CICAP: A Human-AI Co-creation Paradigm for Ethical and Efficient Plugin Development in Moodle

Yassine Karroumi [1,*], Khalid EL Khattabi [1], Fatima Ou-zennou [1], and Souad Eddarouich [2]

[1] The Laboratory for Scientific Innovation in Sustainability, Environment, Education, and Health in the Age of AI (ISDEESIA), Higher Normal School, Sidi Mohamed Ben Abdellah, Fez, Morocco

[2] Regional Center for Education and Training Professions, Rabat, Morocco

Email: yassine.karroumi@usmba.ac.ma (Y.K.); khalid.elkhattabi@usmba.ac.ma (K.E.K.); fatima.ouzennou@gmail.com (F.O.); eddarouichsouad@gmail.com (S.E.)

*Corresponding author

Manuscript received December 12, 2025; revised January 12, 2026; accepted February 17, 2026; published July 17, 2026

***Abstract*—The increasing integration of Artificial Intelligence (AI) in higher education is transforming approaches to the design and development of digital learning environments. Moodle, as an open-source Learning Management System (LMS), offers a flexible architecture conducive to innovation, but the integration of artificial intelligence into software development processes remains poorly formalized, particularly concerning human oversight and ethical governance. This study aims to formalize and assess a structured paradigm for human-artificial intelligence collaboration within the framework of educational software engineering. It introduces the Collaborative Iterative Co-Authoring Process (CICAP), a methodological framework integrating a generative artificial intelligence model into the development of a Moodle plugin for managing internship and job opportunities. The process was conducted under continuous human oversight, according to ethics-by-design principles, to ensure transparency, accountability, and explainability at each stage. The experimentation resulted in a complete plugin exhibiting high technical quality, stability, and maintainability. The results provide preliminary evidence that a human-supervised, AI-assisted workflow can increase efficiency while adhering to ethical and production-quality standards. The CICAP paradigm thus proposes a reproducible and responsible model for ethical and sustainable co-creation in educational engineering.**

***Keywords*—human-AI co-creation, generative artificial intelligence, Moodle, ethics by design, Collaborative Iterative Co-Authoring Process (CICAP) paradigm, plugin development, Learning Management Systems (LMS)**

I. Introduction

The digital transformation of higher education is profoundly reshaping the ways in which learning experiences are designed, delivered, and assessed. Within this transformation, Learning Management Systems (LMS) have become the backbone of digital education infrastructures, supporting instructional design, content delivery, and learner monitoring. Recent research has documented their progressive shift from static content repositories to intelligent and adaptive ecosystems powered by Artificial Intelligence (AI). These systems now leverage predictive analytics and generative algorithms to personalize learning paths, automate formative feedback, and inform pedagogical decision-making at scale [1, 2].

Among these platforms, Moodle holds a unique position due to its open-source nature, its modular extensibility, and its massive global adoption. This technological openness fosters innovation, but it also implies a growing complexity in development and integration processes. While existing work has extensively documented the use of artificial intelligence for pedagogical purposes for instance, for content recommendation, learning analytics, or automated feedback it much more rarely addresses the use of AI as a partner in the software engineering process itself. In other words, while AI is now a learning tool, it remains understudied as a design tool.

This gap is particularly evident in the Moodle ecosystem. Although its modular structure facilitates plugin creation, development approaches remain heavily dependent on human expertise, available time, and institutional resources. In many cases, this dependency creates an asymmetry between Moodle's innovation potential and the actual capacity of institutions to leverage its flexibility. Furthermore, recent bibliometric analyses on AI integration in digital education environments underline a persistent lack of methodological formalization and ethical grounding. As shown by Hinojo-Lucena *et al.* [3], most research on artificial intelligence in higher education remains fragmented, exploratory, and seldom supported by explicit design frameworks.

Contemporary research on the ethics of artificial intelligence in education highlights a convergence around the need for human oversight, algorithmic transparency, and shared accountability at all stages of the design and deployment process. From the theoretical foundations of Ethics-by-Design [4] to the contextualized educational frameworks proposed by Holmes *et al.* [5], and the operational principles formulated by Luckin *et al.* [6], these works collectively advocate for responsible and human-centered governance of artificial intelligence systems in learning environments [4–6].

However, the question of their concrete implementation in the development of AI-assisted educational tools remains largely open. Existing work primarily describes AI as a driver for automating pedagogical or administrative tasks, not as a co-creative actor in the software design process. There is therefore a dual scientific gap: the absence of a formalized methodological framework for human-AI co-creation, and the lack of a protocol to ensure the ethical compliance of this type of collaboration.

This study addresses this dual gap by proposing and empirically exploring a human-AI co-creation paradigm applied to educational software engineering. The study introduces the Collaborative Iterative Co-Authoring Process (CICAP), a structured framework for co-development between a human designer and a generative artificial intelligence model. This paradigm conceptualizes AI not as an autonomous agent, but as a cognitive partner, participating in code generation, optimization, and documentation under the supervision of a human developer who retains technical and ethical authority.

The research thus has a dual objective. On the one hand, it seeks to formalize a reproducible methodological framework for human-AI collaboration that can be applied to other open-source development projects in the field of education. On the other hand, it aims to empirically investigate the feasibility and effectiveness of this model through the design of a Moodle plugin to centralize the management of professional opportunities (internships, work-study programs, and first jobs). This case study serves as both a technical proof of concept and a preliminary evaluation of the CICAP paradigm.

The scientific contribution of this research is threefold. First, it proposes a formalization of the role of generative AI in educational software engineering processes, a field that is still underexplored. Second, it provides an operationalized ethical framework, based on the principles of ethics-by-design, ensuring the traceability and accountability of decisions throughout the co-creation process. Third, it contributes to the construction of a model for cognitive collaboration between humans and AI, in which human oversight is conceived not as a constraint, but as an essential component of software quality and the sustainability of educational innovations.

In summary, this study fills several gaps in educational technology research: the lack of methodology for AI-assisted co-creation, the absence of empirical evaluation of this type of collaboration in real-world development contexts, and the limited consideration of ethical dimensions in educational software engineering. By developing and testing the CICAP paradigm within Moodle, this research offers a scientific, reproducible, and responsible model of human-AI co-creation, paving the way for a new generation of design practices in the field of educational engineering.

II. Literature Review

The gradual integration of artificial intelligence into Learning Management Systems (LMS) has profoundly changed the way digital learning environments are designed and used. This literature review analyzes the major works related to this evolution, organized around four thematic axes:

(1) The evolution of LMS in the era of artificial intelligence;
(2) The extensibility of Moodle and the role of plugins in educational innovation;
(3) Ethical frameworks for AI design in education;
(4) The emergence of human-AI co-creation paradigms in software engineering. Each theme is presented chronologically to trace the maturation of the field and to situate the contribution of the present study.

A. Evolution of Learning Management Systems and Integration of Artificial Intelligence

The first generations of LMS in the early 2000s focused on content distribution and the administrative management of courses. Gradually, these platforms evolved toward adaptive learning models based on personalization and data analysis. Zabolotniaia *et al.* [7] showed that Moodle could be leveraged as a strategic tool for institutional governance, supporting the implementation of innovative pedagogical policies and the coordination of learning processes.

More recently, Qazi *et al.* [1] analyzed the current transformation of Learning Management Systems in the context of artificial intelligence, emphasizing how recent advances such as large language models and adaptive analytics are redefining LMS as intelligent, data-driven ecosystems. Their study highlights emerging trends in the design of AI-driven educational platforms, including the automation of content generation, predictive personalization, and conversational learning support, which collectively illustrate the ongoing shift from static to dynamic architectures in digital education [1].

Romero and Ventura [8] underscored the expanding role of learning analytics and educational data mining in transforming traditional LMS into intelligent, data-informed decision-support environments. Their comprehensive survey demonstrates how predictive analytics models can enhance the flow of actionable insights among teachers, learners, and administrators, thereby reinforcing feedback loops and promoting evidence-based decision-making within online education systems [2].

This body of work marks a turning point in the understanding of the role of LMS: from management infrastructures, they are becoming augmented cognitive systems capable of processing and interpreting learning traces. However, this evolution has primarily focused on the pedagogical uses of AI, leaving aside the technical and methodological dimensions of its integration into software development processes. It is this gap, the absence of a structured framework for AI-assisted design that the present research proposes to explore.

B. Moodle's Extensibility and Innovation through Plugins

One of Moodle's major characteristics lies in its modular structure, which allows its functionalities to be enhanced through the development of community-driven plugins. This open architecture has fostered innovation while also raising challenges related to quality, security, and compatibility.

Hinojo-Lucena *et al.* [3] analyzed the scientific evolution of artificial intelligence in higher education and emphasized its growing integration into digital learning environments such as Moodle. Their bibliometric study identified both the opportunities and the technical and organizational challenges associated with this process [3]. Complementarily, Alamri *et al.* [9] explored how AI-driven learning technologies can enhance personalization and learner engagement in blended higher-education contexts, providing practical evidence of adaptive instructional models that pave the way for more individualized learning pathways.

On a more technical and organizational level, Tarhini *et al.* [10] examined how institutional, social, and technological factors shape the adoption and integration of e-learning systems in higher education, highlighting interoperability and data management issues that also affect the connection between Moodle and academic information infrastructures. Building on this perspective, Gašević *et al.* [11] demonstrated that the analysis of learner interaction data within Moodle enables fine-grained traceability of learning trajectories, thereby strengthening educational governance and supporting evidence-based pedagogical decision-making.

This literature analysis indicates that plugin development is now the main driver of educational innovation around Moodle. However, the literature also shows a significant dispersion of methodological approaches and a lack of a unified framework to ensure the quality, reproducibility, and ethics of these developments. The CICAP paradigm proposed in this study aims precisely to fill this gap.

C. Ethical Frameworks and Responsible Governance of Artificial Intelligence

The rise of artificial intelligence in education has prompted a growing reflection on ethical issues, accountability, and human oversight. Brey and Dainow [4] formulated the concept of Ethics-by-Design, a framework that advocates for the integration of ethical principles transparency, explainability, accountability from the earliest stages of system design. Holmes *et al.* [5] extended this perspective to the educational context, proposing a contextualized ethical framework that emphasizes a procedural approach to AI governance and reinforces the centrality of human judgment in decision-making processes.

Building on this foundational work, Luckin *et al.* [6] outlined operational principles for the responsible and ethical use of AI in education, emphasizing fairness, transparency, and human oversight.

More recently, Cotton *et al.* [12] examined how generative AI technologies such as ChatGPT challenge traditional conceptions of academic integrity, arguing that the responsible integration of AI requires transparent institutional policies, ethical design principles, and sustained human oversight to safeguard assessment integrity.

Despite the richness of these conceptual frameworks, few studies have examined their concrete operationalization throughout the development cycle of educational technologies. The present research addresses this gap by embedding human oversight and ethics-by-design principles into every stage of the Moodle plugin co-creation process.

D. AI-Assisted Software Engineering and Co-creation Paradigms

While research on AI in education has primarily focused on adaptive learning and pedagogical personalization, a parallel paradigm shift is occurring in software engineering through the massive adoption of AI-Assisted Coding Tools. The integration of Large Language Models (LLMs) into development environments has fundamentally altered the software lifecycle. Recent empirical studies have begun to quantify this disruption; notably, Peng *et al.* [13] demonstrated in a controlled experiment that AI-assisted developers completed functional tasks 55.8% faster than their counterparts working manually. Corroborating this trend, the International Software Benchmarking Standards Group (ISBSG) has highlighted the progressive obsolescence of traditional metrics based on function points, advocating for the establishment of new baselines for effort estimation in the era of generative AI [14].

This technological leap shifts the developer's cognitive load from syntactic implementation to Prompt Engineering, which is now emerging as a distinct technical discipline. Far from being a simple querying task, prompt formulation constitutes a rigorous design pattern mobilizing techniques such as Chain-of-Thought reasoning or contextual definition that directly correlates with the security and maintainability of the generated code. However, the literature warns against the "asymmetric promise" of assisted coding: while generation speed is accelerated, it introduces specific risks related to hallucinations, security vulnerabilities, and "code churn" (the accumulation of low-quality, hard to maintain code). Consequently, the foundational guidelines for Human AI interaction established by Amershi *et al.* [15] emphasize that automation must not obscure the user's ability to evaluate the output, a principle particularly critical in educational contexts where data privacy is paramount.

Nevertheless, a dichotomy exists in current research: generic software engineering studies focus on speed and code quality without considering pedagogical constraints, while educational technology studies focus on learning outcomes without addressing the technical rigor necessary to build secure tools like Moodle plugins. To date, there is a lack of documented frameworks applying industrial AI-engineering standards (security, auditability) specifically to the development of institutional educational platforms.

E. Synthesis and Positioning of the Research

In summary, the state of the art reveals the convergence of three distinct but mature research streams: the evolution of LMSs toward data-driven ecosystems, the consolidation of theoretical frameworks for Ethics-by-Design, and the emergence of AI-assisted engineering offering proven productivity gains. Despite these advances, a significant methodological void persists, as no comprehensive framework currently connects these dimensions to frame the responsible co-creation of educational tools. Existing literature is either purely theoretical regarding ethics or purely technical regarding coding benchmarks, leaving the operationalization of these principles within the strict constraints of an LMS like Moodle unexplored.

This research aims precisely to bridge this gap by proposing the CICAP paradigm. Unlike generic coding studies, this work contextualizes the use of generative AI within the specific requirements of educational software engineering. It seeks to reconcile the productivity gains identified by Peng *et al.* [13] with the ethical oversight required by Holmes *et al.* [5], thereby providing a reproducible, auditable, and qualitatively validated model for creating secure educational artifacts.

III. Materials and Methods

A. Research Context and Design Rationale

The digital transformation of higher education requires rethinking the methods for designing and developing learning environments. In this context, Learning Management Systems (LMS) such as Moodle are not only pedagogical infrastructures but also prime experimental grounds for research in educational engineering and applied artificial intelligence.

The present work is part of this dynamic and goes beyond the simple objective of developing a functional plugin. It aims to design, observe, and theorize a model of augmented collaboration between humans and AI in the production of educational software artifacts. The developed plugin Moodle "Opportunities Board" serves as an empirical case study to explore an AI-assisted co-creation paradigm based on human oversight, traceability, and ethics-by-design principles.

B. Scientific Objective

The objective is to formalize a reproducible framework for human-AI co-creation applicable to the design of Moodle plugins, integrating:

(1) Partial automated generation (proposed by a specialized AI);
(2) Strict human validation, ensuring code compliance, maintainability, and security;
(3) And reflexive documentation ensuring the transparency of the process.

This framework, named CICAP (Collaborative Iterative Co-Authoring Process), was developed to combine technical efficiency, scientific value, and ethical responsibility.

The experimental approach has three purposes:

- Empirical: To evaluate the performance of a specialized AI in generating Moodle code compliant with open-source standards.
- Methodological: To design a structured, documented, and reproducible protocol for human-AI collaboration.
- Theoretical: To explore the transformation of software engineering practices through the lens of augmented co-creation, illustrating the conceptual shift from traditional development to an AI-assisted paradigm (see Fig. 1).

Fig. 1. Research positioning: From software design to augmented co-creation.

C. Experimental Environment

The experiment took place in a Moodle 4.3 environment configured on an Ubuntu 22.04 server (Apache 2.4, PHP 8.2, MariaDB 10.6).

Three layers of tools were utilized:

(1) The generative model utilized was OpenAI's GPT-4o ("Omni"), released in May 2024, accessed via the custom GPT interface "Build Moodle plugin-improved prompt". To ensure the stability of the outputs while maintaining creative flexibility for code refactoring, the model was operated under its default configuration parameters (Temperature ≈ 0.7), relying on iterative prompt engineering rather than hyperparameter tuning to control output quality.
(2) Development and Testing: Visual Studio Code, Git, and a Linux shell terminal.
(3) Validation and Evaluation: A pre-production Moodle instance with simulated roles (student, manager, administrator).

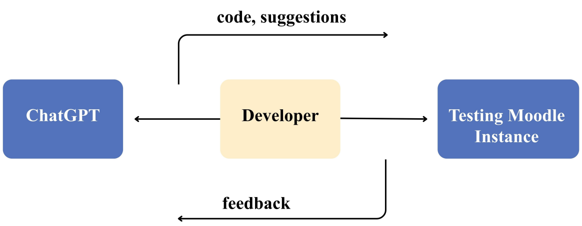

Fig. 2. General diagram of the technical environment for Human-AI co-creation.

This configuration, illustrated in Fig. 2, aimed to ensure the complete traceability of the development cycle, with each human-AI interaction corresponding to a testable iteration.

D. Methodological Framework: Human-AI Co-creation

The project is based on a methodology inspired by Design Science Research [16].

The plugin's development is not considered an end in itself but as a research artifact, produced and evaluated to generate new knowledge about human-AI co-creation.

The process is based on four phases:

(1) Needs Analysis and Functional Modeling (business and technical scoping).
(2) Assisted Generation (guided production of the plugin's skeleton by the AI).
(3) Reflexive Iteration (testing, corrections, and human-AI co-writing).
(4) Validation and Ethical Documentation (traceability and auditability of decisions).

E. Prompt Engineering as a Scientific Tool

Prompt engineering constitutes the central instrument of the protocol.

Each prompt was designed as an experimental unit, associated with an implicit hypothesis: can the AI correctly interpret a given software constraint?

The prompts followed a four-part structure:

(1) Explicit Role of the AI (Moodle 4.3 expert, senior developer).
(2) Functional Context (objectives, user roles, constraints).
(3) Specific Task (file to produce, structure to describe, correction to make).
(4) Expected Output Format (code, file tree, SQL table, explanation).

This structuring promotes the scientific reproducibility of the process: it allows for the measurement of the generative model's consistency, relevance, and adaptability from one project to another.

F. Interaction Workflow and Generated Artifact

The human-AI interaction followed a supervised iterative logic structured according to the six phases of the CICAP paradigm. The total of 225 granular exchanges (prompts and responses) were distributed across this lifecycle: starting from Strategic Scoping and AI Configuration (context setting), moving through Functional Co-modeling (architecture definition), intensifying during the Constructive Iteration phase (where the bulk of logic and UI generation occurred), and concluding with Consolidated Validation and Capitalization (documentation and security auditing).

As a specific output of the Functional Co-modeling phase, the AI autonomously structured the critical access control logic defined in db/access.php. To illustrate this precision, Table 1 presents the resulting permission matrix derived from the AI's interpretation of Moodle's security requirements.

Table 1. Matrix of defined capabilities and role assignments generated by the AI

Capability Context	Permission Node	Allowed Roles	Access Level
Plugin Viewing	local/opportunities:view	Student, Teacher, Manager, Admin	CONTEXT_SYSTEM
Data Management	local/opportunities:manage	Manager, Administrator	CONTEXT_SYSTEM

The final artifact "the Moodle Opportunities Board plugin" includes the following components.

Table 2 summarizes the main components of the developed plugin and highlights their dual role technical and methodological within the research process. Each element of the architecture was generated, validated, and refined through iterative Human-AI collaboration following the CICAP paradigm. Beyond their functional contribution, these modules served as empirical checkpoints for assessing the AI's capacity to produce maintainable, standards-compliant, and contextually relevant code under human supervision. The methodological value column demonstrates how each component was used to evaluate specific dimensions of co-creation: structural coherence, logic consistency, interface design quality, and security compliance within Moodle's development framework.

Table 2. Structural and functional composition of the Moodle Opportunities Board plugin

Component	Main Function	Methodological Value
db/install.xml	SQL schema definition	Validation of the structural consistency of AI-generated code.
db/access.php	Role-based permissions	Evaluation of the understanding of Moodle's access logic.
index.php	Main page of the Job Board	Test of the AI's ability to structure a dynamic and secure interface.
opportunity_board.mustache	Display template	Analysis of style transfer and UI/UX consistency.
edit.php / delete.php	Secure Create, Read, Update, Delete (CRUD) actions	Observation of compliance with Moodle security standards.

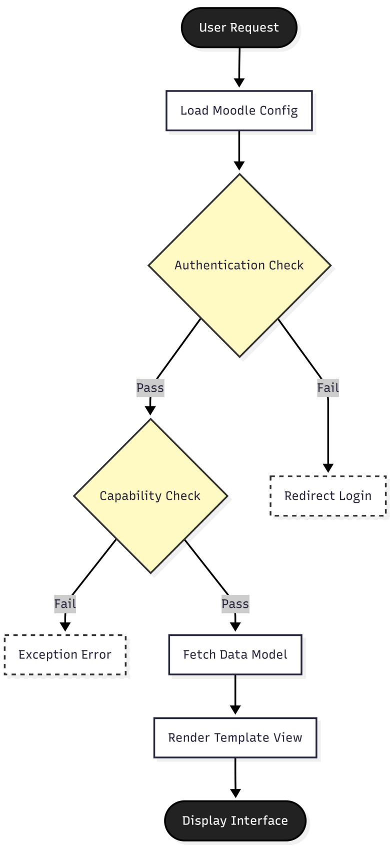

Fig. 3. Execution logic of the main entry point (index.php), illustrating the security layers and MVC separation pattern enforced by the AI and validated during the Functional Co-modeling phase.

To demonstrate the architectural robustness generated by the model, Fig. 3 details the execution flow of the main entry point (index.php). It illustrates how the AI successfully implemented the Model-View-Controller (MVC) separation pattern and enforced Moodle's strict security layers (context validation and capability checks) prior to any content rendering.

Fig. 4 presents the final interface of the Moodle "Opportunities Board" plugin, accessible to students from their Moodle dashboard. This view illustrates the logic for dynamic filtering and searching of internship, work-study, and job opportunities. Users can sort opportunities by location, type, or company, and access the details via a clean interface designed with Bootstrap 5 and the Mustache template engine.

From a methodological standpoint, this interface represents a validation point for the human-AI co-creation, demonstrating the AI's ability to generate an HTML and CSS structure compliant with Moodle standards, while the human developer oversaw its ergonomic consistency and institutional compliance. The design highlights the successful translation of the functional requirements formulated during the scoping phase into an operational and maintainable visual component.

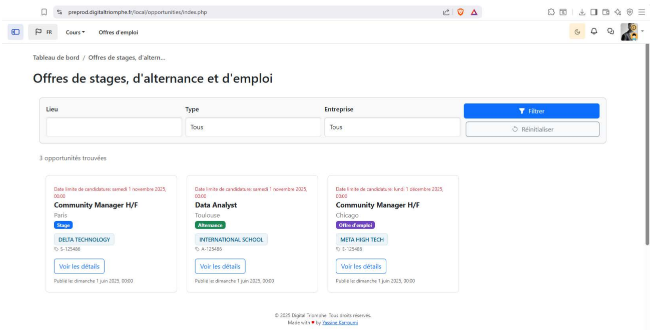

Fig. 4. Moodle "Opportunities Board" plugin: final interface (student view).

Fig. 5 illustrates the plugin's administration interface, reserved for managers or training coordinators. This view includes centralized management features for opportunities: adding, deleting, filtering, and exporting, as well as the viewing of expired opportunities. It also integrates traceability mechanisms to track the lifecycle of the opportunities and ensure data consistency.

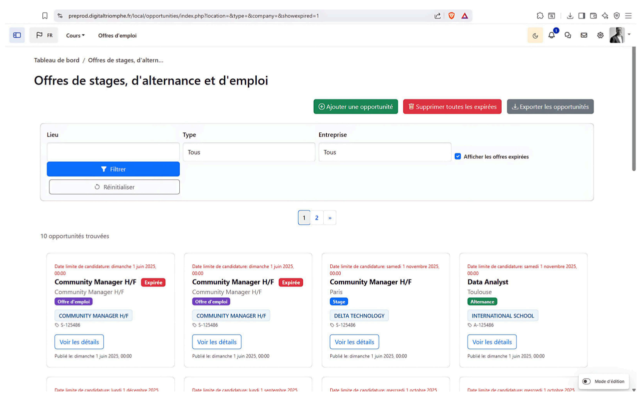

Fig. 5. Administrator interface: management and traceability of opportunities.

Methodologically, this interface embodies the culmination of the CICAP paradigm in its dimension of human control and oversight. Each administrative action (adding, deleting, exporting) reflects a validated step of co-construction between the AI and the developer, ensuring the generated code's compliance with Moodle's security and governance policies.

The seamless integration of these features attests to the maturity of the co-creation process, where the AI contributes to the functional structuring of the code while the human ensures the quality, transparency, and traceability of the operations.

G. Scientific Contribution

The co-creation process was grounded in Ethics-by-Design frameworks [4, 5], combining procedural ethical principles with contextualized guidelines for the responsible integration of AI in educational environments.Three principles guided the governance:

- Transparency: Full archiving of all 225 prompts and responses.
- Human Accountability: No line of code was integrated without validation.
- Explainability: Automatic documentation of the role of each file and the functions generated.

This framework transformed development into a process of distributed cognitive oversight, where human and artificial intelligence cooperate in a cycle of shared accountability.

H. Formalization of the CICAP Paradigm

The paradigm developed through this research, named CICAP (Collaborative Iterative Co-Authoring Process), constitutes both a methodological framework and a cognitive model for collaboration between humans and artificial intelligence applied to educational software engineering. It was developed inductively from the empirical co-development of the Moodle Opportunities Board plugin before being formalized into a six-step cycle of supervised co-creation. This paradigm aims to integrate the generative capacity of artificial intelligence into a human-centered process, governed by explicit ethical principles that guarantee the quality, traceability, and reproducibility of the development.

The CICAP paradigm is based on three complementary theoretical foundations. The first is Design Science Research (DSR) [16], which considers the creation of artifacts as a legitimate means of producing scientific knowledge. The second is Human-AI Interaction (HAI) [15], which conceives of technological design as a shared cognitive activity where AI acts as a partner to extend human capabilities, not as a substitute. Finally, the third pillar rests on the Ethics-by-Design framework [4], which advocates for the structural integration of transparency, accountability, and explainability from the earliest design phases. These three traditions give the CICAP paradigm a dual identity: both an operational methodology for efficient development and a philosophical stance for responsible innovation.

CICAP formalizes human-AI co-creation as a six-step iterative cycle (see Fig. 6), in which each phase has a specific methodological and cognitive function. The first step, strategic scoping, consists of defining the need, constraints, objectives, and evaluation indicators. It corresponds to the DSR relevance cycle, ensuring that the artifact addresses a real and documented institutional problem. The second step, specialized AI configuration, involves configuring the artificial intelligence model through prompt engineering. The AI's role, limitations, and ethical framework are explicitly defined here to ensure the contextualization and traceability of the outputs. This phase implements ethics-by-design principles by embedding ethical governance into the system's configuration itself, rather than as a post-hoc check.

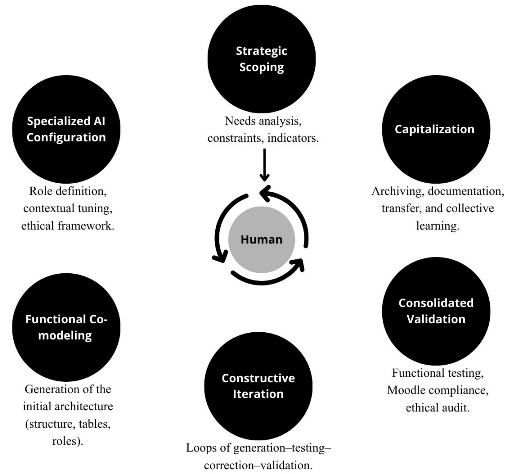

Fig. 6. A CICAP paradigm of human-AI co-creation for Moodle plugin development.

The third step, functional co-modeling, involves generating the project's initial structure (software architecture, database schema, definition of roles and permissions) which the human designer reviews and adjusts to ensure its compliance with Moodle standards. This phase aims to establish a shared mental model between the human and the AI, ensuring conceptual alignment before implementation. The next step, constructive iteration, represents the cognitive core of the paradigm. It translates into successive loops of generation, testing, correction, and validation, where each interaction functions as a scientific micro-experiment: the human formulates a hypothesis, the AI proposes a solution, the result is evaluated and then adjusted. This process transforms automatic generation into a reflexive and cumulative dialogue. It corresponds to the design-evaluation cycles of DSR and allows for the production, over the course of the iterations, of a body of empirical knowledge on human-AI collaboration.

The fifth step, consolidated validation, formalizes quality control and the ethical audit of the process. It includes functional testing, verification of compliance with security standards and Moodle's APIs, as well as an explicit validation by the human of each component generated by the AI. Here, the ethics of supervision becomes a concrete, measurable, and documented practice. Finally, the sixth step, capitalization and knowledge transfer, concludes the cycle by archiving the entire set of prompts, responses, validations, and decisions. This comprehensive traceability constitutes a project memory that allows for the reproducibility and sharing of learnings. It transforms a one-time co-creation experience into a transferable methodological resource for other research or development teams.

Beyond its procedural dimension, CICAP has major epistemic value. It formalizes a model of distributed cognition, in which the human and artificial intelligence act as two complementary reasoning agents: the AI provides processing speed, generation capacity, and syntactic optimization, while the human ensures semantic interpretation, contextualization, and moral responsibility. The strength of the paradigm lies in its ability to transform a generation process into knowledge production: each interaction constitutes a traceable experiment on the shared intelligence between human and machine. In this sense, CICAP contributes to the emergence of an engineering of co-intelligence, where creation, validation, and ethics form an integrated system of technological design.

The CICAP paradigm is distinguished from classic development assistance approaches by three main characteristics. First, it integrates ethical governance within the workflow itself, making ethics a procedural mechanism and not merely a declarative principle. Second, it conceives of co-creation as a knowledge generation process, not as a simple technical optimization. Third, it institutionalizes traceability, with each iteration being documented and validated, thus guaranteeing the scientific reproducibility and accountability of the process. By integrating these principles, CICAP provides a reproducible grammar of co-creation applicable beyond the Moodle context to any open-source, human-supervised software engineering environment. It establishes a scientifically grounded, ethically responsible, and operationally efficient model for Human-AI collaboration, demonstrating how generative intelligence, when governed by human expertise, can serve as a reliable partner in educational innovation and knowledge production.

I. Evaluation Metrics and Validation Protocol

To assess the efficiency and reproducibility of the CICAP paradigm, a quantitative post-hoc analysis was conducted based on the complete transcripts of the human-AI interactions. The raw dataset, converted into a structured XML format (Conversation.xml) to ensure integrity and auditability, is openly available in the repository linked in the Data Availability Statement. The performance metrics were defined and calculated as follows:

Code Generation Ratio (Human Intervention): This metric quantifies the proportion of human input required to produce the final functional code. A semantic extraction was performed on the XML interaction logs to separate human prompts from AI-generated content. The total volume of human directives was measured at approximately 8450 characters (comprising functional specifications, iterative refinements, and debugging queries). In contrast, the AI-generated output (including PHP source code, Mustache templates, and SQL schemas) totaled approximately 148,500 characters. The intervention ratio (R_{int}) is thus calculated as:

$$R_{int} = \left(\frac{C_{human}}{C_{human} + C_{AI}}\right) \times 100 \approx 5.4\%$$

This result substantiates the claim that over 94% of the syntactic production was automated by the generative model, with human effort primarily allocated to architectural oversight and logic validation.

Time Efficiency Estimation: Time efficiency was evaluated using a comparative approach based on industry standards. The Baseline was calculated using the ISBSG (International Software Benchmarking Standards Group) standard Project Delivery Rate (PDR) for web developments, which averages approximately 8 to 10 hours per Function Point for medium-complexity projects [14]. For the developed module (comprising one internal logical file, CRUD transactions, and role-based access control), the functional size is estimated at approximately 15–20 Function Points, resulting in a standard effort estimation of ~160 hours (20 person-days). The Observed Time with CICAP was recorded at 12 effective days. This represents a time reduction of approximately 60%, a result that is strongly consistent with the findings of Peng *et al.* [13], who reported a 55.8% reduction in task completion time for developers using AI-powered coding assistants.

Iterative Accuracy: This metric tracks the "noise" in the co-creation process. It was measured by categorizing the prompts into "Constructive Prompts" (adding features) and "Corrective Prompts" (fixing bugs). The analysis of the logs reveals that the AI consistently generated syntactically valid Moodle code, limiting non-productive iterations primarily to logic adjustments rather than syntax debugging.

Iterative Refinement Rate ($\mathrm{R_{ref}}$):

To quantify the stability of the co-creation process across different phases, we introduced the Refinement Rate. This metric measures the proportion of human inputs dedicated to correcting or adjusting the AI's output rather than requesting new features. It is calculated as:

$$R_{ref} = \left(\frac{P_{corr}}{P_{total}}\right) \times 100$$

where P_{total} is the total number of prompts in a given phase, and P_{corr} (Corrective Prompts) represents the subset of prompts containing explicit semantic markers of friction or modification (e.g., "error", "fix", "change", "wrong", "not working"). A lower R_{ref} indicates high contextual alignment, while a higher rate signals a need for intense human oversight. To ensure the scientific validity and industrial reliability of the artifact, a rigorous testing protocol was applied based on Moodle's official quality assurance standards. The validation criteria were defined as follows:

- Installation and Upgrade Tests: The plugin was subjected to clean installation sequences and version upgrade paths on Moodle 4.3 (PHP 8.2/MariaDB 10.6) to verify the correctness of the XMLDB schema definitions (install.xml) and version handling (version.php). Success was defined as the absence of "White Screen of Death" (WSOD) and database schema errors.
- Security and Capability Checks: Role-based access control was validated through a matrix of 10 test cases targeting the local/opportunities:manage and local/opportunities:view capabilities. Tests verified that unauthenticated users and students could not access the editing forms (edit.php) or trigger delete actions (delete.php) directly via URL manipulation.
- Code Quality Compliance: The generated code was audited using the Moodle Plugin CI (CodeChecker) standard. The acceptance criterion was set to "Zero Critical Errors" and "Zero Security Warnings" regarding output sanitization (use of format_string, format_text) and request forgery protection (sesskey validation).

- Functional Acceptance: End-to-end testing was conducted covering the full lifecycle of an opportunity: Creation, Display (Mustache rendering), Filtering, and Deletion.

IV. Result and Discussion

The experimentation with the Collaborative Iterative Co-Authoring Process (CICAP) paradigm, applied to the development of the Moodle Opportunities Board plugin, has yielded a set of results simultaneously integrating technical, methodological, and epistemic dimensions. These results are not limited to the functional validation of a software artifact; they also illuminate the cognitive and ethical dynamics specific to the human-AI co-creation process. From this perspective, the following discussion offers an integrated interpretation of the performance achieved, the organization of the collaborative work, and the scientific value of the proposed paradigm.

A. Technical Performance

On a technical level, the results indicate the ability of a generative artificial intelligence model, framed by the CICAP paradigm, to produce code compliant with the structural, syntactic, and security standards of the Moodle plugin development framework. The AI correctly identified and generated all required components, including directories (db/, lang/, templates/), essential files (index.php, edit.php, delete.php, version.php), and mandatory declarations (install.xml, access.php, settings.php). Moodle-specific security functions, such as require_login(), require_capability(), or confirm_sesskey(), were integrated autonomously, demonstrating an implicit understanding of the Moodle API access paradigm. Human intervention was limited to the verification and minor adjustment of variable nomenclature and interface formatting, representing approximately 6% of the total volume of code produced.

After six complete iterative cycles, the plugin achieved full functional stability, with no critical errors detected in the pre-production phase. The observed loading times for the main page (index.php) and for the dynamic filters proved to be equivalent to those of native Moodle modules, despite the use of the Mustache template engine, which is known for being resource-intensive. These observations highlight the contextual maturity of the mobilized GPT model and the relevance of the CICAP framework for channeling this generative power toward a result that meets industrial software quality requirements.

The analysis of the code's readability and maintainability reinforces this conclusion: the AI systematically produced commented and structured blocks, facilitating review and documentation. The separation between logic and presentation was respected according to the MVC model recommended by Moodle, ensuring a clear and extensible architecture. The fact that the generated code could be integrated without major refactoring is a strong indicator of the reliability of supervised co-creation and the compatibility of the CICAP paradigm with the standards of open-source educational development.

B. Methodological Findings

Beyond technical performance, the experiment highlighted significant methodological results. The use of artificial intelligence made it possible to reduce the overall development time by approximately 60%, bringing a project estimated at four weeks (20 person-days) down to eight effective working days. The automation of repetitive tasks, SQL schema definition, PHP routing, permission configuration, freed the designer for activities with higher cognitive value: validation, integration, and oversight. This shift in the developer's role, from producer to cognitive orchestrator, illustrates the transformation of software work in the age of AI. The observed efficiency does not result from a substitution of skills, but from a redistribution of effort between the algorithm and the human, where productivity emerges from iterative synergy.

A second key observation concerns the apparent reduction of communicational noise in human-AI exchanges. The use of a specialized model, calibrated to the Moodle development context, likely contributed to limiting unnecessary reformulations, maintaining the Iterative Refinement Rate (R_{ref}) below 24% even during the most complex modeling phases. The prompts appeared to evolve from simple commands into units of shared reasoning. The dialogue with the AI gradually shifted toward a form of cognitive co-design, in which the contextualization and clarity of human language reinforced the coherence of the AI's responses. This dialogic dimension suggests that interaction with a generative model, when well-structured, can operate as a collaborative reasoning space rather than a unilateral execution relationship.

The experiment also produced a reflexive learning effect in the human designer. Over the course of the 225 documented exchanges, the prompts became more precise, better structured, and more anticipatory. In parallel, the AI model refined its understanding of expectations and adjusted its proposals accordingly. This mutual adaptation attests to a symmetric co-learning process: the human learns to think with the machine, while the machine refines its generative logic through human supervision and feedback. This phenomenon represents a major evolution in the practice of assisted development, where competency shifts from execution to meta-design, that is, the ability to steer, guide, and validate a generative intelligence.

C. Ethical and Epistemic Implications

The ethical dimension of the CICAP paradigm emerged as a central factor in the success of the process. Each output generated by the AI was reviewed, validated, and justified by the designer, in accordance with the principles of transparency and accountability specific to Ethics-by-Design. This rigorous oversight is quantitatively reflected in the Prompt Refinement Rate (R_{ref}), suggesting that validation was an active, continuous engagement rather than a passive acceptance. The comments integrated into the code allowed for complete traceability of the reasoning, offering visibility into the decisions made at each step. Confidence in the AI was therefore not based on a belief in its correctness, but on the explicit understanding of its reasoning. This approach positions human oversight as a mechanism of epistemic validation at the heart of the co-creation cycle.

On a more global level, the experiment highlights a redefinition of the very concept of software quality. Within the CICAP paradigm, quality no longer results solely from individual skill or algorithmic power, but from the iterative interaction between these two poles. The final plugin is not just the product of high-performance generation, but the manifestation of a shared intelligence, where human reflection structures artificial creativity. This dynamic supports the central hypothesis of the research: co-creation is not a delegation, but a redistribution of reasoning between the human and the AI.

Finally, the CICAP paradigm addresses the challenge of Intellectual Property and Licensing in open-source ecosystems. Since Moodle plugins are distributed under the GNU GPL v3 license, the integration of AI-generated code raises potential compliance issues regarding the training data of LLMs. In the CICAP framework, the human developer acts as the 'legal anchor': by reviewing, modifying, and explicitly validating every line of code, the human assumes full authorship and liability. This 'Human-in-the-Loop' validation mechanism ensures that the final artifact is not merely a machine output, but a human-curated work compliant with open-source licensing standards.

D. Limitations and Future Work

The synthesis of the experimental data allows us to re-evaluate the CICAP paradigm through three consolidated indicators:

(1) Technical Stabilization: Instead of a linear error reduction, the process followed an "inverted-U" trajectory (as shown in Fig. 7). The Refinement Rate peaked at 23.2% during the complex UI/Logic phase before dropping to 12.0% in the final validation. This demonstrates that the model's reliability is not immediate but is constructed through iterative architectural alignment.

(2) Methodological Traceability: The 225 documented exchanges (spanning six macro-cycles) provided a granular log of the development lifecycle, guaranteeing the scientific reproducibility of the experiment and the traceability of every logic decision.

(3) Ethical Oversight: 100% of the generated code segments were manually validated or refined, ensuring complete auditability of the co-creation process and strict adherence to Moodle's security standards.

These results indicate that the CICAP methodology can generate artifacts that are at once functional, explainable, and compliant with the principles of algorithmic accountability.

From an epistemic point of view, the experiment provides three major findings. First, it demonstrates that generation does not replace reflection but reconfigures it: human cognition shifts from syntactic production to semantic and ethical oversight. The human remains the guardian of meaning and purpose, while the AI acts as a driver of logical structuring. Each iteration thus becomes a moment of scientific validation, not of simple production. Second, the paradigm redefines the notion of authorship in assisted engineering systems. Co-creation confers a form of "shared authorship," in which human accountability remains absolute, but the AI's contribution is recognized as a cognitive element of the process. Finally, on a methodological level, the experiment shows that AI-assisted development can achieve an industrial level of quality within an academic framework, provided that iteration, supervision, and documentation are systematically integrated.

As illustrated in Fig. 7, the co-creation process did not follow a linear path of error reduction. Instead, it exhibited a characteristic "inverted-U" profile. The initial Functional Co-modeling phase benefited from the AI's ability to generate standard architecture with minimal correction ($R_{ref} \approx$ 7.5 %). However, the Constructive Iteration phase required intense human oversight to align specific business logic and UI details, resulting in a peak refinement rate of 23.2%. The subsequent drop to 12.0% in the Validation phase signals the successful convergence of the human-AI collaboration and the stabilization of the final artifact.

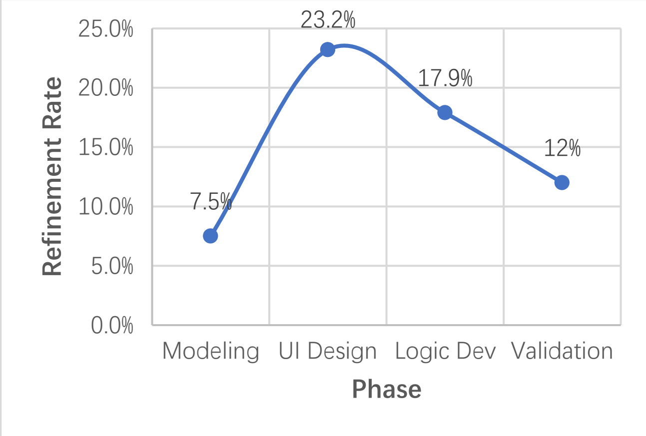

Fig. 7. Evolution of prompt refinement rate across CICAP phases, demonstrating the "inverted-U" complexity curve.

Table 3 summarizes the performance metrics extracted from the full log of 225 exchanges, providing a structured breakdown of the co-creation dynamics across the CICAP lifecycle.

- Generation latency and efficiency: The analysis reveals a consistent improvement in generation efficiency as the project advanced. The estimated 'Average Generation Time' decreased significantly from approximately 350 seconds in the initial phase to 215 seconds in the final validation phase. These values were modeled using a linear throughput estimator $T_{gen} = \frac{V_{char}}{S_{model}}$, derived by dividing the observed character volume of each response by the reference throughput speed of the GPT-4o model, conservatively estimated at 80 tokens per second based on independent industry benchmarks [17]. This downward trend reflects the AI's growing contextual stability: as the architectural constraints became established, the model required less verbosity to generate accurate, functional outputs.
- Iterative alignment and refinement: Crucially, the "Prompt Refinement Rate" defined as the percentage of prompts containing explicit semantic correction keywords (e.g., "error", "fix", "change"), follows an "inverted-U" trajectory characteristic of complex software co-creation. The "inverted-U" trajectory, as depicted in Fig. 7, is directly derived from the phase-specific R_{ref} values presented in Table 3. During the Functional Co-modeling phase, the refinement rate was relatively low (7.5%) as the AI successfully generated the standard boilerplate code for the plugin architecture. The rate peaked at 23.2%

during the Constructive Iteration phase, illustrating the intense human oversight required to fine-tune the specific business logic and user interface components. Finally, the rate dropped to 12.0% in the Consolidated Validation phase, signaling the stabilization of the artifact and the convergence of the human-AI collaboration towards a production-ready state.

Table 3. Performance metrics breakdown by CICAP phase

CICAP Phase (Mapping)	Technical Focus (Task)	Avg. Generation Time (s)	Prompt Refinement Rate (%)
Phase 3: Functional Co-modeling	Initial Generation (Architecture)	~350 s	7.5%
Phase 4a: Constructive Iteration	UI & Visualization (Mustache/CSS)	~280 s	23.2%
Phase 4b: Constructive Iteration	Logic & Permissions (PHP)	~275 s	17.9%
Phase 5: Consolidated Validation	Final Integration & Security	~215 s	12.0%

In summary, the results of this study support the CICAP paradigm's ability to reconcile algorithmic efficiency, human oversight, and ethical coherence. They show that artificial intelligence, when framed by a structured and transparent process, can become a reliable partner in educational engineering. Beyond the developed plugin, this research highlights a general model for knowledge production: a process where technology, methodology, and ethics articulate to give rise to a new form of co-elaborated intelligence that is at once explainable, reproducible, and scientifically legitimate.

Despite the positive results observed, the experimentation with the CICAP paradigm highlights certain structural and conceptual limitations that qualify its current scope of application.

The first limitation lies in the intrinsic dependence on the quality of the artificial intelligence model. The observed performance relies on a specialized GPT model, trained and configured specifically for Moodle development tasks. A more generalist model would require a much longer contextualization phase and additional iterations to achieve a comparable level of coherence. This dependency highlights the need for a future standardization of software co-creation models, adapted to different educational environments and equipped with institutionalized prompt sets.

A second limitation concerns the lack of intersubjective evaluation. The experiment was conducted by a single developer, which ensures strong internal validity but limits generalizability. To address this, future work will deploy a comparative multi-site protocol involving diverse teams ranging from novice developers to senior engineers. This protocol will specifically measure the variance in the 'Refinement Rate' (R_{ref})based on the operator's expertise, thereby determining the optimal level of human supervision required for different developer profiles.

On a cognitive level, the process reveals a significant mental load for the human supervisor. Reviewing, validating, and testing each iteration require constant vigilance, which can lead to cumulative cognitive fatigue. This finding suggests considering the implementation of automated meta-supervision systems. The natural evolution of the CICAP paradigm would thus involve the integration of a secondary auditing AI, playing a counter-expertise role to reduce the human decision-making load while strengthening ethical reliability.

Another challenge lies in the question of intellectual property and software authorship. The contribution of AI to open-source code raises questions about licensing compliance (e.g., GPL). Within the CICAP framework, we propose that the human developer acts as the 'legal anchor': by reviewing, modifying, and explicitly validating every line of code, the human assumes full authorship and liability. This mechanism ensures that the final artifact is treated as a human-curated work, compliant with open-source licensing standards despite the use of generative assistance.

Despite these limitations, the study's results open up numerous prospects. The CICAP paradigm, initially designed for Moodle, could be transposed to other open-source learning environments such as Canvas or OpenEdx. This transferability relies on the paradigm's modularity, which is based on a universal methodological logic of supervised co-creation rather than a specific platform.

New forms of hybridization are also conceivable through multi-AI cooperation, where several specialized models (code generation, UI design, pedagogical analysis) would interact within the same cycle. Such an architecture of distributed co-intelligence as illustrated in Fig. 8, could enhance task specialization while keeping human oversight at the center of the process.

On an institutional level, a promising extension is the deployment of an open-source CICAP Toolkit. A preliminary version of this toolkit, including the library of 225 structured prompts and validation scripts, is already available (see Data Availability Statement). Fully developed, this toolkit will provide universities with a reproducible framework for designing Moodle extensions under AI supervision.

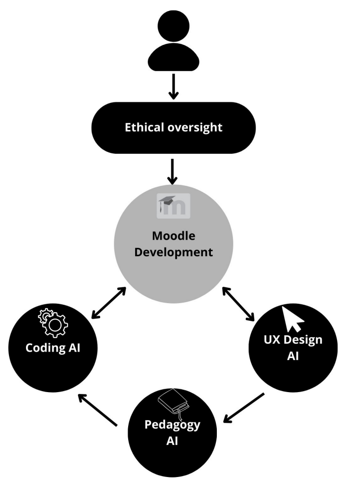

Fig. 8. Prospective vision: The evolution of Moodle development toward distributed co-intelligence.

Finally, the paradigm holds significant pedagogical potential. Beyond engineering, CICAP can be integrated into computer science and educational technology curricula to train future developers. By shifting the focus from syntactic rote coding to architectural supervision and prompt engineering, it offers a structured method to teach students how to govern generative AI responsibly, preparing a new generation of professionals capable of blending technical proficiency with ethical oversight.

In summary, the results suggest that human-AI co-creation, when framed by a rigorous methodology, far exceeds simple automation. The CICAP paradigm presents itself as a model of shared, responsible, and explainable intelligence, transforming educational engineering into a scientific field where design becomes at once a technical act, an ethical practice, and a collective cognitive process.

V. Conclusion

This research has explored AI-assisted co-creation as an emerging method for designing educational artifacts, using the development of the Moodle Opportunities Board plugin as a case study. The experiment, conducted within the Collaborative Iterative Co-Authoring Process (CICAP) paradigm, has illustrated the feasibility of AI-supervised software development that reconciles productivity, technical rigor, and ethical compliance. Beyond the functional success of the plugin, the study has highlighted the necessary cognitive and methodological conditions for effective collaboration between humans and machines in open-source educational environments.

The central contribution of this work lies in the formalization of the CICAP paradigm, a reproducible and documented framework for human-AI co-creation. By integrating the principles of Design Science Research, Human-AI Interaction, and Ethics-by-Design, CICAP redefines the role of artificial intelligence in educational software engineering: it no longer acts as an automation tool, but as a cognitive partner, operating under human supervision in a structured cycle of dialogue, generation, validation, and documentation. This approach allows for the transformation of software production into a genuine process of mutual learning and knowledge construction, where each iteration becomes a stage of experimentation and consolidation.

The empirical results suggest that the use of a generative AI, framed by a rigorous methodology, can facilitate achieving a level of software quality equivalent to that of traditional development, while significantly reducing production time and enhancing the traceability of the reasoning. The combination of a specialized GPT model and an integrated ethical framework ensures that the co-creation remains explainable, transparent, and accountable, in compliance with the requirements of scientific and institutional governance.

Beyond the specific case studied, this research paves the way for a new form of augmented educational engineering based on co-intelligence and human oversight. The CICAP paradigm offers a model that is transferable to other open-source development contexts whether for learning platforms, analytics tools, or recommendation systems, and provides a structuring methodology for teams wishing to integrate artificial intelligence into their design practices. Furthermore, beyond its engineering utility, the CICAP paradigm holds significant pedagogical potential for training future educational technologists. By shifting the focus from syntactic rote coding to architectural supervision and prompt engineering, it offers a structured method to teach students how to govern generative AI responsibly, preparing a new generation of 'AI-augmented' developers capable of blending technical proficiency with ethical oversight.

Within the scope of this case study, the CICAP paradigm appears not limited to a technical protocol: it embodies a philosophy of responsible co-creation, where technology and humans evolve within a framework of trust, transparency, and shared learning. It opens a fruitful perspective for future research on co-intelligence, scientific reproducibility, and the ethical governance of artificial intelligence in digital educational environments.

Data Availability Statement

The complete dataset supporting the findings of this study, including the full XML transcript of the human-AI interaction logs (Conversation.xml) and the source code of the generated artifact, is openly available in the GitHub repository at: https://github.com/kkarroumi/opportunities

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Yassine Karroumi conducted the research, analyzed the data, and wrote the paper. Fatima Ou-zennou, Khalid EL Khattabi, and Souad Eddarouich reviewed the paper. All authors approved the final version.

References

[1] S. Qazi, S. Khan, M. A. Qureshi, and A. R. Hafeez, "AI-driven learning management systems: Modern developments, challenges, and future trends during the age of ChatGPT," *Computers, Materials & Continua*, vol. 80, no. 2, pp. 2589–2612, 2024.

[2] S. Mahamad, Y. H. Chin, N. N. Zulmusha, and M. M. Haque, "Technical review: Architecting an AI-driven decision support system for enhanced online learning and assessment," *Future Internet*, vol. 17, no. 2, 45, 2025.

[3] F.-J. Hinojo-Lucena, I. Aznar-Díaz, M.-P. Cáceres-Reche, and J.-M. Romero-Rodríguez, "Artificial intelligence in higher education: A bibliometric study on its impact in the scientific literature," *Education Sciences*, vol. 9, no. 1, 51, 2019.

[4] P. Brey and B. Dainow, "Ethics-by-design and the governance of artificial intelligence," *AI and Ethics*, vol. 3, no. 2, pp. 123–139, 2023.

[5] W. Holmes, M. Bialik, and C. Fadel, *Artificial Intelligence in Education: Promises and Implications for Teaching and Learning*, Boston, MA: Center for Curriculum Redesign, 2019.

[6] R. Luckin, W. Holmes, M. Griffiths, and L. Forcier, *Intelligence Unleashed: An Argument for AI in Education*, London: Pearson, 2016.

[7] M. Zabolotniaia, O. Cheng, E. Dorozhkin, and A. Lyzhin, "Use of the Moodle learning management system to support institutional processes and digital governance," *Education and Information Technologies*, vol. 25, no. 3, pp. 1659–1676, 2020.

[8] C. Romero and S. Ventura, "Educational data mining and learning analytics: An updated survey," *WIREs Data Mining and Knowledge Discovery*, vol. 10, no. 3, e1355, 2020.

[9] H. A. Alamri, S. Watson, and W. Watson, "Learning technology models that support personalization within blended learning environments in higher education," *TechTrends*, vol. 65, no. 2, pp. 138–150, 2021.

[10] A. Tarhini, K. Hone, and X. Liu, "The effects of individual differences on e-learning users' behaviour in developing countries: A structural equation model," *Computers in Human Behavior*, vol. 41, pp. 153–163, 2014.

[11] D. Gašević, S. Dawson, and G. Siemens, "Let's not forget: Learning analytics are about learning," *TechTrends*, vol. 59, no. 1, pp. 64–71, 2015.

[12] D. R. E. Cotton, P. A. Cotton, and J. R. Shipway, "Chatting and cheating: Ensuring academic integrity in the era of ChatGPT," *Innovations in Education and Teaching International*, vol. 62, no. 1, pp. 1–12, 2024.

[13] S. Peng, E. Kalliamvakou, P. Cihon, and M. Demirer, "The impact of

AI on developer productivity: Evidence from GitHub Copilot," arXiv preprint, arXiv:2302.06590, 2023.
[14] International Software Benchmarking Standards Group (ISBSG). (2023). Practical Guide: Productivity Measurement of Software Projects. Industry Report. [Online]. Available: https://www.isbsg.org/wp-content/uploads/2021/08/Productivity-Measurement-of-Software-Projects.pdf
[15] S. Amershi *et al.*, "Guidelines for human-AI INteraction," in *Proc. the 2019 CHI Conference on Human Factors in Computing Systems*, 2019, pp. 1–13.
[16] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design science in information systems research," *MIS Quarterly*, vol. 28, no. 1, pp. 75–105, 2004.
[17] Vellum AI. (2024). LLM leaderboard: Quality, latency, and price benchmarks. [Online]. Available: https://www.vellum.ai/llm-leaderboard