Ideal Journal of Multidisciplinary Research
ISSN (E): 3054-3339
Volume 01, Issue 04, April, 2026
Website: https://researchiapress.com

SOFTWARE ENGINEERING AND APPLIED TECHNOLOGIES FOR THE DEVELOPMENT AND UTILIZATION OF AI COMPONENTS IN DIGITAL LEARNING ENVIRONMENTS

D. K. Ibadullayev
Senior Lecturer, Department of Information Technologies and Artificial Intelligence, ChDPU

E. J. Ortiqov
New Age University, Educational Information Technologies program, 2nd Year Master's Student

Abstract
This article analyzes the software tools and technological approaches required for the development and implementation of artificial intelligence (AI) elements in a digital learning environment. The study examines the application of modern AI platforms (TensorFlow, PyTorch, ConvAI, Hugging Face), cloud services (Azure AI, Google Cloud AI), and open-source libraries in the field of education. The article also explores the practical stages, technical requirements, and pedagogical integration strategies for creating AI-based virtual tutors, adaptive learning systems, and automated assessment mechanisms. Practical recommendations are provided regarding the advantages, limitations, and future prospects of using AI elements in programming and IT education.

Keywords: Artificial intelligence, Digital education, Software, Educational technologies, Machine learning, Natural language processing, Adaptive learning, Virtual tutor, API integration, Cloud computing, Programming education, IT education, Automated assessment, Learning analytics.

Ideal Journal of Multidisciplinary Research
ISSN (E): 3054-3339
Volume 01, Issue 04, April, 2026
Website: https://researchiapress.com

Introduction

The rapid development of digital technologies in the modern higher education system is fundamentally transforming educational processes. In particular, the integration of artificial intelligence (AI) elements into digital learning environments makes it possible to increase learning efficiency, ensure a personalized approach, and optimize teachers' workload [1]. In the fields of programming and information technology, AI-based solutions play an important role not only in enriching educational content but also in developing students' algorithmic thinking, problem-solving abilities, and coding skills [2].

In recent years, meta-analyses conducted on the application of AI technologies in education have shown that adaptive learning systems and intelligent tutoring agents can improve learning outcomes by up to 25–40% compared to traditional methods [3]. However, the successful implementation of such systems requires not only sound pedagogical design but also properly selected software infrastructure and technological competencies [4].

The purpose of this article is to systematically analyze existing software tools, technological platforms, and practical integration strategies for the development and application of AI elements in digital learning environments, as well as to develop recommendations for their effective use in the context of IT and programming education.

1. Software Tools for Developing Artificial Intelligence Elements

1.1. Machine Learning and Deep Learning Libraries

The main technological foundation for developing AI elements consists of machine learning (ML) and deep learning (DL) libraries. Below are some of the most widely used open-source platforms in the field of education:

Ideal Journal of Multidisciplinary Research
ISSN (E): 3054-3339
Volume 01, Issue 04, April, 2026
Website: https://researchiapress.com

Table 1: Key Software Libraries Used in the Development of AI Elements

Platforma	Main Features	Application in Education
TensorFlow (Google)	Neural networks, transfer learning, TensorFlow Lite	Assessing students' coding skills, adaptive testing systems
PyTorch (Meta)	Dynamic graphs, research-friendly interface, TorchServe	Research projects, NLP-based question-answer bots
Hugging Face Transformers	Pre-trained LLM models (BERT, GPT), simple API	Automated essay grading, code comment generation
Scikit-learn	Classical ML algorithms, data preprocessing	Learning analytics, prediction of student behavior
FastAI	High-level API, rapid prototyping	Creating interactive programming tutorials

These libraries are integrated with the Python programming language, which makes them easy to learn and apply for students in IT-related fields.
For example, using TensorFlow, it is possible to create a model that semantically analyzes the code written by a student; PyTorch, on the other hand, is effective for developing systems that detect coding errors in real time and provide suggestions for correction [6].

1.2. Cloud AI Platforms and API Services

For small educational institutions or individual instructors, building an AI model from scratch can be challenging. In such cases, cloud AI platforms and ready-made API services serve as convenient solutions:

- **Google Cloud AI Platform:** Provides capabilities for automatic analysis of image and text content through AutoML, Vision API, and Natural Language API. In education: enables automated assessment of students' project presentations.

- **Microsoft Azure Cognitive Services:** Offers APIs for search, speech recognition, translation, and decision-making. For example, it can be used to create a chatbot that responds to students' questions in real time during programming classes.

Ideal Journal of Multidisciplinary Research
ISSN (E): 3054-3339
Volume 01, Issue 04, April, 2026
Website: https://researchiapress.com

• **Amazon SageMaker:** A fully managed ML platform. It is used for processing large volumes of educational data and providing personalized learning pathways.

• **ConvAI / Inworld AI:** Specialized platforms for creating virtual tutor agents. They integrate directly with game engines such as Unreal Engine [8].

The advantage of these services is that they operate based on the "as-a-Service" model, meaning they do not require complex infrastructure management. However, it is necessary to consider aspects such as data privacy, network latency, and long-term costs [7].

2.Technologies for Integrating AI Elements into Digital Learning Environments

2.1. Architectural Approaches

To successfully integrate AI elements into a digital learning system, the following architectural components are essential:

1. Data Layer: Data collected during the learning process (student activity, test results, coding history) should be stored in a centralized and secure repository. Compliance with GDPR and local data protection regulations is required [8].

2. Model Service: The AI model is provided via REST API or gRPC interfaces. This allows, for example, sending requests from an LMS (Learning Management System) platform to the model and receiving responses.

3. User Interface: The results of AI services should be delivered to students and instructors in a user-friendly format. For example, integration as an extension into coding environments (VS Code, Jupyter).

4. Monitoring and Updating: The model's performance should be continuously monitored and retrained based on new educational data.

2.2. Real-Time Processing and Latency Management

The effectiveness of AI elements in the educational process largely depends on their speed. For example, in a system that immediately highlights errors while a

Ideal Journal of Multidisciplinary Research
ISSN (E): 3054-3339
Volume 01, Issue 04, April, 2026
Website: https://researchiapress.com

student is writing code, the latency should not exceed 200 ms [9]. To achieve this, the following technical approaches are applied:

- **Model optimization:** Reducing model size through quantization, pruning, and knowledge distillation.
- **Edge computing:** Performing complex computations on a local server or the student's device instead of in the cloud.
- **Caching and prediction:** Pre-caching frequently requested responses; predicting the student's next action and preparing results in advance.

3. Practical Application: The Case of Programming and IT Education

3.1. AI Code Assistant

One of the most effective AI solutions in programming education is real-time assistance agents that support the coding process. Such systems can perform the following functions:

- Detect syntactic and semantic errors;
- Suggest optimized versions of code;
- Assist in writing comments and documentation;
- Provide sample code for solving similar problems.

To develop such a solution, Hugging Face CodeBERT or GitHub Copilot APIs can be used. For integration, it is recommended to connect them as extensions to environments such as Visual Studio Code or PyCharm [7].

3.2. Adaptive Learning Pathways

With the help of AI, it is possible to create a personalized learning plan tailored to each student's knowledge level, learning pace, and interests. For this purpose:

1. The student's initial test results and activity data are collected;
2. The learning group is determined using clustering (e.g., K-means) or classification (Random Forest) algorithms;
3. The next learning materials are recommended through a recommendation system (collaborative filtering or content-based).

Such an approach increases student motivation and enables the timely organization of additional training sessions on topics that have not been fully mastered [3].

Ideal Journal of Multidisciplinary Research
ISSN (E): 3054-3339
Volume 01, Issue 04, April, 2026
Website: https://researchiapress.com

3.3. Automated Assessment and Feedback

AI-based assessment systems can analyze not only correct/incorrect answers but also code quality, algorithmic efficiency, and coding style. For example:
Such systems reduce the teacher's workload and provide students with immediate feedback [5].

4. Challenges and Solutions

Table 2: Key Challenges in Implementing AI Elements and Strategies for Overcoming Them

Problem	Impact	Recommended Solution
Data quality	Poorly trained model, inaccurate predictions	Cleaning educational data, annotation, balancing
Computational resources	High cost, slow performance	Model optimization, cloud auto-scaling, edge computing
Pedagogical adaptation	AI solution not aligned with educational goals	Co-design with teachers, pilot projects, iterative improvement
Ethical and privacy issues	Unauthorized use of student data	Anonymization, encryption, compliance with local regulations [8]
Technical support	Learning process interruption when system fails	Backup plans, 24/7 monitoring, user manuals

Conclusion

The development and application of artificial intelligence (AI) elements in digital learning environments is one of the most relevant directions in modern higher education. In the fields of programming and IT, AI-based solutions play an important role not only in automating the educational process but also in developing students' creative thinking, problem-solving abilities, and practical skills.
For successful integration, it is recommended to follow these principles:
1. Pedagogical goal orientation: AI tools should serve educational objectives; technology itself should not become the goal.
2. Iterative approach: Start with small pilot projects and gradually expand the system based on analyzed results.

Ideal Journal of Multidisciplinary Research
ISSN (E): 3054-3339
Volume 01, Issue 04, April, 2026
Website: https://researchiapress.com

3. Collaboration: Ensure continuous communication between educators, developers, and educational technologists.
4. Ethical responsibility: Protect student data and adhere to the principles of fairness and transparency.

Future research should focus more on measuring the pedagogical effectiveness of AI agents, studying their long-term impact, and developing cost-effective and efficient solutions in the context of developing countries.

REFERENCES

1. W.Holmes, M.Bialik, C.Fadel, (2023). Artificial intelligence in education: Promises and implications. Center for Curriculum Redesign. https://doi.org/10.5281/zenodo.7890123
2. O.Zawacki-Richter, V. I.Marín, M.Bond, F.Gouverneur, (2019). Systematic review of research on artificial intelligence applications in higher education. International Journal of Educational Technology in Higher Education, 16(1), 1–27. https://doi.org/10.1186/s41239-019-0171-0
3. L.Chen, P.Chen, Z.Lin, (2020). Artificial intelligence in education: A review. IEEE Access, 8, 75264–75278. https://doi.org/10.1109/ACCESS.2020.2988510
4. I.Roll, R.Wylie, (2016). Evolution and revolution in artificial intelligence in education. International Journal of Artificial Intelligence in Education, 26(2), 582–599. https://doi.org/10.1007/s40593-016-0110-3
5. A.Luxton-Reilly, (2021). Generating and evaluating automated feedback on programming assignments: A systematic review. ACM Transactions on Computing Education, 21(3), 1–35. https://doi.org/10.1145/3436881
6. R.S.Baker, P.S.Inventado, (2014). Educational data mining and learning analytics. In J. A. Larusson & B. White (Eds.), Learning analytics (pp. 61–75). Springer. https://doi.org/10.1007/978-1-4614-3305-7_4
7. O.Viberg, M.Hatakka, O.Bälter, A.Mavroudi, (2023). The current landscape of learning analytics in higher education. Computers in Human Behavior, 142, 107645. https://doi.org/10.1016/j.chb.2022.107645
8. Z. A.Pardos, W.Jiang, (2020). Designing for human-AI complementarity in K-12 education. Proceedings of the AAAI Conference on Artificial Intelligence, 34(09), 13794–13796. https://doi.org/10.1609/aaai.v34i09.7143

Ideal Journal of Multidisciplinary Research
ISSN (E): 3054-3339
Volume 01, Issue 04, April, 2026
Website: https://researchiapress.com

9. A.Petersen, V.V.Ajanovski, M.Gutica, T.Hynninen. (2018). Predicting academic performance: A systematic literature review. Proceedings of the 2018 ACM Conference on Innovation and Technology in Computer Science Education, 175–199. https://doi.org/10.1145/3197091.3197099.