Pinnacle Academic Press Proceedings Series

Vol. 2 2025

Article **Open Access**

Research on the Design of Modern Distance Education System Based on Agent Technology

Le Xiu [1,*]

[1] University of California, Los Angeles, California, 90024, United States
* Correspondence: Le Xiu, University of California, Los Angeles, California, 90024, United States

Received: 14 April 2025
Revised: 20 April 2025
Accepted: 10 May 2025
Published: 14 June 2025

Abstract: With the rapid development of information technology, modern distance education systems have gradually become an important part of the education field. However, traditional distance education systems face several shortcomings in personalized learning, real-time interaction, and intelligent teaching. This paper proposes a modern distance education system design based on Agent technology, aiming to achieve intelligent management of learning resources, personalized teaching recommendations, and real-time monitoring of teacher-student interaction through the autonomy, collaboration, and adaptability of intelligent Agents. The paper first analyzes the current needs of distance education systems, then designs an Agent-based system architecture, and describes the core algorithms of task allocation, intelligent recommendation, and collaboration. Finally, the system's performance was tested and evaluated, suggesting potential improvements in learning outcomes and user experience. This research provides new insights and technical support for the development of modern distance education systems.

Keywords: agent technology; distance education system; personalized recommendation; intelligent teaching; task allocation algorithm

1. Introduction

With the rapid advancement of information technology and the internet, modern distance education systems have emerged globally, offering learners more convenient and flexible methods of accessing education. However, traditional systems face limitations in personalized learning, real-time interaction, and intelligent teaching, making it difficult to meet diverse learner needs. Agent technology, with its autonomy, collaboration, and adaptability, has become a key solution. It enables intelligent management of the teaching process, personalized content recommendations, and optimized task allocation, improving learning efficiency and real-time interaction between teachers and students. This paper designs a modern distance education system based on Agent technology, analyzing its needs and proposing an architecture that incorporates key algorithms for task allocation, intelligent recommendations, and collaboration. Performance analysis confirms the system's effectiveness in enhancing teaching outcomes and user experience, offering valuable technical support for the development of intelligent distance education systems [1].

2. Overview of the Design of Modern Distance Education System Based on Agent Technology

As education needs diversify and become more personalized, traditional distance education systems reveal several shortcomings in providing personalized teaching, real-time feedback, and intelligent learning support. To address these challenges, the design of modern distance education systems based on Agent technology has become a research focus [2]. Figure 1 illustrates the overall architecture of the system, aiming to enhance the learning experience of learners by achieving efficient teaching content management, intelligent question generation, and personalized learning path recommendation through the collaboration of intelligent Agents.

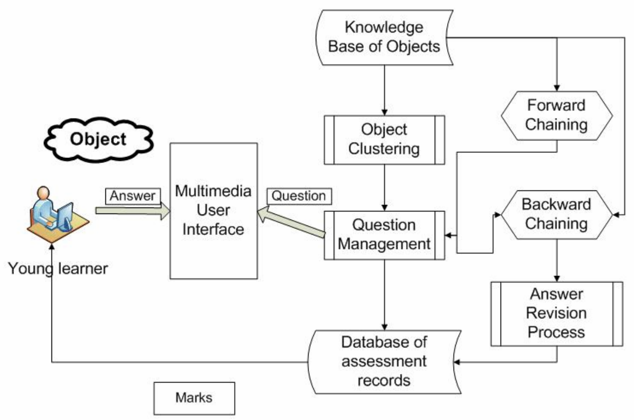

Figure 1. Architecture Diagram of an Intelligent Distance Education System Based on Agent Technology.

The core workflow revolves around a Multimedia User Interface, which bridges learners with the system, enabling learning activities and forming a feedback loop. The system collects and analyzes learners' responses in real-time to optimize learning paths and content recommendations. The system's Object-Oriented Knowledge Base stores learning materials and resources. An Object Clustering Module organizes these resources, improving the system's ability to quickly retrieve and present relevant content, ensuring efficiency even during large-scale data inquiries. Intelligent reasoning is achieved through Forward Chaining and Backward Chaining mechanisms. Forward chaining generates appropriate questions and content based on existing resources, while backward chaining starts from the learner's goals and works backward to ensure they achieve their objectives quickly. These reasoning methods help the system flexibly adapt to different learning needs, especially when learners require personalized guidance. A Question Management Module personalizes the learning experience by adjusting the difficulty of questions based on learner profiles. All responses are stored in a Database of Assessment Records, which supports future evaluations, answer revisions, and system optimization. The Answer Revision Process refines the learning process by adjusting the learning path and recommendations based on student performance, ensuring continuous improvement and correction of issues. Through these modules, the Agent-based system supports intelligent reasoning, dynamic content adjustments, and personalized recommendations. This design reduces teacher workload, enhances teaching efficiency, and sets a strong foundation for the future of intelligent distance education [3].

3. Analysis of the Needs of Modern Distance Education Systems

3.1. The Current Situation and Problems of Modern Distance Education Systems

Currently, most distance education systems rely on standardized course settings and resource distribution. While this model can cover a wide range of learners, it exposes several issues in practical applications, mainly focused on the following aspects: lack of personalized teaching, insufficient real-time interaction, weak intelligent learning support, inefficient task allocation and collaboration management, disorganized learning resource management, and problems with system scalability and stability [4]. Table 1 below summarizes these situations and problems, serving as a basis for system design and improvements.

Table 1. Requirements Analysis.

Category of Need	Current Situation	Existing Problems
Personalized Teaching	Most distance education systems offer standardized courses, lacking personalized learning paths.	Unable to provide personalized learning content based on students' progress and needs, reducing motivation.
Real-time Interaction	Existing systems typically offer recorded or static content with limited real-time interaction options.	Students cannot get timely feedback and guidance when encountering problems, reducing learning effectiveness.
Intelligent Support	The teaching content is mostly preset, lacking intelligent process management and automated recommendations.	The system cannot perform intelligent recommendations through data analysis and fails to optimize learning paths dynamically.
Task Allocation	Current systems mostly provide a single task allocation mechanism, primarily reliant on manual assignment by teachers.	Low task allocation efficiency and heavy teacher workload make it difficult to adjust tasks dynamically.
Learning Monitoring	Most systems rely on exam results for evaluation, lacking comprehensive tracking of the learning process.	The evaluation mechanism is limited, not reflecting students' overall learning outcomes, and lacks behavioral feedback.
Resource Management	Distance education systems typically provide abundant resources but lack efficient organization.	Students find it difficult to locate the necessary resources quickly, leading to lower learning efficiency and poor resource utilization.
Scalability	Current systems often lack flexibility, making it difficult to scale or personalize configurations.	As user numbers and demands grow, the system's expansion becomes slow, affecting response speed and user experience.
Teacher-Student Interaction	The interaction methods are fixed, mainly through traditional assignments and exams.	The lack of diverse interaction methods fails to meet the varied needs of students at different learning stages.
Diverse Learning Needs	Existing systems predominantly use a standardized teaching model, struggling to meet learners' varied needs.	The system lacks personalized support for different learner groups (e.g., children, adults, professional learners).
Data Collection	Some systems can collect partial learner data, but analysis capabilities are limited.	The data collection scope is narrow, making it difficult to provide personalized teaching effectively.
Cross-platform Compatibility	Many distance education systems can only run on specific devices or operating systems, lacking cross-platform compatibility.	It struggles to adapt to the variety of devices learners use, affecting convenience and continuity of learning.

The current distance education systems overly depend on standardized teaching models, making it difficult to design personalized learning paths based on each learner's progress and needs. This results in learning content that does not accurately match students' levels, hindering learners from acquiring relevant and targeted knowledge, ultimately reducing learning effectiveness. Therefore, the system needs to introduce intelligent learning path recommendation features that dynamically adjust learning content based on student behavior and historical data to meet personalized learning needs.

Most traditional distance education systems rely on recorded courses and lack real-time teacher-student interaction, making it difficult for students to receive timely feedback during the learning process, thereby reducing students' motivation and effectiveness. Future distance education systems should enhance teacher-student communication through dedicated real-time interaction modules. Additionally, intelligent Q&A features should be incorporated to help students quickly resolve learning difficulties and receive immediate feedback [5]. Existing systems generally lack intelligent analysis and automated recommendation features, making it difficult to adjust teaching strategies based on student performance. Therefore, the introduction of Agent technology, through intelligent analysis of students' learning data and habits, allows personalized learning resources to be recommended, effectively improving learning outcomes. Intelligent Agents can assist with certain repetitive teaching tasks, helping to reduce teacher workload while providing more targeted learning guidance.

The task allocation model of traditional systems is too rigid, relying primarily on manual allocation by teachers, making it inefficient and difficult to adjust tasks flexibly. An Agent-based distance education system needs to have intelligent task allocation capabilities, dynamically adjusting tasks based on students' learning status and task completion, ensuring that each student receives tasks suitable for their learning stage. Moreover, the collaboration mechanism between teachers and students should be supported by intelligent systems that facilitate task coordination, communication, and feedback, thereby improving collaboration efficiency and interaction quality. In conclusion, the needs of modern distance education systems focus on personalization, intelligence, and interaction. Future distance education systems should introduce intelligent technologies to enhance adaptability and user experience, ultimately achieving an efficient, flexible, and intelligent learning model [6].

3.2. User Needs Analysis and Functional Design

Modern distance education systems serve students, teachers, and administrators, each with distinct needs. Therefore, system design must optimize for these user groups to enhance teaching effectiveness and the overall learning experience. By analyzing user needs, the system should offer personalized learning support, intelligent management, real-time feedback, and progress monitoring to create a flexible, efficient education experience.

Students, the core users, need personalized learning paths, intelligent guidance, and real-time interaction. They expect dynamic content adjustments based on their progress, along with personalized learning plans generated by intelligent Agents. Real-time Q&A support, instant messaging, online discussions, and virtual classrooms are essential for interactive learning. An automated evaluation system should provide real-time feedback on assignments to help students adjust their learning strategies. "Teachers" focus on efficient content management, student progress monitoring, and task allocation. The system should include a flexible teaching content management module and an intelligent task allocation module that assigns learning tasks based on students' progress [7]. Teachers also need real-time tracking of student performance through a progress monitoring module, enabling them to adjust teaching strategies based on detailed data reports. "Administrators" require comprehensive monitoring and data analysis of system performance, student outcomes, and teacher effectiveness. A data analysis module should generate reports on learning performance and system usage, offering administrators key insights for decision-making. As user numbers grow, scalability and stability are crucial, with dynamic load balancing ensuring smooth system performance under high concurrency. The system's core modules (Personalized Learning, Intelligent Task Allocation, Real-time Interaction, Progress Monitoring, and Data Analysis) work together to meet the needs of all users. These functions enhance learning efficiency, reduce teacher workload, and support system optimization, paving the way for a more intelligent and personalized distance education model [8].

4. Architecture Design of the Agent-Based Distance Education System

The architecture of the Agent-based distance education system operates through the collaboration of various functional modules, comprehensively covering the needs of students, teachers, and system administrators, creating a highly integrated and intelligent distance education platform. Figure 2 illustrates the overall architecture of the system, which not only ensures smooth teaching activities but also dynamically adjusts system functions based on user behavior to provide personalized services for different user groups [9].

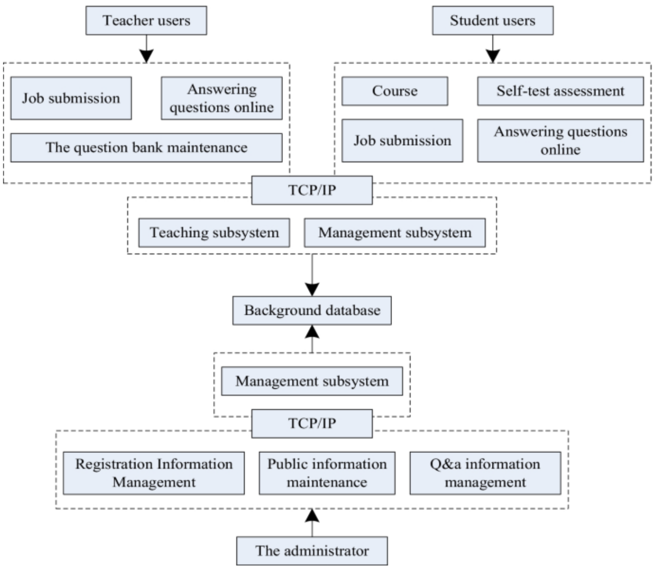

Figure 2. Functional Architecture Diagram of the Agent-Based Distance Education System.

The architecture supports three main user groups: Teachers, Students, and Administrators. Each user group interacts with its respective subsystem, and all modules are interconnected via TCP/IP for efficient data transmission. The Teacher User Module allows teachers to manage assignments, answer questions, and maintain the question bank. These tasks are streamlined through the Teaching Subsystem, which is connected to a backend database. Teachers can adjust question difficulty dynamically based on student performance, providing personalized evaluations. The Student User Module focuses on student learning needs, enabling course access, self-assessments, assignment submissions, and real-time feedback on test results. Self-assessment tools assist in identifying individual learning gaps and generating tailored recommendations, with all related data being synchronized in real-time. The Management Subsystem oversees registration, public information, and Q&A management, ensuring all registration data is accurately entered and maintained. Administrators use this system to monitor system activity and ensure smooth information flow and ongoing system maintenance. The Backend Database lies at the core of the system, supporting concurrent operations while ensuring data security and integrity. The TCP/IP Protocol facilitates stable, real-time communication across all system components, ensuring scalability and stability. In summary, this architecture offers a personalized, flexible, and efficient distance education system. It provides intelligent task management, real-time learning recommendations, and secure data handling, significantly improving the overall quality and effectiveness of distance education [10].

5. Algorithm Design and Optimization

In the Agent-based distance education system, algorithm design and optimization are central to achieving intelligent learning, personalized recommendations, and dynamic task allocation. By introducing and optimizing various algorithms, the system can intelligently analyze learner behavior, make real-time decisions, and dynamically adjust learning content based on each user's needs. The primary goal of these algorithms is to enhance system efficiency and precision, improving learning outcomes and user experience. To provide personalized learning paths and resource recommendations, the system needs to analyze students' learning behavior in real-time. This analysis can be described by the following formula 1:

$$S(t) = \alpha \cdot P(t) + \beta \cdot E(t) + \gamma \cdot L(t) \quad (1)$$

Where $S(t)$ represents the student's comprehensive learning score at time t, $P(t)$ represents the student's current learning progress, $E(t)$ reflects the student's performance in assessments, based on accuracy and difficulty of responses, $L(t)$ represents the student's learning duration or engagement level, α、β、and γ are weight parameters indicating the influence of each factor on the overall score. This algorithm analyzes student behavior data through weighted aggregation, generating a personalized learning score $S(t)$. Based on this score, the system dynamically recommends subsequent learning content or tasks for the student.

For instance, if a student performs poorly in a certain knowledge area, the system prioritizes recommending supplementary learning materials on that topic. If the student's learning time is short and progress is slow, the system may recommend an appropriate learning plan to improve efficiency. Through continuous adjustment and optimization of the weight parameters α、β、and γ, the system can dynamically adapt to different students' learning habits and paces, ensuring the precision of personalized recommendations. In distance education systems, intelligent task allocation is crucial for enhancing learning efficiency and maintaining student engagement. The intelligent Agent dynamically adjusts assigned learning tasks and assessment content based on the student's learning progress and performance. This task allocation algorithm primarily determines task difficulty and priority based on students' historical data and current status. Task allocation can be handled through the Task Priority Formula 2:

$$T_i = \lambda \cdot D_i + \mu \cdot R_i \quad (2)$$

Where T_i is the priority score of task i, D_i is the difficulty coefficient of task i, R_i is the student's rating in that task area, λ and μ are the weights for difficulty and performance rating, respectively. The system dynamically adjusts students' task allocation based on each task's priority score T_i, ensuring the task's difficulty matches the student's current abilities. For example, if a student excels in a particular area, the system will assign more challenging tasks, and vice versa. This dynamic task allocation ensures students remain appropriately challenged, thereby avoiding the adverse effects of tasks that are mismatched in difficulty, which can hinder learning outcomes. Learning path optimization is essential for ensuring that students effectively master knowledge in the shortest possible time. The intelligent Agent analyzes students' current knowledge levels, learning progress, and performance feedback to construct an optimal learning path. Learning path optimization can draw inspiration from the Bellman-Ford Algorithm used in shortest path problems, with knowledge points modeled as graph nodes and learning transitions as weighted edges. Dynamic programming techniques can then be applied to determine the optimal sequence of content acquisition.

The basic formula for the Bellman-Ford Algorithm as shown in Formula 3:

$$d(v) = \min(d(u) + w(u, v)) \quad (3)$$

Where $d(v)$ represents the minimum learning cost for reaching knowledge point v, u is the previous knowledge point, $w(u, v)$ is the learning cost from point u to v, typically measured by task completion time or assessment performance. The system uses this path optimization algorithm to determine which knowledge points students should prioritize, forming the shortest learning path and ensuring students can master all necessary knowledge in the shortest time. By introducing dynamic programming concepts, the system can adjust learning paths based on real-time feedback, ensuring that each learner receives a personalized and efficient learning experience. Evaluation and feedback are critical components of providing personalized learning guidance for students. Based on students' assessment performance, the system uses a Weighted Feedback Algorithm to offer detailed learning suggestions and future learning plans. This algorithm considers factors such as assessment accuracy, task completion speed, and question difficulty. The basic formula for evaluation feedback as shown in Formula 4:

$$F(t) = \frac{\sum_{i=1}^{n}(A_i \cdot W_i)}{n} \quad (4)$$

Where $F(t)$ represents the feedback score for the student at time t, A_i reflects the student's performance in assessment i, W_i is the weight of assessment i, indicating its importance in overall learning, n is the total number of assessments. This algorithm generates a comprehensive feedback score $F(t)$ for the student and, based on this score, recommends corresponding future learning resources. For example, if a student performs poorly in a specific assessment, the system may suggest revisiting relevant knowledge points and adjust the learning path in real time through the feedback system. To ensure efficient system operation under high user concurrency, algorithm optimization is essential.

The following strategies are employed:

1) Parallel Computing: By leveraging parallel processing, the system can improve task handling efficiency during simultaneous multi-user access, ensuring real-time algorithm responsiveness.
2) Dynamic Weight Adjustment: The system can dynamically adjust the weight parameters in algorithms based on the historical data of different students, ensuring personalized and accurate recommendations and learning paths.
3) Caching Mechanism: Frequently queried data is cached to reduce database access, improving system response time and enhancing overall user experience.
4) Data Preprocessing: Before executing the algorithms, the system uses data cleaning and feature extraction techniques to minimize noise, improving the accuracy and robustness of the algorithms. Through these optimization strategies, the system can maintain high performance while ensuring accuracy, even when managing a large number of users.

In summary, By introducing and optimizing algorithms for recommendation, task allocation, learning path planning, and feedback evaluation, the Agent-based distance education system can dynamically adapt to individual learning needs. Continuous optimization of these algorithms enhances the system's intelligence, improving learning outcomes for students while providing teachers and administrators with precise data analysis and decision support.

6. System Performance Analysis and Evaluation

To evaluate the performance of the Agent-based distance education system, several experiments were conducted, covering system response speed, task allocation efficiency, learning path recommendation accuracy, and user satisfaction. The experiments aimed to assess the system's performance before and after algorithm optimization. The core objective was to evaluate system responsiveness under high user concurrency, the precision of task allocation and recommendation algorithms, and user feedback in practical educational scenarios. The evaluation process was structured into three distinct phases:

6.1. Phase One: Response Speed Test

This phase measured the system's response time under different user loads, focusing on handling various types of requests. The experiment tracked response speeds for different operations, such as loading learning resources, submitting assignments, and receiving quiz feedback. The system was tested with 100, 500, and 1000 users concurrently online. The recorded data is shown in the Table 2 below:

Table 2. Response Speed Test.

Concurrent Users	Course Resource Load Time (seconds)	Assignment Submission Response Time (seconds)	Quiz Feedback Response Time (seconds)
100	0.8	1.2	0.9
500	1.3	1.8	1.5
1000	2.1	2.6	2.3

The data shows that as the number of concurrent users increases, system response time decreases. However, even at 1000 users, the response time remained within the acceptable threshold of 3 seconds, indicating that the system can maintain acceptable performance under high load conditions, and assignment submission and quiz feedback times similarly lengthened. However, the system maintained a reasonable response time overall, especially with 500 users, where response times stayed under 2 seconds. Based on these results, further improvements can be made by introducing parallel computing and caching optimization to enhance system scalability under high concurrency.

6.2. Phase Two: Task Allocation and Learning Path Recommendation Algorithm Test

In this phase, 100 students were selected to participate in task allocation and learning path recommendation testing. After completing the recommended learning tasks, the system dynamically adjusted subsequent learning content based on student performance, leading to more targeted practice and reducing redundant or overly difficult content. The following Table 3 presents the accuracy evaluation of the learning path recommendation algorithm:

Table 3. Accuracy Evaluation Data.

Student ID	Task Completion Rate (%)	Learning Path Adaptability Score (out of 10)
01	92	9.5
02	88	8.8
03	94	9.3
04	87	8.7
05	90	9.0

The data shows an average task completion rate of 90.2%, and the adaptability score for recommended learning paths averaged 9.1 out of 10. This indicates that the intelligent Agent performed with high accuracy in task allocation and learning path recommendation. Most students reported that the difficulty level of tasks was appropriate and that the recommended learning paths aligned well with their individual progress, enhancing continuity and personalization in their learning. The system efficiently adjusted its recommendation strategy based on real-time student data, improving learning outcomes.

6.3. Phase Three: User Satisfaction and Learning Effectiveness Evaluation

In the final phase, the system collected feedback and satisfaction surveys from the students who participated in the test. After using the system, students completed a satisfaction survey and provided subjective evaluations of the system's impact on their learning effectiveness. The experiment data is summarized in the Table 4 below:

Table 4. Summary of Experimental Data.

Student ID	Learning Effectiveness Improvement (out of 10)	User Satisfaction Score (out of 10)
01	9.2	9.0
02	8.5	8.8

03	9.0	9.3
04	8.7	8.7
05	9.3	9.2

The average score for perceived learning effectiveness improvement — measured through self-reported understanding, task completion ease, and knowledge retention — was 8.94, with user satisfaction averaging 8.9. Most students reported that the system's personalized recommendations and real-time feedback significantly improved learning efficiency, while the intelligent task allocation minimized redundant efforts and enhanced learning efficiency, enhancing the overall learning experience. The high user satisfaction scores demonstrate that the system was well-received in terms of usability, response speed, and personalized recommendations. Through comprehensive performance testing and analysis, the experiment results show that the Agent-based distance education system maintains high response speed under large-scale concurrent usage, with task allocation and learning path recommendation algorithms showing high accuracy and adaptability. Students' learning outcomes significantly improved, and user satisfaction was high. To mitigate the performance degradation observed under higher user concurrency, the system can be further optimized through caching mechanisms and algorithm improvements. Overall, the Agent-based distance education system provides strong support for enhancing student learning experience and efficiency, effectively meeting the needs of diverse learners.

7. Conclusion

This study designed and validated an Agent-based distance education system. Through intelligent algorithms, the system effectively delivers personalized task allocation, learning path recommendations, and feedback evaluation, improving both learning efficiency and user experience. The experimental results show that the system maintains high response speed under large-scale concurrent access, with task allocation and recommendation algorithms demonstrating good accuracy. Although response speed decreased with higher concurrent users, further optimization can enhance system performance. In summary, the Agent-based distance education system provides an intelligent, adaptive, and data-driven solution tailored to the evolving demands of modern education, contributing to improved learning outcomes and teaching quality.

References

1. R. Li, "An artificial intelligence agent technology based web distance education system," *J. Intell. Fuzzy Syst.*, vol. 40, no. 2, pp. 3289–3299, 2021, doi: 10.3233/JIFS-189369.
2. M. Garlinska, K. Musial, J. Kurek, J. Marcinkiewicz, P. Domanski, and A. Puczynski, et al., "The influence of emerging technologies on distance education," *Electronics*, vol. 12, no. 7, p. 1550, 2023, doi: 10.3390/electronics12071550.
3. S. Li and W. Han, "Design of distance education system for ideological and political courses of electronic engineering specialty based on big data," *J. Interconnect. Netw.*, vol. 22, suppl. 05, p. 2147009, 2022, doi: 10.1142/S0219265921470095.
4. S. Xie and J. Xu, "Design and implementation of physical education teaching management system based on multi-agent model," *Int. J. Comput. Intell. Syst.*, vol. 16, no. 1, p. 172, 2023, doi: 10.1007/s44196-023-00349-9.
5. Y. Huang, "Design of personalised English distance teaching platform based on artificial intelligence," *J. Inf. Knowl. Manag.*, vol. 21, suppl. 02, p. 2240017, 2022, doi: 10.1142/S0219649222400172.
6. O. Barabash, A. Shestakov, L. Barabash, D. Yankovoy, I. Konovalova, and S. Ivanova, et al., "Comprehensive methods of evaluation of efficiency of distance learning system functioning," *Int. J. Comput. Netw. Inf. Secur.*, vol. 14, no. 1, pp. 16–23, 2021, doi: 10.5815/ijcnis.2021.01.02.
7. O. Barabash, A. Shestakov, L. Barabash, D. Yankovoy, I. Konovalova, and S. Ivanova, et al., "Comprehensive methods of evaluation of distance learning system functioning," *Int. J. Comput. Netw. Inf. Secur.*, vol. 13, no. 3, pp. 62–71, 2021, doi: 10.5815/ijcnis.2021.03.06.
8. S. de O. Durso and E. P. Arruda, "Artificial intelligence in distance education: A systematic literature review of Brazilian studies," *Probl. Educ. 21st Century*, vol. 80, no. 5, pp. 679–692, 2022, doi: 10.33225/pec/22.80.679.
9. A. M. M. Mahmoud, "An agent-based framework for providing security in a cloud-based E-learning system," *Int. J. Adv. Appl. Sci.*, vol. 7, pp. 19–24, 2020, doi: 10.21833/ijaas.2020.07.003.

10. Y. Liu and M. Shabaz, "Design and research of computer network micro-course management system based on JSP technology," *Int. J. Syst. Assur. Eng. Manag.*, vol. 13, suppl. 1, pp. 203–211, 2022, doi: 10.1007/s13198-021-01368-5.