eISSN: 2229-3213 pISSN: 2229-3205
The Journal of Applied Science and Social Science
Volume 16,Issue 03 March, 2026

METHODOLOGY FOR DEVELOPING STUDENT COMPETENCIES USING TOOLS THAT INITIALIZE REPETITIVE PROBLEM SITUATIONS BASED ON PRIOR KNOWLEDGE

Hamroyev Sanjar Hoshimovich

Bukhara Regional Office of the Silk Road Bank

E-mail: sanjar.hamroyev@gmail.com

Ibragimov Ulugbek Muradilloyeivch

Bukhara state technical university

Email: ciulugbek@list.ru

Abstract: This article presents a structured methodology for developing student competencies through the systematic use of software tools capable of generating and reinitializing repetitive problem situations grounded in students' prior knowledge. The core premise is that meaningful competency formation requires repeated, contextually varied encounters with problem types that activate and build upon existing cognitive schemas. Drawing on principles of spaced repetition, adaptive scaffolding, and problem-based learning, the proposed methodology integrates intelligent tutoring systems, scenario-generation platforms, and learning management systems to produce personalized, cyclically refreshed problem sets. An experimental study conducted at Asia International University, Bukhara, involving 62 undergraduate students across two cohorts, demonstrated that the proposed approach yields a normalized learning gain of g = 0.64 in technical problem-solving competencies, significantly outperforming the conventional curriculum (g = 0.29). The article discusses the architectural requirements for competency-initializing tools, their integration into the academic workflow, and recommendations for practical adoption in higher education institutions.

Keywords: competency development, prior knowledge, problem-based learning, spaced repetition, adaptive scaffolding, intelligent tutoring system, scenario generation, formative assessment, higher education, personalized learning.

Modern higher education is increasingly expected to produce graduates who not only possess declarative knowledge but are capable of applying that knowledge flexibly across novel, unpredictable situations — the hallmark of genuine competency. Yet a persistent challenge in university pedagogy is the tendency of students to acquire knowledge in isolated episodes without sufficient rehearsal or contextual variation to consolidate it into durable, transferable competencies [1, 3].

Cognitive science research dating back to Ebbinghaus (1885) and systematically validated by Cepeda et al. (2006) establishes that distributed practice — the deliberate re-encounter with material across spaced intervals — produces retention far superior to massed study. More recent work in problem-based learning (PBL) demonstrates that competency formation is accelerated when each practice episode is designed to activate and extend prior knowledge rather than treat every session as a blank-slate introduction [2, 5].

Software tools capable of dynamically generating and reinitializing problem situations offer a technologically scalable means of implementing these principles in classroom practice. By parameterizing problem templates with variables drawn from a student's established knowledge profile, such tools can produce an effectively unlimited supply of contextually appropriate, appropriately challenging tasks. This article proposes a complete methodology for exploiting this capability within a university curriculum and presents empirical evidence of its effectiveness.

1. Theoretical Foundations

Three interlocking theoretical frameworks underpin the proposed methodology:

1.1 Schema Theory and Prior Knowledge Activation. Ausubel's assimilation theory (1968) posits that new knowledge is most effectively anchored when it is explicitly connected to existing cognitive structures — what he termed 'advance organizers'. When problem-initializing tools are designed to surface and interrogate prior knowledge before presenting a new challenge, they fulfill the function of advance organizers algorithmically, ensuring that each problem episode begins from a pedagogically sound activation baseline [4].

1.2 Spaced Repetition and Desirable Difficulty. Bjork's (1994) concept of 'desirable difficulties' identifies spacing, interleaving, and variation as learning conditions that feel harder in the short term but produce markedly superior long-term retention and transfer. Tools that reinitialize problem situations with controlled parameter variation operationalize all three desirable difficulties simultaneously, creating a practice environment that is both demanding and educationally efficient [6].

1.3 Competency-Based Education (CBE). CBE frameworks, as formalized by the European Qualifications Framework (EQF) and the national higher education standards of Uzbekistan, define competency as the integrated ability to mobilize knowledge, skills, and attitudes to perform complex tasks in varied contexts. The proposed methodology directly targets this integration by ensuring that each repeated problem situation demands not mere recall but the coordinated application of multiple previously acquired competency components [7].

2. Architectural Requirements for Competency-Initializing Tools

For a software tool to effectively initialize competency-building problem situations grounded in prior knowledge, it must satisfy a set of functional and architectural requirements. Based on a systematic analysis of existing intelligent tutoring systems (ITS) and adaptive learning platforms, the following five requirements are identified as essential:

Student Knowledge Model (SKM). The tool must maintain and continuously update a structured representation of each student's demonstrated competency across all relevant sub-domains. This model serves as the primary input for problem parameterization, ensuring that generated tasks are appropriately calibrated to the student's current knowledge frontier.

Problem Template Repository. A library of parameterized problem templates must be provided, each annotated with the competency components it targets and the prerequisite knowledge it assumes. Templates are distinguished from fixed exercises by the presence of variable slots that can be populated with values drawn from the SKM.

Parameter Initialization Engine (PIE). The PIE is the core algorithmic component that selects and populates problem templates based on the current SKM state, the desired difficulty level, and the spaced repetition schedule. It ensures that each generated problem activates known prior knowledge while introducing a controlled increment of novelty — the pedagogical 'stretch zone'.

Formative Assessment and Feedback Module. Immediate, structured feedback is essential for competency development. The tool must evaluate student responses, update the SKM accordingly, and provide explanatory feedback that explicitly connects the correct solution strategy to the prior knowledge components it mobilizes.

Analytics Dashboard. Instructors require a real-time view of aggregate and individual competency trajectories to calibrate classroom instruction, identify struggling students early, and adjust the difficulty parameters of the problem generator.

Requirement	Primary Function	Output	Beneficiary
Student Knowledge Model	Track competency state	Personalized profile	PIE, Instructor
Template Repository	Store problem patterns	Parameterizable tasks	PIE
Parameter Init. Engine	Generate problems	Calibrated exercises	Student
Feedback Module	Evaluate & explain	Formative guidance	Student, SKM
Analytics Dashboard	Monitor progress	Competency reports	Instructor

Table 1. Functional components of a competency-initializing tool.

3. The Proposed Methodology

The methodology is organized as a five-stage cyclical process. Unlike linear instructional designs, the cycle is intentionally recursive: upon completion of Stage 5, the updated Student Knowledge Model feeds back into Stage 2, initiating a new iteration with recalibrated problem parameters. This recursive architecture is what distinguishes the proposed approach from conventional problem-set-based practice.

Stage	Name	Duration	Tool Component Activated
1	Prior Knowledge Elicitation	Week 1	SKM Initialization
2	Problem Parameter Calibration	Ongoing	Parameter Init. Engine

Stage	Name	Duration	Tool Component Activated
3	Guided Problem Resolution	Weeks 2–10	Template Repository + Feedback
4	Independent Problem Sprint	Weeks 11–14	PIE (autonomous mode)
5	Competency Assessment & SKM Update	Week 15–16	Analytics Dashboard

Table 2. Five-stage cyclical methodology for competency development.

Stage 1 — Prior Knowledge Elicitation. At the outset, students complete a diagnostic assessment designed not as a gatekeeping instrument but as a knowledge-mapping tool. The results populate the initial SKM, identifying confirmed competency components, partially mastered areas, and knowledge gaps. This stage takes approximately one academic week and is conducted entirely within the LMS using adaptive diagnostic items.

Stage 2 — Problem Parameter Calibration. The PIE uses the initial SKM to select problem templates and instantiate their variable slots with contextually appropriate values. Difficulty is calibrated using the Zone of Proximal Development (ZPD) model: each generated problem should be solvable by the student with moderate cognitive effort, requiring activation of confirmed prior knowledge plus one new inferential step. This calibration runs continuously throughout the course, updating after every completed problem session.

Stage 3 — Guided Problem Resolution. Students engage with generated problems in weekly laboratory sessions of 90 minutes. Each session begins with a brief prior knowledge activation prompt — a 5-minute micro-quiz targeting the specific knowledge components the upcoming problems will require. Students then work through 4–6 generated problems, with the feedback module providing step-level scaffolding and explaining solution strategies in terms of the prior knowledge components they mobilize. The instructor monitors the analytics dashboard in real time and intervenes with targeted mini-lectures when the dashboard reveals common error patterns.

Stage 4 — Independent Problem Sprint. In the final three weeks of the taught curriculum, the PIE operates in autonomous mode: students receive daily problem sets generated without instructor intervention. The difficulty ramp is steepened by setting the ZPD calibration parameter to require two new inferential steps rather than one. No scaffolded hints are provided; feedback is delayed until the student submits a complete solution, simulating the conditions of professional practice.

Stage 5 — Competency Assessment and SKM Update. A summative assessment conducted in the final two weeks evaluates all targeted competency components using novel problem instances generated by the PIE (ensuring that students cannot have encountered identical problems previously). Assessment results update the SKM, which is archived and passed to the instructor of the subsequent course module, creating a longitudinal competency record across the student's program of study.

4. Quantitative Model of Competency Growth

To formalize the expected competency growth under the proposed methodology, we adopt a modified logistic growth model. Let C(t) denote a student's competency score in a given domain at time t, measured on a normalized scale [0, 1]. The rate of competency growth is modeled as:

$$dC/dt = r \cdot C(t) \cdot (1 - C(t)) \cdot A(t) \quad (1)$$

where r is the intrinsic learning rate of the student, and A(t) is the prior knowledge activation coefficient at time t, defined as the proportion of prerequisite competency components confirmed in the current SKM. The term A(t) is the key contribution of the proposed methodology: by ensuring that each problem session maximizes A(t) through deliberate prior knowledge activation, the tool systematically amplifies the effective learning rate without changing r.

The cumulative competency gain over a course of duration T is thus:

$$\Delta C = \int_0^T r \cdot C(t) \cdot (1 - C(t)) \cdot A(t)\, dt \quad (2)$$

In the limit where A(t) = 1 (perfect prior knowledge activation at every session), Equation (2) reduces to the standard logistic growth integral, representing the theoretical ceiling of competency gain achievable with a given student's intrinsic learning rate. The proposed methodology aims to approach this ceiling by maximizing A(t) through systematic tool-mediated prior knowledge elicitation at the start of each problem session [2].

5. Experimental Validation

The methodology was evaluated over one full academic semester (16 weeks) at Asia International University, Bukhara, in the context of a second-year undergraduate course on algorithms and data structures — a domain in which the dependency of new knowledge on prior competencies is particularly explicit and well-structured. Two cohorts were compared:

Control group (n = 30): Traditional instruction comprising lectures, textbook exercises, and end-of-chapter tests. Problem sets were fixed and identical across all students regardless of prior knowledge profile.

Experimental group (n = 32): Instruction delivered through the five-stage cyclical methodology described above, with problems generated by a prototype competency-initializing tool implemented on the Moodle LMS platform using a custom adaptive quiz plugin integrated with a Python-based PIE backend.

Pre-test and post-test assessments measured competency across five sub-domains: algorithmic thinking, recursive problem decomposition, data structure selection, complexity analysis, and debugging. The normalized learning gain g (Hake, 1998) was computed as:

$$g = (S\_post - S\_pre) / (100 - S\_pre) \quad (3)$$

Competency Domain	Control g	Experimental g	Improvement

Competency Domain	Control g	Experimental g	Improvement
Algorithmic Thinking	0.31	0.67	+116%
Recursive Decomposition	0.24	0.61	+154%
Data Structure Selection	0.28	0.65	+132%
Complexity Analysis	0.30	0.63	+110%
Debugging	0.33	0.62	+88%
Overall Mean	0.29	0.64	+121%

Table 3. Normalized learning gain by competency domain (Hake, 1998).

The experimental group achieved a mean normalized gain of $g = 0.64$, placing it in the 'medium-high' effectiveness range and approaching the boundary of high effectiveness ($g \geq 0.70$). Crucially, the experimental group demonstrated gains across all five sub-domains, whereas the control group showed stagnation particularly in recursive decomposition and data structure selection — precisely the domains most dependent on the activation of prior knowledge from earlier course units.

Post-course surveys ($n = 32$, Likert scale 1–5) revealed that 91% of experimental group students rated the repetitive problem-generation approach as 'helpful' or 'very helpful' for consolidating prior knowledge, and 84% reported that the tool-generated problems felt more relevant to their individual level than conventional textbook exercises. Instructor workload in delivering personalized formative feedback was reduced by approximately 40%, as the automated feedback module handled first-level response analysis.

6. Implementation Recommendations. Based on the experimental pilot and subsequent analysis, the following recommendations are offered to institutions seeking to adopt the proposed methodology:

Platform selection. Open-source LMS platforms such as Moodle provide sufficient extensibility through their quiz and plugin APIs to support the PIE backend. Institutions without in-house development capacity may consider commercial adaptive learning platforms (e.g., ALEKS, Realizeit) as off-the-shelf alternatives, though these offer less flexibility in problem template design.

Faculty preparation. Instructors must be trained not only in tool operation but in the pedagogical rationale of prior knowledge activation and spaced repetition. Without this understanding, faculty tend to override the system's difficulty calibration, undermining the adaptive benefits. A minimum of 8 hours of professional development is recommended prior to first deployment.

Template authoring. The quality of the problem template repository is the primary determinant of tool effectiveness. Template authoring workshops should involve subject-matter experts who can map problem structures to specific competency components and prerequisite knowledge dependencies.

Ethical and equity considerations. Adaptive tools that personalize problem difficulty must be monitored for equity implications: students who enter with weaker prior knowledge profiles may receive systematically easier problems, potentially reinforcing rather than closing competency gaps. The PIE should be configured with a 'stretch floor' — a minimum difficulty threshold below which generated problems will not fall, regardless of the SKM state.

Conclusion. This article has proposed and empirically validated a cyclical, five-stage methodology for developing student competencies through software tools that initialize repetitive problem situations grounded in prior knowledge. The methodology integrates schema theory, spaced repetition research, and competency-based education principles into a coherent instructional design supported by a parameterized problem-generation architecture. Experimental results at Asia International University demonstrate a normalized learning gain of $g = 0.64$ across five competency domains — a 121% improvement over the conventional curriculum — with particularly strong gains in sub-domains characterized by high prior-knowledge dependency.

Future research will focus on three directions: (1) extending the competency model to include collaborative and cross-disciplinary problem types; (2) integrating large language model-based natural language generation into the PIE to produce textually varied problem narratives around the same structural template; and (3) longitudinal tracking of competency trajectories across multiple course modules to assess the long-term transfer effects of the methodology.

References

1. Ausubel, D. P. (1968). Educational Psychology: A Cognitive View. Holt, Rinehart and Winston, New York. 685 p.

2. Bjork, R. A. (1994). Memory and metamemory considerations in the training of human beings. In: Metcalfe, J. and Shimamura, A. (Eds.), Metacognition: Knowing about Knowing. MIT Press. pp. 185–205.

3. Cepeda, N. J., Pashler, H., Vul, E., Wixted, J. T., Rohrer, D. (2006). Distributed practice in verbal recall tasks: A review and quantitative synthesis. Psychological Bulletin, 132(3), pp. 354–380.

4. VanLehn, K. (2011). The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. Educational Psychologist, 46(4), pp. 197–221.

5. Hmelo-Silver, C. E. (2004). Problem-based learning: What and how do students learn? Educational Psychology Review, 16(3), pp. 235–266.

6. Hake, R. R. (1998). Interactive-engagement versus traditional methods: A six-thousand-student survey of mechanics test data for introductory physics courses. American Journal of Physics, 66(1), pp. 64–74.

7. European Commission. (2017). European Qualifications Framework for Lifelong Learning. Publications Office of the European Union, Luxembourg. 48 p.

8. Ibragimov, U. (2025). Application of the Harrington method in the food industry. Journal

of Applied Science and Social Science, 1(3), pp. 544–549.

9. Ibragimov Ulugbek Muradilloyevich. Rainbow table threat and defense mechanisms. Journal of Applied Science and Social Science. Volume 15, Issue 10, October 2025. pp. 280–287.

10. Ibragimov Ulugbek Muradilloyevich. Analysis of linear, binary and interpolation search algorithms. Journal of Multidisciplinary Sciences and Innovations. Volume 04, Issue 10, November 2025. pp. 425–428.