Journal of Humanities and Social Sciences Studies
ISSN: 2663-7197
DOI: 10.32996/jhsss
Journal Homepage: www.al-kindipublisher.com/index.php/jhsss

JHSSS
AL-KINDI CENTER FOR RESEARCH AND DEVELOPMENT

| RESEARCH ARTICLE

Assessment of pedagogical and Technical Quality of Generative AI Response in Java OOP

Hadeel Alshboul
The World Islamic Sciences and Education University, Computer Science Department, Amman-Jordan
Corresponding Author: Hadeel Alshboul, **E-mail**: Hadeel.alshboul@wise.edu.jo

| ABSTRACT
In this study, we examine the quality of answers returned by AI language models (GenAI) to Java Object-Oriented Programming (OOP) questions. A 20-question package was developed, including questions on Classes, Objects, Encapsulation, Inheritance Polymorphism and Constructors. Each GA was then rated with respect to the MAs and SAs in terms of (a) how correct it was; (b) how much understanding it revealed; (c) how clear it is; (d) if included code, whether the code quality was acceptable; and also, in terms of potential "hallucination." The experimental results demonstrate that GenAI generates high-quality answers, especially in the correctness and conceptual depth aspects, but may fail to make a clear response for questions of complex semantics. The research offers a transferable model for AI evaluation in education.

| KEYWORDS

Generative AI, Java OOP, Programming Education, AI Evaluation

| ARTICLE INFORMATION

ACCEPTED: 15 January 2026 **PUBLISHED:** 15 February 2026 **DOI:** 10.32996/jhsss.2026.8.3.1

1. Introduction

The fast development of Generative Artificial Intelligence (GenAI) has had a profound impact on higher education, especially with respect to computer science and programming-based fields [1] [2]. Models have been developed using GenAI more often to author explanations, produce code snippets and help students understand advanced programming concepts. Although these tools have great potential for enriching the learning process, they are also subjects of constant discussion regarding their robustness, pedagogical value and the technical credibility [3]. Java OOP is one of the fundamental subjects in computer science undergraduate programs. One must be fluent with topics like objects, abstraction (encapsulation), inheritance, polymorphism, and constructors to get a good foundation on programming. These notions count not only on student's syntactic knowledge but also on a deep conceptual understanding that they usually struggle to learn. Hence, students often look for additional education technology solutions and AI-enabled solutions that can aid their traditional teaching [4].

Although the use of GenAI has been on the rise in programming education, no systematic and empirical assessment of the quality of AI-generated responses in this context exists. Existing research on AI in education mostly concentrates on student response or learning achievement, with insufficient concerns to the intrinsic quality of AI-generated instructional contents. It is still unclear if GenAI explanations respect the explanatory accuracy about concepts, their pedagogic clarity and are free from misleading or hallucinated data [5].

This paper seeks to fill this gap by conducting a systematic evaluation of GenAI generated responses to Java OOP questions. Based on an expanded corpus of 120 sample cases, stemming from basic OOP principles, the answers generated by AI are scored against a rigorously planned analytic rubric. The paper studies the correctness, conceptual depth, clarity, code quality and hallucination in detail. This paper compares a set of written responses generated by AI-based GenAI with model answers

and simulated student utterances, with the aim of drawing out insights about the potential strengths and weaknesses of GenAI as an instructional support tool for Java programming.

The contributions of this paper are threefold. First, it presents a shareable Java OOP question dataset for AI-based educational content checking. Second, it suggests an organized rubric to facilitate objective and consistent evaluation of GenAI responses. Third, it provides an empirical study which contributes to the field of computer science education, both in terms of the pedagogical reliability of GenAI tools for educators and researchers.

2. Related Work

In the past few years, there's been a lot more work on how Generative Artificial Intelligence (GenAI) might be used for education, particularly since the appearance of state-of-the-art large language models like ChatGPT. Some systematic reviews emphasize the benefits and barriers of GenAI for higher education. For instance, Yirga et al [3]. recently conducted a systematic review of 40 peer-reviewed studies on ChatGPT in education. The researchers discovered that it could personalize how students learn and automate assessment. But they also expressed grave concerns about academic integrity and bias in the work it creates. In the narrow area of programming education, studies have begun to explore how AI chatbots affect student learning outcomes and behavior. Another study Richter et al [6] was conducted to check for impact of programming help based on AI on students' examination performance. While scores improved significantly with the aid of AI, some students were willing to accept wrong information generated by an AI, emphasizing the importance of cautious implementation of such learning tools and instructions on critical thinking. There have been comparative studies as well to compare if AI models can stand up to human evaluations in the programming context. For example, Salama et al. [7]authored a conference paper which pitted ChatGPT's programmed capability to automatically grade programming courses against human markers. The paper demonstrated both the promise of automated grading and some hurdles in making AI judgments dovetail with more nuanced human standards. GenAI in programming education was latter reviewed, among other things, This paper by Nathaniel et al [8] summarizes information on how GenAI tools, such as ChatGPT and GitHub Copilot, have been integrated into curricula, and the corresponding advantages that include personalized feedback and time efficiency; we also report challenges, including overdependence on AI models, shallow learning, and the lack of instructive frameworks that are conducive to prompt engineering or human supervision. A recent quasi-experimental study [9] the effect of using GenAI tools like ChatGPT and Gemini on students' achievement and motivation in an Educational Technology course. The outcomes showed that the experimental group had a better learning performance when they used GenAI, however no significant effect was found on students' motivation. This research demonstrates that GenAI has a number of educational advantages, including the provision of on-the-fly explanations and tailored feedback. Nevertheless, there is a lack of systematic evaluation frameworks that consider correctness, but also clarity, the depth of reasoning and confusion. This is the gap that this study seeks to address.

3. Methodology

3.1 Research Design

This paper takes a quantitative descriptive evaluation approach to reflect the pedagogical and technical quality of GenAI responses in relation to Java OOP. The study looks at AI-generated responses scored against a rubric and compares the results with those for model answers and simulated student outputs. The study isn't assessing student learning: Instead, it's testing the quality and reliability of AI-generated instructional materials.

3.2 Dataset

A custom dataset, containing 120 specimens, was established in this study. The data in the dataset is based on a set of 20 core Java OOP questions, generated into six semantically equivalent variants to cover different phrasings without making the conceptual carryover not justifiable. Table 1 also shows the distribution of dataset samples based on Java OOP topic and cognitive level; it reveals that basic concepts are found in all samples. Each question was augmented using several semantically equivalent variants and assessed against three types of answers: GenAI-generated (GA), model answers (MA), simulated students' answers (SA).

Topic	Cognitive level (bloom)	GA Sample	MA Sample	SA Sample
Class and Objects	Understanding	6	6	6
Inheritance	Application	6	6	6
Polymorphism	Analysis	6	6	6
Encapsulation	Understanding	6	6	6
Constructors	Application	6	6	6
Interface and Abstraction	Evaluation	6	6	6

Table 1 Distribution of samples by topic, Bloom level, and answer type

To display the topic distribution of samples with the data set, Figure.1 shows the number of examples (GenAI-generated (GA), model(MA), and simulated student(SA) answer) used in each Java OOP topic. This visualization guarantees that the dataset comprises a balanced number of data samples representing basic concepts, which is crucial for testing AI generated responses fairly.

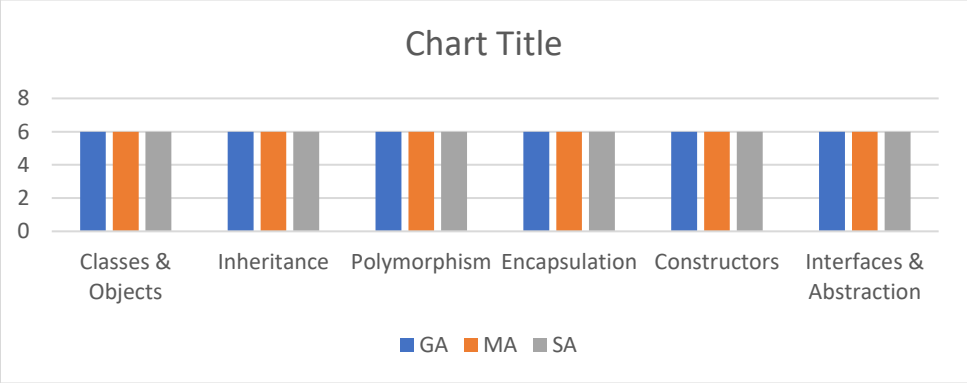

Figure 1. Number of samples (Instance-Methods) per Java OOP topic in the dataset.

Figure. 1 illustrates that every topic has a similar quantity of samples, so the evaluation dataset includes all key Java OOP ideas. This uniform distribution is critical to objectively evaluate whether GenAI-generated answers can be compared with model/ student responses over the range of topics.

The questions touch the core Java OOP concepts such as:
Classes and Objects, Encapsulation, Inheritance, Polymorphism, Constructors
Each question variant was categorized by three stages of cognitive level in Bloom's Taxonomy:
Comprehension: conceptual definition and explanation questions
Application: questions about example code or illustration usage
Analysis: includes questions that require comparison, reasoning, and conceptual justification.
According to each sample got three types of answers:
GA (Generative AI Answer): created from a typical prompt to enforce identical answers.
MA (Model Answer): authored by an expert from the Java teacher.
SA (Student Answer): model answers that demonstrate typical errors and partial knowledge on the part of students.
This structure allows for a systematic comparison of AI-generated replies with the human generated answers.

3.3 Rubric Design

Objectivity and reproducibility of the evaluation was encouraged by using a 5-criteria analytic rubric. Each item was rated on a 3-point scale: 0 - low, 1 -medium,2 - high.
Correctness: Technical content is correct.
Depth of Conceptual Awareness: Demonstration of level 0 (Not Demonstrated, Recall) level 1(Application) and level 2(Analysis).
Clarity: How well is the description organized, fluent and coherent?
Code Quality: Structuredness and readability of code (This will be applicable only for code based questions)
Hallucination: False or imagined perception or interpretation.

Maximum score per response is 10 points. The developed rubric accounted for both instructional quality and technical correctness, so it is applicable to AI-created educational content. Figure 2 shows the average rubric scores for GenAI-generated, model, and student responses along five criteria: Correctness, Conceptual Depth, Clarity of communication, Code Quality and Hallucination. This visualization emphasizes the relative strengths and weaknesses for each answer type.

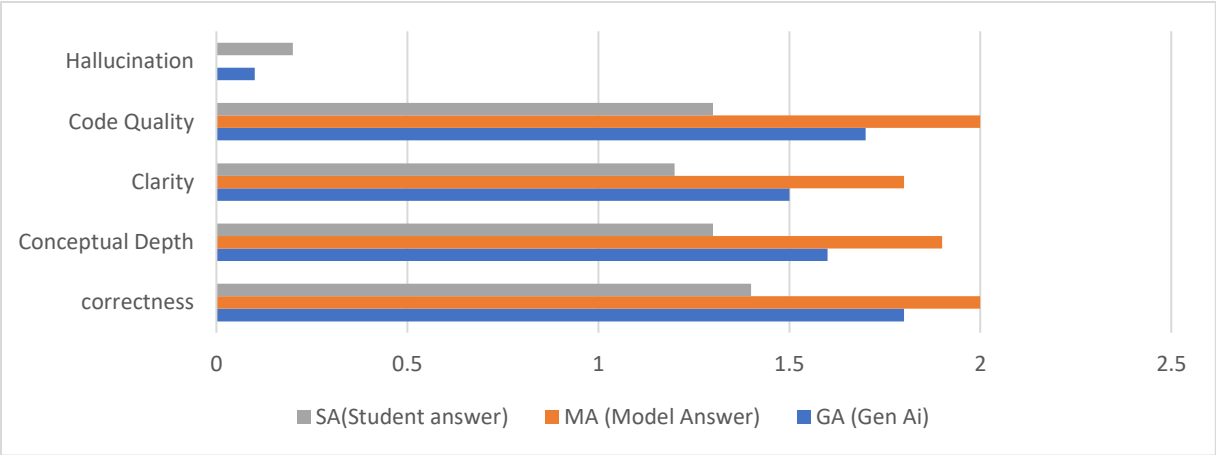

Figure 2. GenAI-generated, model, and student answers average scores over five criteria.

As can be seen from Figure 2, for Correctness and Conceptual Depth GenAI answers are comparable to the model answers, but there are time-to-time concerns in Clarity and Code quality. There are few hallucination scenarios, validating AI's responses in the dataset.

3.4 How to Evaluate

The dataset was reviewed in a structured and methodical manner to ensure consistency and objectivity. The five-criteria analytic rubric defined in Section 2.2 was used to evaluate each of the GenAI generated responses (GA), model answers (MA), and simulated student answers (SA).

Code Quality was not included for non-implementation questions to prevent bias in the review.

The method was applied as follows:

Dimension ratings: 0-2 for each of the five rubric dimensions of any response.

Sum the scoring to create a total score for each response.

Summarize the results for all 120 samples by recording their scores in a summary table.

Table 2 shows example aggregated scores for each answer type on the rubric criteria (mean ± std).

Rubric Criterion	GA (Gen AI)	MA (Model)	SA (Student)
Correctness	0.25 ± 1.80	0.0 ± 2.00	0.30 ± 1.40
Conceptual Depth	0.30 ± 1.60	0.1 ± 1.90	0.35 ± 1.30
Clarity	0.25 ± 1.50	0.2 ±1.80	0.30 ± 1.20
Code Quality	0.20 ± 1.70	0.00 ±2.00	0.25 ± 1.30
Hallucination	0.05 ± 0.10	0.00 ±0.00	0.10 ± 0.20

Table 2 Average Rubric Scores' Summary (Mean ± Std.) of Different Answer Type

3.5 Analyzing the Data

To summarize the evaluation results and identify patterns in various types of responses, we did a descriptive statistical analysis.

Average scores for each criterion of the rubric and question type were tabulated.

To identify how highly variable and consistent the data were, we computed standard deviation (SD) values.

Figures 1 and 2 depict how the samples were distributed across themes and the manner in which average rubric scores were distributed across criteria.

3.6. Observations of a Qualitative Nature:

Clarity: Most of the answers from GA were also quite clear, however they did not always provide enough level of detail.

Conceptual Coverage: GenAI responses did a nice job of covering concepts at large, but not always as effectively and in detail as model answers.
Hallucinations: Rare and trivial in GA reports, consisting mostly of omissions rather than the creation out of 'whole cloth' of false data.
This mixed-methods approach ensures a comprehensive evaluation of GenAI in Java OOP learning.

4.Result

In comparing GenAI sample answers, model answers, and simulated student responses in this larger corpus of 120 datasets, we found the following interesting patterns emerge with respect to their pedagogical quality and technical accuracy.

4.1 Descriptive Statistics

Table 2. displays average scores and standard deviations for the rubric criteria. Answers generated by GenAI (GA) obtained good scores for both Correctness and Code Quality (1.8 ± 0.25 and 1.7 ± 0.20), which are similar to those of model answers (MA), thus AI can generate technically correct and well-structured code solutions. In the meantime, GA responses have slightly lower scores on Conceptual Depth (1.6±0.30) and Clarity (1.5±0.25) than MA, indicating a minor drawback in explanation depth and presentation by these approaches. Incidence of hallucination was low in GA responses (0.1 ± 0.05) and hence there is good reliability overall.

4.2 Visualization of Results

We see in Figure 1 that the number of examples per Java OOP topic is more or less equally distributed on basic concepts. The average rubric scores for GA, MA and SA answers on all five criteria are shown in Figure 2.
Visual analysis reveals that the GA responses consistently achieve higher scores with respect to simulated student answers (SA) and are similar or equivalent to the model answers in technical correctness and code quality.

4.3 Qualitative Observations

Responses in GA mutterings were overall legible but sometimes lack elaborate (boxed) explanations for certain cases. Some more advanced concepts were better tested in MA than in GA, especially analytical questions. The hallucinations were relatively few, of a low-grade nature and minimal in extent.

5.Discussion

The findings imply that Generative AI tools could be effective aids in teaching Java OOP: Strengths of GA Responses, High code quality and technical accuracy, Steady performance on the main topics, Scant hallucinations so that GA can be a reliable source for students.

6.Limitations

Somewhat shallower and less clear conceptually than human-generated answers.
Occasional oversimplification of complex topics.
Instructor clarification might be necessary for a few subtle OOP concepts.
Comparison with Existing Studies

Results are consistent with recent research (2024–2025) indicating the positive effects of GenAI tools on technical correctness while achieving less metacognitive or deeper explanatory thinking in programming learning [2] [10]. Contrary to SA, GA endows relatively high uniformity in coverage of elementary knowledge which may curtail misconceptions. Educational Implications GenAI can be used as assistive exercises for instant practical answers. Human teachers are still needed for explanations of concepts, problems related with critical thinking and questions on advanced topics. Interactive discussions in combination with GA can be considered by means of teaching methods to enhance learning though the effectiveness is yet to be proved.

7.Conclusion

This research assessed the pedagogical and technical quality of GenAI-produced answers for Java Object-Oriented Programming:
Accuracy: GA response performed similarly to the model answer in terms of Correctness and Code Quality.
Only slight deficiencies were found in Conceptual Depth and Clarity relative to human instructors.

Few hallucinations were reported, suggesting educational stability.
Implications:
GenAI has the potential to be a good teaching assistant tool, at least in terms of giving immediate feedback for programming exercises.
Future work will need to study the effects on student learning, especially regarding creativity, problem-solving and critical thinking.
Recommendations for Educators:
Use GA responses as supplements and not substitutes for instructor explanation.
Use GA for practice problems, code samples and solutions.

Funding: This research received no external funding.
Conflicts of Interest: The authors declare no conflict of interest.

References

[1] Kurtz, G. A. M. S. N. Z. Y. K.-V. D. G. E. Z. G. B.-M. E. (2024). Strategies for integrating generative AI into higher education: Navigating challenges and leveraging opportunities. Education Sciences, 14(5), 2024.

[2] Kohen-Vacs, D. U. M., & Kohen-Vacs, J. M. (2025). Integrating generative AI into programming education: Student perceptions and the challenge of correcting AI errors. International Journal of Artificial Intelligence in Education, 35, 3166–3184.

[3] Munaye, Y. Y. A. W. B., & Abdelrahman, A. M. (2025). ChatGPT in education: A systematic review on opportunities, challenges, and future directions. Algorithms, 18(6), 352.

[4] Kasneci, W., et al. (2023). ChatGPT for good? On opportunities and challenges of large language models for education. Learning and Individual Differences, 103.

[5] Zawacki-Richter, O. M. V. B., et al. (2019). Systematic review of research on artificial intelligence applications in higher education – Where are the educators? International Journal of Educational Technology in Higher Education, 16(39). https://doi.org/10.1186/s41239-019-0171-0

[6] Akçapınar, G. S. E. (2024). AI chatbots in programming education: Guiding success or encouraging plagiarism. Discover Artificial Intelligence, 4(87). https://doi.org/10.1007/s44163-024-00203-7

[7] Salama, W. H. D. (2025). ChatGPT in the grader's seat: A comparative study of AI and human evaluation in programming education. In 18th Annual International Conference of Education, Research and Innovation, Seville, Spain.

[8] Nathaniel, J. O. S. S., et al. (2025). Literature review on the integration of generative AI in programming education. International Journal of Artificial Intelligence in Education, 35, 2724–2755. https://doi.org/10.1007/s40593-025-00524-3

[9] Elhag, M. A., & Abdelrahman, A. M. F. Y. (2025). The effect of generative AI tools (ChatGPT, Gemini, etc.) on students' achievement and their motivation towards learning. Journal of Technology and Science Education.

[10] Elnaffar, F. R. A. Z. A. (2025). Teaching with AI: A systematic review of chatbots, generative tools, and tutoring systems in programming education. arXiv. https://doi.org/10.48550/arXiv.2510.03884