting a learner-declared objective into a persistent multi-agent coordination condition, with explicit goal-drift detection and learner autonomy mechanisms (4).
2. A four-agent architecture in which teachers, students, and domain experts provide structured HITL input through dedicated agents with defined responsibilities and feedback channels (3).
3. A competency-space formulation of learner state, target state, goal gap, and constrained plan selection, with a direct connection between mathematical quantities and Orchestrator decisions (4).
4. A falsifiable evaluation framework with explicit baselines, an ablation study, and research questions that isolate the contribution of goal locking from general multi-agent coordination (6).

The evaluation framework includes four research questions that govern the experimental design:

- RQ1. Does explicit goal locking improve learner goal achievement compared with reactive or non-goal-conditioned AI?
- RQ2. Does goal-conditioned multi-agent coordination reduce the weighted goal distance more efficiently than a single agent or non-goal-conditioned baseline?
- RQ3. Does prediction-guided planning improve the timing of interventions relative to reactive rule-based planning?
- RQ4. Do the bounded autonomy mechanisms preserve learner agency while maintaining or improving learning outcomes?

2. Related Work and Background

2.1. Intelligent Tutoring Systems and Learner Modelling

Intelligent Tutoring Systems formalised the problem of adapting instruction to individual learners [9, 16]. Bayesian Knowledge Tracing (BKT) [6] provided a principled probabilistic model of per concept mastery as a latent binary variable, and Deep Knowledge Tracing (DKT) [14] extended this using recurrent neural networks to capture cross-concept dependencies that the original Hidden Markov formulation cannot represent. PSIS uses both as state-estimation components; neither constitutes a novel contribution of the present work.

The critical distinction between state estimation and goal-conditioned coordination is this: BKT can estimate that a learner has low mastery of dynamic programming, but it cannot determine which

intervention should be selected to reduce that deficit within the learner’s available time before their examination. The additional mechanism required is a formal representation of the learner’s declared objective, a way of measuring the distance between the current state and that objective, and a planning process conditioned on both. PSIS provides all three through GLCP.

2.2. LLM Agents and Multi-Agent Systems in Education

Chu et al. [5] provide a comprehensive survey of LLM agents for education, identifying memory, planning, tool use, personalisation, and multi-agent coordination as key research directions. Zeng et al. [18] proposed GraphMASAL, combining a persistent learner model, a dynamic knowledge graph, and multiple coordinated agents. Guo et al. [7] survey multi-agent LLM systems more broadly, cataloguing progress in task decomposition, role-based specialisation, and inter agent communication.

Among existing systems, GenMentor [17] is the closest to PSIS in motivation and approach. GenMentor maps learner goals to required skills using a fine-tuned LLM, schedules a learning path through an evolving optimisation procedure, and tailors content to individual learner needs. It has been deployed in practice and validated through a human study. PSIS acknowledges and builds on this contribution. The four structural differences motivating PSIS cross-session accountability, multi stakeholder HITL feedback, goal drift detection, and explicit learner autonomy are elaborated in the Introduction and directly addressed in the GLCP design (4).

2.3. Goal-Conditioned Decision Making

Goal-conditioned reinforcement learning [1] establishes a principled framework for agents that pursue declared objectives rather than optimising for a fixed reward. Hindsight Experience Replay demonstrated that agents can learn efficiently from failed attempts by retrospectively treating reached states as alternative goals. The educational problem introduces constraints not typically present in robotic or game environments: goals may be qualitative, temporally bounded by real-world deadlines, subject to expert knowledge constraints, and governed by learner

Table 1

PSIS positioned against representative educational AI approaches. All PSIS entries are proposed. References: Cognitive Tutor [9]; LLM tutors [5]; GraphMASAL [18]; GenMentor [17].

System	Pers	Goal-c	M-ag	HITL	Acct	Auton
Cog. Tutor	Ltd	No	No	Ltd	No	No
LLM tutors	Session	No	No	No	No	No
GraphMASAL	Mod	Implicit	Yes	Ltd	No	No
GenMentor	Yes	Yes	Yes	No	No	No
PSIS (prop.)	Yes	Yes	Yes	Yes	Yes	Yes

Pers=persistent learner model; Goal-c=goal-conditioned; M-ag=multi-agent;

HITL=structured human-in-the-loop; Acct=cross session accountability; Auton=explicit learner autonomy requirements that preclude fully autonomous pursuit.

2.4. Human-in-the-Loop Learning

HITL machine learning provides a framework for maintaining AI systems aligned with human intent through structured feedback [13]. Alignment through instruction-following [12] demonstrates that human feedback can systematically shape model behaviour. UNESCO has documented a gap between HITL governance principles for educational AI and the architectures needed to realise them in practice [15]. PSIS addresses this gap by giving each category of human stakeholder, student, teacher, domain expert, a dedicated agent with defined responsibilities and feedback protocols.

Table 1 summarises how PSIS positions against representative existing approaches across six dimensions. Entries marked “Yes” for PSIS reflect proposed rather than implemented capabilities.

3. Methodology: The PSIS Architecture and GLCP

3.1. Overview

PSIS organises around four agents across three functional layers, illustrated in Figure 1. The *human input layer* is where the Teacher, Student, and Expert agents receive signals from their respective stakeholders. The state and coordination layer maintains the learner model, goal state, knowledge graph, and decision history. The action and output layer produces weekly plans, alerts, recommendations, opportunities, and human escalation events. The Orchestrator Agent spans all three layers, executing GLCP and coordinating the other three agents.

3.2. Why Four Agents and Not a Centralised Architecture

The four agent decomposition is motivated by functional and epistemic separation, not by an assumption that more agents are inherently better. The question of whether the decomposition is justified is in fact one of the research questions the evaluation is designed to answer (see A2 in 6).

Each agent maps to a distinct category of human stakeholder with a different relationship to the educational system. Teacher pedagogy, learner state, domain knowledge, and coordination logic have different data sources, different update frequencies, and different validation requirements. Collapsing all four into a single agent would require that agent to simultaneously handle teacher style preference encoding, probabilistic learner state estimation, expert knowledge graph maintenance, and goal-conditioned planning, each of which involves distinct model types, update mechanisms, and human oversight channels. The decomposition also provides the structural basis for ablation: removing individual agents in controlled experiments directly tests whether each component contributes measurable value.

Agents vs functional modules. Each component is designated an agent rather than a module because each maintains its own persistent state, receives inputs from a human stakeholder, applies a domain-specific reasoning process, and produces outputs that are independently interpretable and auditable. The Teacher Agent does not merely retrieve records; it builds and updates a dynamic Pedagogical Digital Twin. The Student Agent does not merely log responses; it maintains a probabilistic belief state over competency mastery. The Expert Agent does not merely store knowledge; it validates incoming updates against the existing knowledge graph and flags conflicts for human resolution. These state-bearing, reasoning-performing, stakeholder-facing components satisfy the standard definition of agents in multi-agent systems [7].

3.3. The Student Agent

The Student Agent maintains the longitudinal learner model $\mathbf{s}_t \in [0, 1]^d$, updated after every interaction from assessment performance per concept, time on task, error type distributions, and session consistency. The d dimensions correspond directly to the assessable competencies in C (defined in 4), ensuring that the learner model is always expressed in the same space as the target competency profile. This is what makes the gap vector computationally meaningful rather than merely metaphorical.

3.4. The Teacher Agent

The Teacher Agent processes lesson materials, concept-difficulty annotations, and student response evaluations from instructors. Over time, it builds a *Pedagogical Digital Twin*: a persistent representation of which instructional approaches have historically been effective for learners with specific difficulty profiles. This is constructed from interaction logs recording which explanations produced mastery gains in learners with similar error patterns. Quality is evaluated by the correlation between retrieved explanations and subsequent mastery gains in matched learner profiles. When the Orchestrator identifies that a student is struggling with a concept, it queries the Teacher Agent for the Pedagogical Digital Twin most likely to help given the student's current profile.

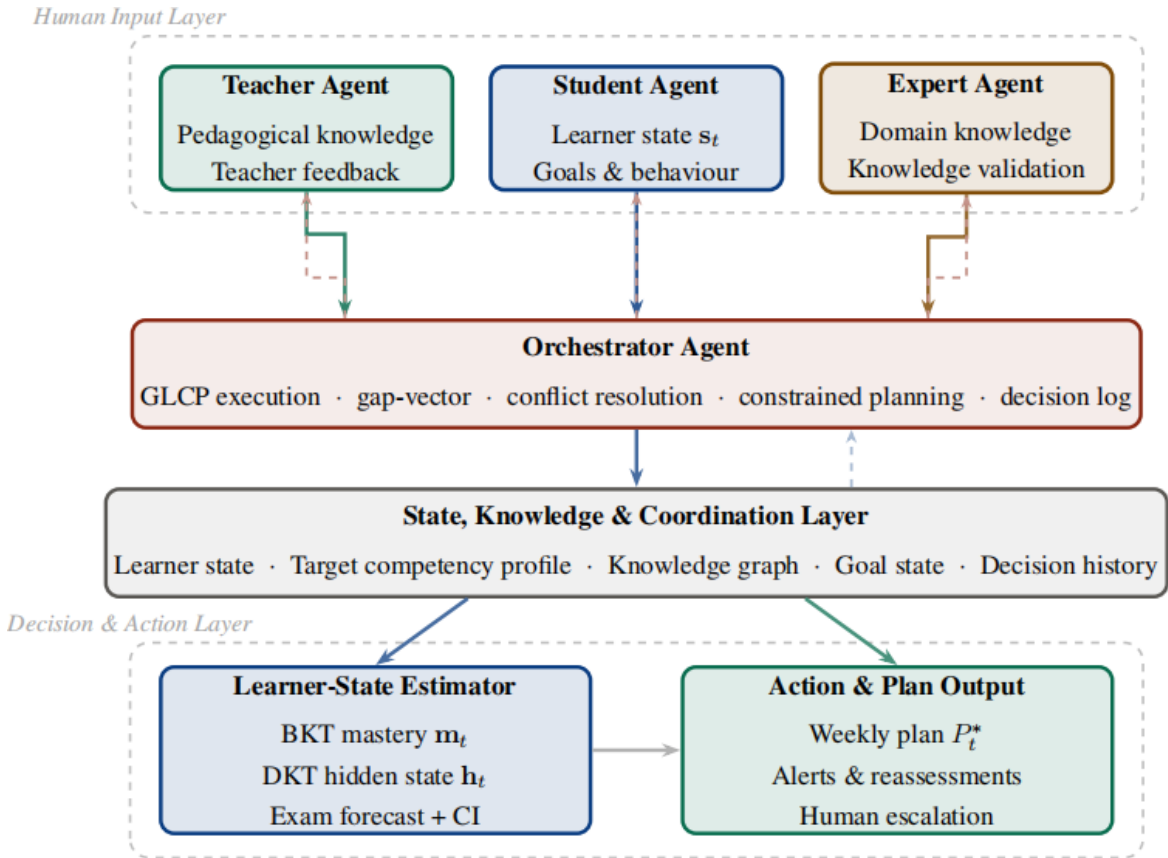


Figure 1: PSIS three-layer architecture. Solid arrows show primary data flow; dashed arrows show Orchestrator feedback to domain agents. The Orchestrator executes GLCP continuously, coupling the learner-state estimator to the constrained planning output across all sessions.

3.5. The Expert Agent

The Expert Agent manages domain knowledge integrity. Expert contributions undergo automated consistency checking against the existing knowledge graph, and conflicting opinions are flagged for human resolution rather than merged automatically. Outdated knowledge is deprecated rather than deleted, preserving the historical validity of older learner models. This separation of concerns is necessary because incorrect knowledge propagated to multiple learners simultaneously would be difficult to detect and correct.

3.6. The Orchestrator Agent

The Orchestrator is the most technically central component. It executes GLCP, resolves inter-agent conflicts, generates constrained weekly plans, and logs every decision with its causal chain. The connection between the mathematical formulation and the Orchestrator’s actual decisions is as follows. When the Orchestrator receives a goal declaration, it calls the competency embedding function to compute y^{g^*} (the target mastery profile) and immediately computes $\Delta_t = \max(0, y^{g^*} - s_t)$ element-wise using Equation 1. The weighted distance D_t from Equation 2 then becomes the objective function for plan selection in Equation 3: the Orchestrator chooses the plan that minimises the expected value of D_{t+1} given the current state and the available study time. After each week, the updated D_{t+1} determines whether the student is on track (gap decreasing as expected), behind (gap decreasing less than projected), or drifting (gap stable or increasing). This directly drives the three-way branching behaviour of the accountability loop.

When agent outputs conflict, the Orchestrator applies a priority-ordered arbitration policy: safety and factual correctness take precedence, followed by explicit learner constraints from the autonomy

mechanisms, followed by goal relevance measured by the gap component each recommendation addresses, followed by expected gap reduction per unit of learner time. When no automatic resolution is possible with sufficient confidence, the Orchestrator escalates to a human stakeholder with an explainable summary of the conflict.

4. The Goal-Locked Coordination Protocol (GLCP)

GLCP is the central contribution. The word “locked” does not mean the goal is immutable; it means the goal remains an active coordination condition on all agent decisions until it is explicitly modified, paused, replaced, or achieved.

4.1. Goal Declaration and Competency Representation

A learner declares $g^* = (d, \tau, q, \pi)$ where d is the desired outcome, τ the deadline, q the performance threshold, and π learner preferences. The system maps g^* to a target competency profile $y^{g^*} \in [0, 1]^d$ where each $y_i^{g^*}$ is the mastery level required for competency $c_i \in \mathcal{C}$. The competency space $\mathcal{C}$ is constructed from curriculum requirements, examination specifications, prerequisite graphs, and expert validation; it is the same space in which the learner model s_t is expressed. This alignment is essential: it is what makes the gap vector a meaningful quantity rather than an arbitrary distance measure.

4.2. Gap Vector Computation

The per competency goal gap is:

$$\Delta_{t,i} = \max\left(0, y_i^{g^*} - s_{t,i}\right). \quad (1)$$

with weighted goal distance:

$$D_t = \left(\sum_{i=1}^d w_i \Delta_{t,i}^2\right)^{1/2}. \quad (2)$$

where w_i encodes examination weighting, prerequisite centrality in the knowledge graph, and deadline pressure. Competencies for which $\Delta_{t,i} = 0$ have been achieved and are excluded from planning. Competencies with the largest weighted $\Delta_{t,i}$ receive the highest priority in plan selection. The $\max(0, \cdot)$ formulation is deliberate: over achievement on a competency is not penalised and does not create negative contributions to D_t .

4.3. Constrained Plan Selection

The Orchestrator selects the weekly plan that minimises expected residual distance:

$$\mathcal{P}_t^* = \arg \min_{\mathcal{P} \in \mathbb{P}} \mathbb{E}[D_{t+1} \mid \mathcal{P}, s_t, g^*] \quad \text{s.t.} \quad \sum_{a \in \mathcal{P}} \text{time}(a) \leq T_t, \mathcal{P} \in P(K_t, C_t, \pi). \quad (3)$$

where T_t is the learner’s declared weekly time budget, K_t encodes prerequisite constraints from the knowledge graph, and π encodes learner preferences from prior autonomy actions. The expectation in Equation 3 is evaluated using the BKT and DKT estimates described below: specifically, each candidate plan is evaluated by simulating the expected mastery update Δ s under the BKT transition model for each concept study assignment, and D_{t+1} is computed from the resulting projected state.

4.4. Bounded Learner Autonomy

The system pursues goals continuously but not unconditionally. Learners retain four explicit controls: (1) goal modification, triggering recomputation of y^{g^*} and Δ_t ; (2) plan rejection of individual weekly assignments, causing the Orchestrator to regenerate from the next best plan; (3) pause, suspending GLCP without deleting the learner model; (4) diagnostic reassessment of any competency, triggering a targeted evaluation and immediate state update. All autonomy actions are logged and feed back into the learner model as valid behavioural signals.

4.5. Goal-Drift Detection

The protocol distinguishes three types of behavioural inconsistency. A temporary external constraint is indicated by a pause request or session frequency change without goal modification; the response is a reduced intensity plan adjustment. A motivational fluctuation is indicated by declining engagement metrics without goal-related autonomy actions; the response is a supportive content format change. Genuine goal abandonment is indicated by sustained disengagement (two consecutive weeks of gap non-decrease) combined with explicit autonomy signals; only this case triggers a reconfirmation dialogue. This three-way classification prevents the system from interpreting every deviation from the planned schedule as goal abandonment, which would be both paternalistic and inaccurate.

4.6. Learner State Estimation

BKT [6] models per-concept mastery as a latent binary variable. After observing a response, the posterior mastery probability is updated by Bayes' rule, and the transition probability is applied:

$$P(L_t^c) = P(L_{t-1}^c \mid \text{obs}_t) + (1 - P(L_{t-1}^c \mid \text{obs}_t)) P(T^c), \quad (4)$$

$$P(\text{correct} \mid L_t^c) = P(L_t^c) (1 - P(S^c)) + (1 - P(L_t^c)) P(G^c). \quad (5)$$

DKT [14] captures cross-concept dependencies that BKT cannot model:

$$\mathbf{h}_t = \text{LSTM}(\mathbf{x}_t, \mathbf{h}_{t-1}), \quad \hat{y}_{t+1} = \sigma(\mathbf{W}\mathbf{h}_t + \mathbf{b}). \quad (6)$$

The combined state estimator $\mathbf{s}_t = f_\theta(\mathbf{m}_t^{\text{BKT}}, \mathbf{h}_t^{\text{DKT}}, \mathbf{z}_t)$ feeds directly into Equation 1. Examination score forecasting with calibrated uncertainty [4]:

$$\hat{y}_{\text{exam}} = f_\theta(\mathbf{s}_t, \mathbf{z}_t), \quad \text{CI}_{95} = [\hat{\mu} - 1.96\hat{\sigma}, \hat{\mu} + 1.96\hat{\sigma}]. \quad (7)$$

The Orchestrator uses $\hat{\sigma}$ directly: when prediction uncertainty is high, the planned gap closure rate is set more conservatively and human review is triggered earlier than the default schedule.

5. Conceptual Results: Illustrative Scenario

Since PSIS has not been implemented, we present a conceptual result in the form of an end-to-end trace through the protocol, intended to demonstrate the architectural properties and make the mathematical formulation concrete.

A student declares $g^* = \text{"Score} \geq 90\% \text{ on CS301 Algorithms by June 20."}$ The Orchestrator maps this to twelve competencies and computes the initial gap vector. The two largest components are $\Delta_{0,\text{rec}} = 0.54$ (recursion) and $\Delta_{0,\text{dp}} = 0.43$ (dynamic programming); sorting is near mastery at $\Delta_{0,\text{sort}} = 0.05$, so sorting receives zero weight in

$$\mathcal{P}_1^*.$$

The Orchestrator selects $\mathcal{P}_1^*$ by evaluating candidate plans against Equation 3 and chooses a plan prioritising recursion in Week 1 and dynamic programming in Week 2, with examination strategy in Week 3. The Teacher Agent surfaces the Pedagogical Digital Twin whose prior outcomes most closely

match the student’s error pattern on recursion. The Expert Agent confirms that all content is aligned with the current CS301 syllabus.

By the end of Week 2, the BKT updates yield $\hat{s}_{\text{rec}} = 0.71$, closing the recursion gap substantially. The state estimator forecasts $82\% \pm 7\%$ on the examination, below the 90% target. Because $\hat{\mu} + 1.96\hat{\sigma} = 96\% > 90\%$ the upper confidence bound clears the target, but the point estimate does not; the Orchestrator interprets this as a moderate risk situation requiring increased recursion intensity. The student rejects the proposed peer study session; the Orchestrator logs this as a preference signal, updates π , and regenerates an alternative plan using additional solo dynamic programming problems. The learner pauses for four days due to a concurrent deadline; the goal state is preserved, and planning resumes automatically when session activity is detected.

This trace illustrates three properties of GLCP that distinguish it from GenMentor [17]: the cross-session persistence of the gap vector D_t without requiring the learner to redeclare their goal; the integration of the forecast uncertainty $\hat{\sigma}$ into Orchestrator decision-making; and the logged response to a specific learner autonomy action (plan rejection) that modifies the plan without abandoning the goal.

Table 2

Experimental conditions for system-level evaluation. Conditions are ordered by increasing architectural complexity; the critical comparison for RQ1 and RQ2 is PSIS vs. conditions 3 and 4, respectively.

Condition	Pers	Goal-c	Multi-ag	Pred	HITL
1. Control	No	No	No	No	Human
2. Reactive LLM tutor	No	No	No	No	No
3. Adaptive ITS	Yes	Implicit	No	Yes	Ltd
4. Non-goal multi-agent	Yes	No	Yes	Yes	Yes
5. PSIS (full)	Yes	Yes	Yes	Yes	Yes

6. Evaluation Framework

The most important methodological principle is to distinguish component validity from architectural validity. A BKT model achieving high prediction accuracy does not demonstrate that the GLCP improves learning outcomes; a multi-agent system producing coherent plans does not demonstrate that goal locking adds value beyond general personalisation. The evaluation is therefore designed at two levels.

6.1. System Level Experimental Conditions

Table 2 specifies five experimental conditions, directly aligned with RQ1 and RQ2. The critical comparison between PSIS vs the non-goal-conditioned multi-agent baseline isolates the specific contribution of goal locking from the general effects of multi-agent coordination and persistent learner modelling.

6.2. Ablation Study

Five ablations remove one architectural component at a time, each directly testing a sub-hypothesis:

A1 (no GLCP): replace the coordination protocol with general personalised recommendations; tests whether goal locking adds value beyond personalisation (direct test of RQ1). A2 (no Orchestrator): use fixed agent routing

without goal-conditioned arbitration; tests whether the multi-agent decomposition is justified. A3 (no prediction): replace forecast-guided planning with reactive rule-based triggers; tests RQ3. A4 (no HITL): freeze stakeholder models, removing teacher and expert feedback; tests the contribution of multi-stakeholder HITL input. A5 (no persistent state): reset learner model between sessions; establishes the importance of longitudinal tracking.

6.3. Metrics

The primary architectural metric is the Goal Completion Rate: $GCR = (D_0 - D_T)/D_0$, where D_0 and D_T are the weighted goal distances at the start and end of the study period. GCR must be interpreted alongside actual learning outcomes, not as a substitute for them.

Component-level benchmarks (prospective targets against published results): BKT AUC-ROC > 0.82 [14]; DKT AUC-ROC > 0.86 [14]; score forecast RMSE < 8 pts relative to the course mean baseline of 9.1 points [14]; dropout recall at six weeks > 0.72 relative to the logistic regression baseline of 0.70 [8]. The 60% milestone rate target is explicitly exploratory, calibrated against typical unsupported online completion rates of 15–40%.

6.4. Detailed Experimental Protocol

Each experimental session follows a structured procedure to ensure reproducibility. At enrolment, each participant completes a 20-minute diagnostic assessment mapped to the competency space C , establishing the initial learner state s_0 . Participants in goal-conditioned conditions (4 and 5) then declare a specific, measurable learning goal for the study period; those in conditions 1,3 receive equivalent time on task exposure without a formal goal declaration. All participants interact with their assigned system for twelve weeks, with weekly assessments administered every seven days to update the learner model.

Primary outcome data are collected at week 12. For conditions 3, 4, and 5, the system logs all Orchestrator decisions with their causal chain, enabling post hoc verification that decisions are mathematically consistent with the gap vector and plan selection equations (Eqs. 3 and 2). Inter-rater reliability of educator plan evaluations is assessed using Cohen's κ with a minimum acceptable threshold of 0.70. Power analysis indicates that $n = 120$ learners per condition (600 total across five conditions) achieves 80% power to detect a medium effect size ($d = 0.5$) on the primary outcome at $\alpha = 0.05$. Sample size is consistent with published ITS evaluation studies [16]. All outcome analyses are pre-registered before data collection to prevent post-hoc hypothesis adjustment.

Students would be assigned via stratified randomisation controlling for prior knowledge, motivation, instructor, and declared goal difficulty. A minimum of two participating universities in the MENA region are required for statistical power and generalisability. IRB approval and explicit informed consent for all forms of behavioural monitoring are non negotiable prerequisites.

7. Discussion

7.1. Ethical Considerations and Safety

PSIS adopts four safety principles: reversibility of autonomous recommendations; unconditional human override authority; uncertainty aware escalation (increasing prediction uncertainty triggers more conservative planning and earlier human review); and data minimisation (only data required for the stated educational function is collected). Dropout risk estimates are treated as intervention signals, not labels; surfacing them to instructors without appropriate training and interface design risks stigmatising learners or producing self-fulfilling effects.

Learner behavioural data constitutes detailed surveillance. Explicit informed consent, clear retention and deletion policies, and compliance with applicable privacy frameworks (GDPR or equivalent national regulations) are prerequisites for any deployment, not optional additions.

If cross-institutional sharing of pedagogical intelligence is pursued in future work, it would require federated learning with differential privacy guarantees [11]. The specific privacy budget (ϵ, δ) would be determined in consultation with an independent ethics board before any data sharing begins.

7.2. Limitations

No implementation. All PSIS components are conceptual. The evaluation framework is a research plan; no empirical claims are made about system performance.

Goal representation. The mapping of natural-language objectives to quantitative competency vectors is a non-trivial open research problem. Whether the computed gap D_t has genuine pedagogical meaning rather than serving as a useful engineering heuristic is an empirical question that the proposed evaluation will partially address.

Cold start behaviour. BKT and DKT require interaction histories. New learners need fallback strategies: population priors combined with a brief diagnostic assessment at goal declaration.

Orchestrator conflict resolution. The priority-ordered arbitration policy is described at the level of design intent. A concrete implementation requires careful engineering, particularly when teacher and student agent outputs conflict on goal-critical concepts.

Continual learning scope. Continual adaptation in PSIS primarily refers to learner state updating and HITL-guided model revision, not to continuous online retraining of neural components. The risk of catastrophic forgetting [10] in the DKT component requires explicit mitigation strategies such as experience replay or elastic weight consolidation in any future implementation.

7.3. Future Work Roadmap

The proposed research proceeds in six stages: (1) validate BKT and DKT independently on the assistments dataset [2]; (2) implement a minimal GLCP with Student Agent and Orchestrator only, simulating Teacher and Expert inputs; (3) evaluate GLCP plans against reactive baselines using synthetic learner trajectories; (4) conduct educator review of generated plans for pedagogical quality and goal relevance; (5) small controlled pilot comparing PSIS against the strongest feasible baseline; (6) full multi-condition study only after component and pilot evidence and developing an advanced prediction and simulation technologies by using quantum concepts. This staged approach avoids validating the entire architecture simultaneously.

8. Conclusion

This paper presented PSIS, a conceptual architecture for goal conditioned, human-in-the-loop continual learning in education. The central contribution is the Goal Locked Coordination Protocol, which converts a learner-declared objective into a persistent coordination condition across four role-specialised agents, implementing a closed loop of gap computation, constrained planning, execution monitoring, autonomy-respecting adjustment, and goal drift detection. PSIS addresses four specific limitations of GenMentor, the closest prior work: the absence of a cross-session accountability loop, structured multi stakeholder HITL input, formal goal drift detection, and explicit learner autonomy mechanisms.

The value of this framework will be determined by empirical evidence. If controlled evaluation demonstrates that goal locking provides benefits beyond personalisation, prediction, and multi-agent coordination, GLCP may serve as a general architectural principle for educational agents with long horizon objectives. If those benefits are not observed, the ablation study will identify which components do not justify their complexity. In either case, the proposed evaluation framework with its five experimental conditions, five ablations, and four research questions provides a principled basis for that determination.

Acknowledgments

The author thanks the SDAS Groupe research team for ongoing discussion and the WAIAMS 2026 programme committee for constructive reviewer feedback.

Declaration on Generative AI

During the preparation of this work, the authors used AI tools for grammar checking, language editing, and structural revision of the authors' own research ideas. The authors reviewed and edited all content and take full responsibility for the scientific content, claims, mathematical formulations, and references in the manuscript. No AI system is listed as an author.

References

- [1] M. Andrychowicz, F. Wolski, A. Ray, J. Schneider, R. Fong, P. Welinder, B. McGrew, J. Tobin, P. Abbeel, W. Zaremba, Hindsight experience replay, in: *Advances in Neural Information Processing Systems*, vol. 30, Curran Associates, 2017.
- [2] ASSISTments, ASSISTments dataset for skill builder 2009–2010, 2015. Accessed: 2026.
- [3] B. S. Bloom, The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring, *Educational Researcher* 13(6) (1984) 4–16.
- [4] T. Chen, C. Guestrin, XGBoost: A scalable tree boosting system, in: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, 2016, pp. 785–794.
- [5] Z. Chu, S. Wang, J. Xie, T. Zhu, Y. Yan, J. Ye, A. Zhong, X. Hu, J. Liang, P. S. Yu, Q. Wen, LLM agents for education: Advances and applications, in: *Findings of the Association for Computational Linguistics: EMNLP 2025*, Association for Computational Linguistics, 2025, pp. 13782–13810. arXiv:2503.11733.
- [6] A. T. Corbett, J. R. Anderson, Knowledge tracing: Modeling the acquisition of procedural knowledge, *User Modeling and User-Adapted Interaction* 4(4) (1994) 253–278.
- [7] T. Guo, X. Chen, Y. Wang, R. Chang, S. Pei, N. V. Chawla, O. Wiest, X. Zhang, Large language model based multi-agents: A survey of progress and challenges, in: *Proceedings of the 33rd International Joint Conference on Artificial Intelligence*, 2024. arXiv:2402.01680.
- [8] M. Kloft, F. Stiehler, Z. Zheng, N. Pinkwart, Predicting MOOC dropout over weeks using machine learning methods, in: *Proceedings of the EMNLP Workshop on Analysis of Large Scale Social Interaction in MOOCs*, 2014, pp. 60–65.
- [9] K. R. Koedinger, J. R. Anderson, W. H. Hadley, M. A. Mark, Intelligent tutoring goes to school in the big city, *International Journal of Artificial Intelligence in Education* 8 (1997) 30–43.
- [10] M. McCloskey, N. J. Cohen, Catastrophic interference in connectionist networks: The sequential learning problem, *Psychology of Learning and Motivation* 24 (1989) 109–165.
- [11] B. McMahan, E. Moore, D. Ramage, S. Hampson, B. Agüera y Arcas, Communication-efficient learning of deep networks from decentralized data, in: *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, PMLR, 2017, pp. 1273–1282.
- [12] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, et al., Training language models to follow instructions with human feedback, in: *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 27730–27744.
- [13] D. H. Peluffo-Ordóñez, et al., Reflections on modern perspectives in human-in-the-loop AI, *Revista Científica* (2026).
- [14] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. Guibas, J. Sohl-Dickstein, Deep knowledge tracing, in: *Advances in Neural Information Processing Systems*, vol. 28, Curran Associates, 2015.
- [15] UNESCO, Beyond the loop: Reclaiming pedagogy in an AI age, Technical report, United Nations Educational, Scientific and Cultural Organization, Paris, France, 2025.
- [16] K. VanLehn, The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems, *Educational Psychologist* 46(4) (2011) 197–221.
- [17] T. Wang, L. Li, Z. Cui, et al., GenMentor: LLM-powered multi-agent framework for goaloriented learning in intelligent tutoring systems, in: *Companion Proceedings of the ACM Web Conference 2025*, 2025.

- [18] B. Zeng, M. Liu, Z. Zhen, GraphMASAL: A graph-based multi-agent system for adaptive learning, in: *Proceedings of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026)*, 2026. arXiv:2511.11035.
- [19] J. Hernandez, D.H. Peluffo-Ordoñez, H Florez. 2026. How do autonomous software agents learn? A systematic literature review. *PeerJ Computer Science* 12:e3756 <https://doi.org/10.7717/peerj-cs.3756>
- [20] J. Hernandez, M.F. Pollo-Cattaneo, D.H. Peluffo-Ordóñez, H. Florez (2026). Cognitive Architecture for Learning in Autonomous Agents: A Study in Discrete Navigation. In: Florez, H., Peluffo-Ordoñez, D. (eds) *Applied Informatics. ICAI 2025. Communications in Computer and Information Science*, vol 2667. Springer, Cham. https://doi.org/10.1007/978-3-032-07175-0_1