

Integrating Virtual Assistants into the Programming Learning Process within IT Studies

Ana Veljić

Faculty of Applied Management,
Economics and Finance
Belgrade, Serbia
ana.veljic@mef.edu.rs
ORCID 0000-0001-7689-044X

Marko Marković

Faculty of Applied Management,
Economics and Finance
Belgrade, Serbia
marko.markovic@mef.edu.rs
ORCID 0009-0002-6449-6589

Srdan Maričić

Faculty of Applied Management,
Economics and Finance
Belgrade, Serbia
srdjan.maricic@mef.edu.rs
ORCID 0000-0002-9261-7061

Abstract—In the modern higher education environment of the field of Information Technology (IT), the application of modern technological tools is more and more oriented toward the integration of the existing gap between theoretical and applied knowledge. One of the major problems in the development of IT courses is the process of introductory programming, often associated with high levels of cognitive load and varied levels of student performance. This paper aims to investigate the pedagogical potential and operational application of the integration of AI-based tools of virtual agents in the process of programming education. This is based on the theoretical background of the constructivist approach and the self-regulated learning (SRL) model. The empirical part of the research, carried out among IT students, examines their views concerning the usefulness of virtual assistants in overcoming syntax barriers and improving problem-solving fluency. The research findings suggest that although AI tools greatly alleviate cognitive load and motivate learners, it is necessary to apply a didactic approach in order to develop genuine computational thinking. The research findings contribute to the discussion aimed at improving the quality of IT study programs and making them compatible with the changing needs of the global labor market.

Keywords—Virtual Assistants, Programming Education, Artificial Intelligence in Education, Information Technology Studies, Student Perception

I. INTRODUCTION

In the context of modern higher education, the teaching of programming skills has to overcome a set of related problems that have a direct influence on the quality of the future experts in the information technology domain. Primarily, the problems are related to the cognitive load and the maintenance of the learners' motivational state, as well as the existing discrepancy between the theoretical logic of algorithms and the actual syntax knowledge [1]. The relevance of the alignment of the learning outcomes with the evolving needs of the labor market and the future professional aspirations of the students has already been highlighted as a crucial factor in the improvement of the quality of the information technology study programs [2]. In the context of the cognitive load theory, the quality of the teaching program has to ensure a careful balance between the intrinsic and extraneous cognitive load, where the multimodal approach may be highly supportive in the development of the learners' skills in computer coding [3]. However, whereas intelligent tutoring

systems have been proven to have the potential to improve computational thinking through feedback mechanisms, there has been a systemic problem in the way in which the mechanics are emphasized to the extent that they have overshadowed the critical analysis and design of algorithms [4]. This, therefore, has affected not only the academic performance but also the problem-solving skills in the modern engineering landscape [5].

To address the persistently high attrition rates in introductory computer programming courses, computer science education in the present day increasingly employs constructivist theory and SRL frameworks. Indeed, constructivist theory changes the paradigm in computer science education by emphasizing knowledge construction and learner autonomy, where the instructor acts as a facilitator of independent thinking [6]. In this regard, SRL allows learners to develop specific learning objectives and self-monitor their cognitive development, a technique that has been directly linked to reduced attrition rates in introductory computer programming courses [7]. Empirical studies indicate that constructivist theory increases learner motivation by promoting reflection and strategic thinking in the completion of intricate computer coding tasks. Moreover, the process of effective 'scaffolding' and the use of instant feedback systems help in filling the knowledge gap between the existing knowledge and the new algorithmic concepts in a way that keeps the learner engaged continuously [8]. This not only facilitates the process of autonomous and collaborative learning but also contributes to the development of the learner's professional flexibility in terms of the fluency of problem-solving in various programming scenarios [9]. This guidance system is vital in the development of the learner's self-efficacy needed in the IT job market and is in accordance with the overall objectives of enhancing the quality of engineering education.

The rapid development in the field of Large Language Models (LLMs) and virtual assistants in the period between 2023 and 2026 has ushered in a new paradigm in programming learning with the introduction of 'tutoring agents.' The recent frameworks developed through generative AI have been based on the concept of dialogic interactions, which are aimed at creating conditions for 'explanatory discourse' or 'scaffolding' rather than providing any direct or complete solution to the problem. In these frameworks, virtual assistants are used as

intelligent agents to provide detailed explanations for the codes, assist in debugging, or provide students with practice exercises based on their individual learning needs [10]. This is aimed at developing computational thinking skills in students by taking them through the logical steps in solving the problem rather than providing them with the complete code. However, while empirical evidence suggests that there is a substantial decrease in perceived cognitive load and an improvement in overall learning attitude, concerns with the accuracy of the generated content and the ethics of using such systems remain prevalent [11]. To overcome such issues and challenges associated with the development of such systems, the development of education-specific LLMs (Ce-LLMs) aims to reduce the hallucinations associated with the generated content through adaptation and prompting [12]. It is argued that the overall integration of such systems is not a replacement for the traditional education system but rather a sophisticated system aimed at enhancing conceptual understanding in the modern engineering curriculum itself [13].

Recent literature has pointed to a need in academic circles to understand that a critical analysis of student perceptions is a basic prerequisite for the successful integration of AI tools in computer science curricula. In this context, empirical surveys are increasingly considered a necessary step to assess the effectiveness of AI-based virtual assistants, as quasi-experimental research has shown that AI co-tutors can have a considerable influence on student engagement and academic performance [14]. In addition, research has shown that personalized guidance through Generative AI can have a considerable influence on student motivation and self-efficacy perceptions [15]. This need is also underscored by the need to ensure that modern IT curricula meet not only the changing requirements of the labor market but also the aspirations of students entering these fields of study [2, 16]. In this context, this research aims to examine the influence of virtual assistants on the student learning process in programming, particularly in terms of how these tools are perceived by students to be helpful in overcoming algorithmic and syntax barriers.

II. METHODOLOGICAL AND HYPOTHETICAL RESEARCH FRAMEWORK

The questionnaire conducting the research consists of seven questions that will help us gather relevant information about Virtual Assistants. The aim of the research is to find out what characteristics a virtual assistant should possess in order to help students learn programming. All questions are closed-ended, and users can choose from the provided options. The survey provides a detailed introduction to the Virtual Assistant, common places where it can be found, segments in which the Virtual Assistant can assist students, the level at which it can help, and an assessment of whether the Virtual Assistant is a quality application for learning programming.

From the data collected in the survey, we have the chance to identify the needs of students, which is critical in order to optimize the effectiveness of virtual assistants in the programming learning process. After the identification of the needs of students, it is possible to optimize the educational framework by prioritizing the support features that students consider to be of greater importance. This way, it is guaranteed

that the implementation of AI tools is focused on the academic development of students and the improvement of teaching strategies. To ensure methodological rigor in this research, it is necessary to implement a deductive method by testing hypotheses.

The main hypothesis of this research is:

H0: The implementation of a virtual assistant significantly enhances the programming learning process for IT students.

The main hypothesis is supported by auxiliary hypotheses which state:

H1: Existing faculty resources and platforms serve as a foundation for integrating new modalities of digital learning support.

H2: A virtual assistant enables faster and more efficient distribution of teaching materials and key information for various programming languages.

In order to gather the highest quality scientific research results, we will use the following methods:

- The fundamental methods of knowledge and research will, thru the writing of this paper, allow us to draw conclusions about the advantages and benefits that virtual assistants provide for these purposes, using the methods of induction and deduction. These methods are important in order to achieve the best possible results during the research.
- The analytical method will explain how and in what way these items affect the execution of these processes thru the analysis of the entire scientific literature that explains and defines information systems, the approach to information, the use of programming languages, and the virtual assistants themselves.
- The statistical method was applied to present the statistical data provided by the surveyed students about virtual assistants and similar services at the faculty.

III. SURVEY ANALYSIS

A. Research methodology

The research was conducted on a sample of 220 students from the Faculty of Applied Management, Economics, and Finance in Belgrade. Based on this research, the following results have been determined and are presented thru graphs. Based on this research, a report will be created that will help us better understand students' attitudes toward Virtual Assistants. In the next section, we will present the data collected during the research, as well as the conclusions drawn from the analysis of the survey results. All questions, as well as conclusions, will be presented separately along with a graphical representation of the results. The analysis encompasses the key parameters of this research and will also provide us with a better insight into the current situation.

B. Analysis of research results

The following paper will present the data collected through the survey and the conclusions reached during the research. The questions will be presented individually, followed by the graphic display and the results obtained.

Are you familiar with the term Virtual Assistant?

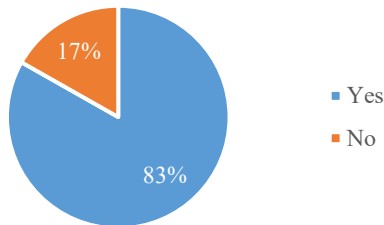


Fig. 1. Display of knowledge of virtual assistants

We see that 83% of respondents, or 183 respondents, are familiar with the term Virtual Assistant, while 17% of respondents, or 37 respondents, are not familiar with the term Virtual Assistant. This data indicates that a significant majority is familiar with the concept, and further research could yield relevant results and a good opportunity for drawing accurate conclusions. Since virtual assistants are a novelty, this presents us with a great opportunity to present this project as an innovation and an excellent technological solution for a higher education institution.

If you used a Virtual Assistant, was it helpful to you?

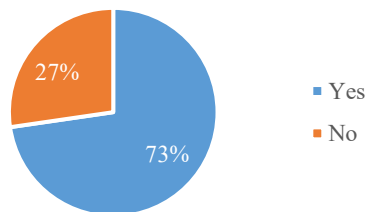


Fig. 2. Display of Virtual Assistant usefulness

In a sample of 220 students, 160 or 73% are satisfied with the work of the Virtual Assistant and claim that it has been helpful in carrying out the requested task. Sixty of them, or 27%, are not satisfied with the performance of the Virtual Assistant. The percentage of negative responses can be viewed from the perspective of the artificial intelligence tool that was used the most.

Where have you encountered a virtual assistant?

Table I - Places where you've used a virtual assistant

Telecommunications companies	29.5%
Banks	38.6%
AI tools	59.5%

Operating systems	48.2%
Other places	15.0%

In Table I, we can clearly see which models of Virtual Assistants the respondents have encountered, as well as which of them they have used so far. According to the research and results, Artificial Intelligence Tools (Chat GPT, Copilot, Claude, Gemini...) rank first, as they have been used the most by students. As many as 131 respondents used these models, which represents a sample of 59.5%. In second place are the operating systems (Siri, Alexa, Cortana), with which as many as 106 students are familiar. This type of virtual assistant is found on mobile phones or computers; they are designed to make it easier for users to work on smart devices. In third place are the models used by banks, with as many as 85 respondents, or 38.6% of those surveyed, having encountered banking models of virtual assistants. In fourth place are the Virtual Assistants of Telecommunications Companies, with 65 respondents having encountered this model. The most prevalent on the market is the Virtual Assistant Mila, from the telecommunications operator Telekom Serbia. 15% of respondents indicated the option "Other," which represents some other source of Virtual Assistant on the internet.

How do you perceive the potential support provided by a virtual assistant?

Table II - Ways in which virtual assistants assist in learning programming

Helping with writing code	42.3%
Assisting with debugging	62.7%
Explaining programming concepts	55.0%
Supporting the learning of IT fundamentals	40.9%
Other	14.1%

The results shown in Table II identify the specific areas where respondents perceive a Virtual Assistant to be most beneficial. To a large extent, as many as 138 respondents, or 62.7%, believe that the most important item is Assisting with debugging. This majority believes that one of the most important services that a Virtual Assistant can provide is this, which further suggests that it is very important for students to have detailed access to this segment. In second place, as many as 121 respondents believe that the ability to explain programming concepts is important. In third place is Helping with writing code, which 93 respondents consider to be an important item. Supporting the learning of IT fundamentals is next in importance for students. In the end, 30 respondents chose the option "other" as their selection; this item may indicate that there is additional information needed for the Virtual Assistant to contain.

Do you think a Virtual Assistant could reduce the time you spend solving programming problems?

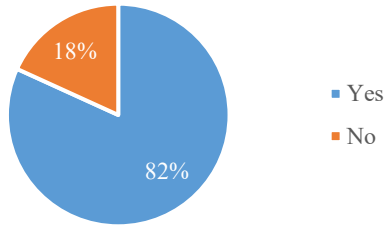


Fig. 3. Perceived impact of Virtual Assistants on programming problem-solving time

By saving time, students could better optimize their schedules and create a better plan for studying and fulfilling all their academic obligations. In the given survey, a large percentage of students, 81.8%, believe that a Virtual Assistant could significantly help them and reduce the time they spend troubleshooting programming issues. A significantly small percentage, 18.2%, believes that a Virtual Assistant would not help them and would not reduce the time spent on debugging.

Rate the usefulness of a virtual assistant on a scale of 1 to 5.

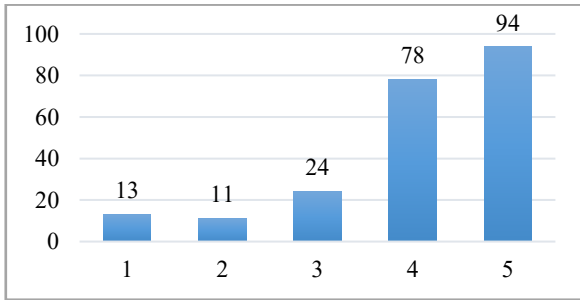


Fig. 4. Virtual Assistant usefulness rating

To determine whether the implementation of a Virtual Assistant is a good option for learning programming, we had to conduct an analysis of the usefulness of the Virtual Assistant. From the chart provided, we can see the ratings that currently indicate that the Virtual Assistant would be of great benefit within IT studies. 13 respondents gave the lowest score of 1, 11 respondents gave a score of 2, while 24 respondents gave a score of 3. This assessment shows that 48 respondents gave a low rating to the Virtual Assistant, indicating that they do not believe the virtual assistant would be useful in learning programming. If we look at the overall results, we see that this percentage is similar to all the negative percentages in the entire survey and that a certain population that filled out the survey is not in favor of the Virtual Assistant. A score of 4 was given by as many as 78 respondents, and a score of 5 was given by 94 respondents. In total, this amounts to 172 respondents who gave a very high rating and who believe that it is desirable to use a Virtual Assistant as an aid in learning programming within IT studies.

Which programming languages would you prefer to learn with the help of a Virtual Assistant?

Table III - Programming languages users would learn with a Virtual Assistant

Java	55.3%
C	68.2%
Python	29.0%
C++	20.9%
Web Languages (HTML, CSS, JS)	30.1%

In Table III, which shows the programming languages that the respondents want to learn with the help of a Virtual Assistant, we have the following results: to a large extent, as many as 150 respondents, or 68.2%, showed interest in learning C, which makes it the most chosen option. This shows that the respondents have recognized the value of using a Virtual Assistant when dealing with programming languages. In second place, 122 respondents or 55.3% preferred Java, which proved its relevance and importance to learners. The next option, web programming represented by HTML, CSS, and JavaScript, was preferred by 66 respondents or 30.1%, while Python was preferred by 64 respondents or 29%. Lastly, the option of C++ was preferred by 46 respondents, 20.9%, making it the least preferred option among the options. This reveals the high interest of the respondents in the use of Virtual Assistants for the acquisition of knowledge in the programming of universally applicable programming languages.

IV. THE DISCUSSION OF THE RESULTS OF THE RESEARCH

The results obtained from the conducted research offer insight into the attitudes of students toward the use of virtual assistants in programming education. The results obtained clearly show that virtual assistants form an essential part in the current education system, especially in the field of IT [17]. First, the level of knowledge about virtual assistants shows that students are already familiar with this technology. This provides grounds for the possibility of integrating it into the learning environment, as the majority of students feel comfortable working with such tools, without the need for significant adjustments. In addition, the majority of the respondents find virtual assistants useful, thus proving their practical utility in the learning process. Nevertheless, a smaller number of users were not satisfied, suggesting that there are some limitations associated with the current solutions. These limitations may be related to aspects, such as accuracy, context, or specifics of the students' use, suggesting that further improvements are needed, as well as better adaptation of virtual assistants within In addition to this, the majority of the respondents find virtual assistants useful, and this proves their practical utility in the learning process. However, a smaller number of users were not satisfied, and this indicates that there are limitations associated with the current solutions education [18]. The analysis of the context of use indicates that tools, such as ChatGPT and Copilot, based on AI are the most used virtual assistants by students. This may be related to a general trend of moving towards advanced tools, capable of providing sophisticated support, rather than traditional tools integrated within operating systems or customer support platforms.

However, as the use of these technologies increases, it is imperative to develop a framework for reporting and ethical evaluation of artificial intelligence studies using a standardized framework, such as the RAISE model. This academic rigor is essential for the development of effective educational assistants that can be successfully integrated into the existing faculty infrastructure [19]. It also underscores the need to develop effective educational assistants who can make good use of AI [20]. When looking at the role of virtual assistants in programming learning, the findings show that the most valued functionality is assistance in debugging the program. This is also important since debugging is usually one of the most challenging aspects of programming for students. Additionally, the role of explaining programming concepts and assisting in coding is also essential. This implies that an efficient virtual assistant should focus more on problem-solving and concepts, not just code generation. Another interesting fact is that the majority of the respondents think that the use of virtual assistants can help reduce the time taken in solving programming problems. This reveals that not just the educational importance of virtual assistants is understood by the students, but they are also aware of the efficiency they provide. Time optimization is a significant aspect of success, which again proves the need for the use of virtual assistants [14]. The overall rating of the usefulness of the virtual assistant reveals the overall positive attitude of the students, as the majority rated it high. As the minority showed negative attitudes, this again proves the previous negative attitudes, showing that there is some resistance, although limited. This again emphasizes the need for constant improvement and user-centric design to overcome the needs of all users. When it comes to programming languages, it was found that students were most interested in languages such as C and Java, followed by web technologies and then Python. This again shows that virtual assistants are considered to be of great use for students who want to learn simple and widely used programming languages and also for programmers who want to know what to focus on when they are engaged in such projects. In conclusion, the results of the research entirely confirm the main hypothesis of the significance of the implementation of a virtual assistant for IT students. In addition, the results of the research entirely confirm the auxiliary hypotheses, which prove the effectiveness of existing faculty resources and platforms as a model for the implementation of new modalities for digital learning support. This confirms the logical evolution of the existing infrastructure for the dissemination of knowledge and the effectiveness of virtual assistants for this purpose.

V. CONCLUSION

The involvement of virtual assistants in the programming learning process is a major move from traditional approaches to a more dynamic and student-centric digital environment. This study has demonstrated that the role of AI-based tools is not limited to providing technical support; it is, in fact, a tool for professional orientation and the creation of a student's academic image. The involvement of virtual assistants in providing real-time feedback and simplifying complex programming issues creates an environment in which students can freely explore their professional ambitions and develop the specific skills needed in the modern IT industry.

The results obtained in this research imply that the success of these tools depends on their ability to connect theoretical knowledge with its application, especially in complex domains like debugging and optimization. This is also in line with general ideas in modern education that emphasize equal opportunities for all participants to utilize their potential and skills. Moreover, the high interest among students in using these assistants in learning languages like C and Java points to a need for more flexible and dynamic learning tools that can cope with the dynamic nature of information technologies development. However, the process of transitioning to an entirely integrated AI-supported curriculum is not without considering the student perspective, as well as the professor's role in the process. While electives promote academic freedom and professional focus, the integration of virtual assistants is meant to promote autonomy, motivation, and eliminate the time-consuming process associated with troubleshooting programming issues.

Future research directions for this particular topic would involve the long-term implications for the career development of IT students, as well as the development of standardized strategies for the seamless integration of such tools within the context of higher learning. At this particular juncture, the authors are concerned with the foundational need for the system, while the actual development of a specific program is left for future research, considering the dynamic evolution of the particular field of AI and its application within education.

REFERENCES

- [1] C. Tsai, Y. Ma, Y. Chang, and Y. Lai, "Integrating multiculturalism into artificial intelligence-assisted programming lessons: Examining inter-ethnicity differences in learning expectancy, motivation, and effectiveness," *Frontiers in Psychology*, vol. 13, 2022. <https://doi.org/10.3389/fpsyg.2022.868698>.
- [2] A. Veljić, S. Maričić, and G. Jocić, "The importance of elective courses for IT students in the function of professional guidance," in *Proc. 12th Int. Conf. Electrical, Electronics and Computer Engineering (IcETran)*, Čačak, Serbia, June 9-12, 2025, pp. 66–71.
- [3] Y. Huang, "Enhancing computational thinking and programming logic skills with App Inventor 2 and robotics: Effects on learning outcomes, motivation, and cognitive load," *Sensors*, vol. 25, no. 22, p. 7059, 2025. <https://doi.org/10.3390/s25227059>.
- [4] A. Yusuf and N. Noor, "Research trends on learning computer programming with program animation: A systematic mapping study," *Comput. Appl. Eng. Educ.*, vol. 31, no. 6, pp. 1552–1582, 2023. <https://doi.org/10.1002/cae.22659>.
- [5] X. Zhong and Z. Zhan, "An intelligent tutoring system for programming education based on informative tutoring feedback: system development, algorithm design, and empirical study," *Interact. Technol. Smart Educ.*, vol. 22, no. 1, pp. 3–24, 2024. <https://doi.org/10.1108/itse-09-2023-0182>.
- [6] L. Manchulenko, N. Nosovets, H. Chorna, O. Fonariuk, and T. Turbar, "Constructivism in the training of students of pedagogical specialties," *Laplace Em Revista*, vol. 7, no. Extra-B, pp. 57–67, 2021. <https://doi.org/10.24115/s2446-622020217extra-b884p.57-67>.
- [7] M. Faridi and S. Izadpanah, "The study of EFL learners' perception of using e-learning, self-regulation and constructivism in English classrooms: Teachers, intermediate and advanced learners' attitude," *J. Lang. Educ.*, vol. 10, no. 2, pp. 45–58, 2024. <https://doi.org/10.17323/jle.2024.12492>.
- [8] C. Tsai, Y. Ma, Y. Chang, and Y. Lai, "Integrating multiculturalism into artificial intelligence-assisted programming lessons: Examining inter-ethnicity differences in learning expectancy, motivation, and effectiveness," *Frontiers in Psychology*, vol. 13, 2022. <https://doi.org/10.3389/fpsyg.2022.868698>.

- [9] H. Do, N. Bich, and M. Nguyen, "How do constructivism learning environments generate better motivation and learning strategies? The design science approach," *Heliyon*, vol. 9, no. 12, p. e22862, 2023. <https://doi.org/10.1016/j.heliyon.2023.e22862>.
- [10] G. Zhao, L. Yang, B. Hu, and J. Wang, "A generative artificial intelligence (AI)-based human-computer collaborative programming learning method to improve computational thinking, learning attitudes, and learning achievement," *J. Educ. Comput. Res.*, vol. 63, no. 5, pp. 1059–1087, 2025. <https://doi.org/10.1177/07356331251336154>.
- [11] M. Zhu, "The sentiments and the impact of ChatGPT on computer programming learning: Data mining from comments on youtube videos," *J. Comput. Assist. Learn.*, vol. 41, no. 2, 2025. <https://doi.org/10.1111/jcal.70013>.
- [12] T. Xie, Y. Zhou, and J. Yu, "Ce-LLMs: Status and trends of education-specific large language models developed in China," *Future in Educational Research*, vol. 3, no. 3, pp. 505–525, 2025. <https://doi.org/10.1002/fer3.70008>.
- [13] C. Cohn, C. Snyder, J. Fonteles, T. Ashwin, J. Montenegro, and G. Biswas, "A multimodal approach to support teacher, researcher and AI collaboration in STEM +C learning environments," *Brit. J. Educ. Technol.*, vol. 56, no. 2, pp. 595–620, 2024. <https://doi.org/10.1111/bjet.13518>.
- [14] K. Thompson, K. Chandler, C. Morgan, D. Khashabi, E. Delinski, and B. Durme, "Artificial intelligence as a co-tutor: Assessing the impact in the advanced learning virtual classroom," *J. Adv. Acad.*, vol. 36, no. 4, pp. 714–745, 2025. <https://doi.org/10.1177/1932202x251356323>.
- [15] S. Hauske and O. Bendel, "How can GenAI foster well-being in self-regulated learning?," *AAAI-SS*, vol. 3, no. 1, pp. 354–361, 2024. <https://doi.org/10.1609/aaaiss.v3i1.31234>.
- [16] M. Mamun, "Fostering self-regulation and engaged exploration during the learner-content interaction process: the role of scaffolding in the online inquiry-based learning environment," *Interact. Technol. Smart Educ.*, vol. 19, no. 4, pp. 482–509, 2022. <https://doi.org/10.1108/itse-11-2021-0195>.
- [17] M. Marković, N. Ilić, A. Kresović Kolundžić, M. Djurovic-Petrovic, and J. Lozanović Šajić, *Multimedia Systems in Education*. Niš: Complex System Research Centre: Mathematical Institute of SASA, 2022, vol. 4, pp. 79–83.
- [18] N. Lettia, M. M. Mosia, and F. O. Egara, "The role of AI-driven personalised learning in enhancing mathematics problem-solving skills: A systematic review," *Frontiers in Computer Science*, vol. 8, 2026. <https://doi.org/10.3389/fcomp.2026.1813431>.
- [19] J. Allison, "RAISE the standard: A framework for transparent reporting of artificial intelligence studies in education," *J. Educ. Comput. Res.*, vol. 64, no. 1, pp. 3–15, 2025. <https://doi.org/10.1177/07356331251377430>.
- [20] D. K. Mutyala, "Artificial intelligence trajectory of impacts, challenges, and opportunities in education," *Int. J. Comput. Trends Technol.*, vol. 73, no. 12, pp. 40–44, 2025. <https://doi.org/10.14445/22312803/IJCTT-V73I12P106>.