The Agent-Based Learning Platform

Natalia Axak[a], Maksym Kushnaryov[a], Andrii Tatarnykov[a]

[a] *Kharkiv National University of Radio Electronics, Nauky Ave. 14, Kharkiv, 61166, Ukraine*

Abstract

A model of an agent-based learning platform (ALP) has been developed, in which, based on the combination of agent technology and computer vision, an approach is proposed that allows combining collective control, coordination and cooperation of agents to provide online educational services. The proposed approach allows, firstly, students to be aware of the progress of their own learning activities, and secondly, to notify parents in the event of a negative incident, i.e., when their child does not attend classes according to the schedule established in this educational institution, or this student received a score based on the results of the final/semester academic performance check in accordance with the schedule of the educational process less than the minimum allowable. The provision of such services is realized through the student behavior monitoring subsystem and the community of intelligent agents.

A software prototype of an agent environment for integrating distributed agents into a single information space that combines the Moodle learning platform, a university web portal, a Student behavior monitoring subsystem developed by the Student and Amazon S3 cloud storage was developed.

Keywords

Learning management system, monitoring, Moodle, learning process, intelligent agents, multiagent system, JSON format

1. An introduction to achievements of multi-agent systems in education

The development of computing systems and the Internet of Things (IoT) has led to the ubiquitous distribution of distributed computing, embodying the successful interaction of disparate objects. In this regard, we see the widespread distribution of sets of autonomous computing objects (agents), called multi-agent systems (MAS). It is thanks to the effective application of agent-oriented technologies that the development of distributed and intelligent programs in complex and highly dynamic environments became possible.

Multi-agent systems are successfully implemented in almost all fields of activity, including production, education, marketing, health care, etc., improving people's daily life.

Getting a good education has always been an important aspect in our lives. Most citizens consider a good education a means of obtaining a decent life. The globalization of the world and the globalization of learning are expanding the application areas of artificial intelligence (AI) in education, including profiling and prediction, assessment, personalization, and intelligent learning systems. The main goal of educational institutions is to improve the quality of education, that is, to improve the process of acquiring knowledge by students.

In addition, we observe an approach to the actual problem of higher education institutions, which are forced to close due to quarantine or due to the war that Russia has unleashed in Ukraine, and switch to online education using virtual environments.

The main problem facing learning in virtual environments is not only the availability of educational content for users, but also the ability to present knowledge in the right place, at the right time, and in the right way. For this reason, researchers are paying more and more attention to the application of intelligent agents to manage the learning process in order to improve the performance of students in the online environment.

ICST-2023: Information Control Systems & Technologies, September 21-23, 2023, Odesa, Ukraine
EMAIL: nataliia.axak@nure.ua (N. Axak); mauxion@gmail.com (M. Kushnaryov); andrii.tatarnykov@nure.ua (A. Tatarnykov)
ORCID: 0000-0001-8372-8432 (N. Axak); 0000-0002-3772-3195 (M. Kushnaryov); 0000-0002-1632-8188 (A. Tatarnykov)

Therefore, the increasing integration of intelligent agents into our lives, especially where computer systems must be able to communicate both with each other and with people, requires research and development of new concepts and tools in the complex field of multi-agent systems.

2. Related works

The technology of multi-agent systems is used in almost all real-world applications, whether it is a simple e-commerce auction or an air traffic control system. It is one of the new technologies that has gained popularity very quickly due to the fact that it allows easy development of complex and distributed systems. It also appeared in the educational environment. Several multi-agent e-learning systems containing different functions have been developed and implemented. Paper [1] analyzes and compares various existing multi-agent e-learning systems based on their characteristics such as interactivity, adaptability, security, etc.

The purpose of the special issue “Advances in Multi-Agent Systems” is to promote MAS by increasing its visibility and increasing its accessibility to the scientific community [2]. This review aims to provide an overview of current research on MAS agents and technology and highlight the high level of activity in this area. The researched works demonstrate the constant growth of the scientific community's interest in new models, techniques and methods for multi-agent systems. Although research related to MAS is still developing in the most "classical" directions, at the same time it is also expanding into new areas - learning huge amounts of data.

The purpose of another special issue "New Insights in Multi-Agent Systems Cooperation, Control and Optimization" [3] is to attract new, high-quality contributions to the theory, modeling, development and application of social processes of multi-agent systems. Approaching many problems in science and technology as multi-agent systems emphasizes the need to investigate problems related to collective control, collective decision-making and optimization, collective and social machine learning, multi-agent reinforcement, as well as coordination, cooperation and evolution of agents, etc.

The use of agents in learning systems is considered in [4, 5, 6, 7, 8, 9, 10, 11].

The purpose of the work [12] is to analyze the use of software architectures in intelligent e-learning systems. In addition, the authors propose an open, distributed, agent-based software architecture for flexible, personalized learning and the development of personalized educational resources. They also discuss an ontology-based information model for personalized learning as part of a proposed agent-based software architecture. The relationships between the components of the presented software model and the information model based on the ontology, which describes the data and knowledge necessary for the effective operation of the learning system, are also considered.

The document [13] presents the architecture of the intelligent learning management system (ILMS) applied to Moodle. They presented the design and implementation of an agent that selects a learning strategy according to a student's learning style. The selected learning strategy used to filter the learning objects displayed to students.

The authors of the work [14] believe that in order to achieve the main goal of electronic learning systems (LMS) - increasing the effectiveness of learning, it is necessary to satisfy individual requirements and provide personalized practices depending on the abilities of students. This paper proposes a dynamic multi-agent system that includes five agents that takes into account the differences in capabilities of different users:

- the project clustering agent is used to cluster a set of educational resources/projects into similar groups;
- the student cluster agent groups students according to their preferences and abilities;
- a student and project matching agent is used to map each student group to an appropriate project or specific learning resources according to specific design criteria;
- the student matching agent is designed to efficiently match different students;
- dynamic student clustering agent is used for continuous tracking and analysis of student behavior in the system, such as changes in knowledge and skill levels.

The article [15] presents a multi-agent system for recommending learning objects in virtual learning environments aimed at improving the customization of learning instructions regarding learning content according to the student's profile. The structure of the system consists of four intelligent agents:

- administrator agent (responsible for student access control);
- profile agent (responsible for determining the learning style of students);

- recommendation agent (analyzes student history to recommend educational facilities);
- tutoring agent (offers help to students).

The document [16] proposes an approach to building a personalized e-learning environment, in which the main emphasis is placed on the development of students' needs. An adaptive agent-based architecture is proposed that extends the Moodle platform to support learning solutions and behavior adaptation. The paper describes the characteristics, functions, and interactions of the agents involved in each module of the adaptive architecture, as well as the intelligent agent for making instructional decisions. The purpose of this agent is to gather information generated by other agents and provide the best personalized support for end users, teachers and students, taking into account their relationship to the learning environment.

The aim of the article [17] is to present the benefits of integrating ubiquitous computing together with distributed artificial intelligence techniques to build an adaptive and personalized context-aware learning system using mobile devices. The authors proposed a multi-agent context-dependent e-learning system with the following functionalities:

- context-dependent learning planning;
- personalized course evaluation;
- selection of educational facilities according to the student's profile;
- search for educational objects in repositories;
- search for thematic teaching assistants;
- access to current context-oriented learning activities.

The system [18] is focused on three main characteristics: the learning style according to the Felder-Silverman learning style model, the level of knowledge and possible defects of the student. Three types of disabilities were taken into account, namely hearing impairment, visual impairment and dyslexia. The system will be able to provide students with a sequence of learning objects that matches their profiles for personalized learning. The main goal of this system is to recommend to students a learning path that matches their characteristics and preferences using the Q-learning algorithm.

The main goal of the study [19] is to build a multi-agent architecture that provides the ability to adapt to the student, preferences in the e-learning environment, analyze and control communication and interaction between various agents of the proposed system. To this end, they introduced a distributed intelligent whiteboard agent that provides communication between participating agents

The document [20] proposes a multi-agent architecture of an emotional-intellectual e-learning system that aims to help children with autism spectrum disorder overcome learning disabilities. The proposed architecture is based on several agents that allow to intelligently solve emotional, cognitive and pedagogical problems.

The document [21] describes an agent-oriented approach, which is aimed at creating learning situations by solving problems. The proposed system is designed as a society of agents that organizes interfaces, coordinators, information sources and mobile devices. The goal of this approach is to challenge learners to engage in multiple learning activities chosen according to their skill level and preferences to enable adaptive learning and reduce the number of learners in an electronic environment.

The aim of the study [22] is to develop an e-learning system for basic mathematics that is able to provide each student with personalized content to overcome misconceptions. The system uses a multi-agent architecture to monitor student activity while simultaneously observing and modeling student knowledge and misconceptions. Lessons and exam questions are selected dynamically by the multi-agent system to cover the prerequisites of new lessons depending on the user's profile.

In order to provide intelligent tuning and improve the adaptation quality of the learning environment, [23] proposes a multi-agent adaptive learning system architecture based on incremental hybrid case-based reasoning to make the right decision in real time under the current learning environment.

The documents [24, 25, 26, 27] presents mobile learning systems using the multi-agent paradigm. The systems provides adaptive learning content, personalized to the learner's style and preferences, to increase learner satisfaction and facilitate the learning process.

Taking into account the research works reviewed earlier, it is found that agents can provide students with personal assistants for personalized learning in order to improve the effectiveness of the learning process. However, the problems that arise during distance learning, namely the monitoring of class attendance and the detection of falsification during testing or exams, still remain unsolved.

This document is aimed at developing the architecture of an agent-based learning platform (ALP) to provide online education services, which allows students to be aware of the progress of their own educational activities and to interact with parents in the event of an incident when their child is at risk.

3. The proposed model of the agent-based learning platform to provide online education services

Learning management processes become efficient and transparent when each participant in the educational process will have access to the information intended for him. This is possible when internal management processes at the university are carried out with the help of information technologies. Educational services become understandable and accessible in electronic form, and the management of the educational institution will always have true data to make effective decisions.

For effective management of internal learning processes in an educational institution, the authors have developed a model of an agent-oriented educational platform, which can be used to provide services to university management, teachers, students, and their parents.

The proposed ALP model is a general structure that covers the relationship of all components: a multi-agent system (MAS), a university web portal, a Moodle learning management system, a cloud data storage and a developed student behavior monitoring subsystem [28]

The architecture of the proposed agent-oriented learning platform for providing online educational services is shown in Figure 1.

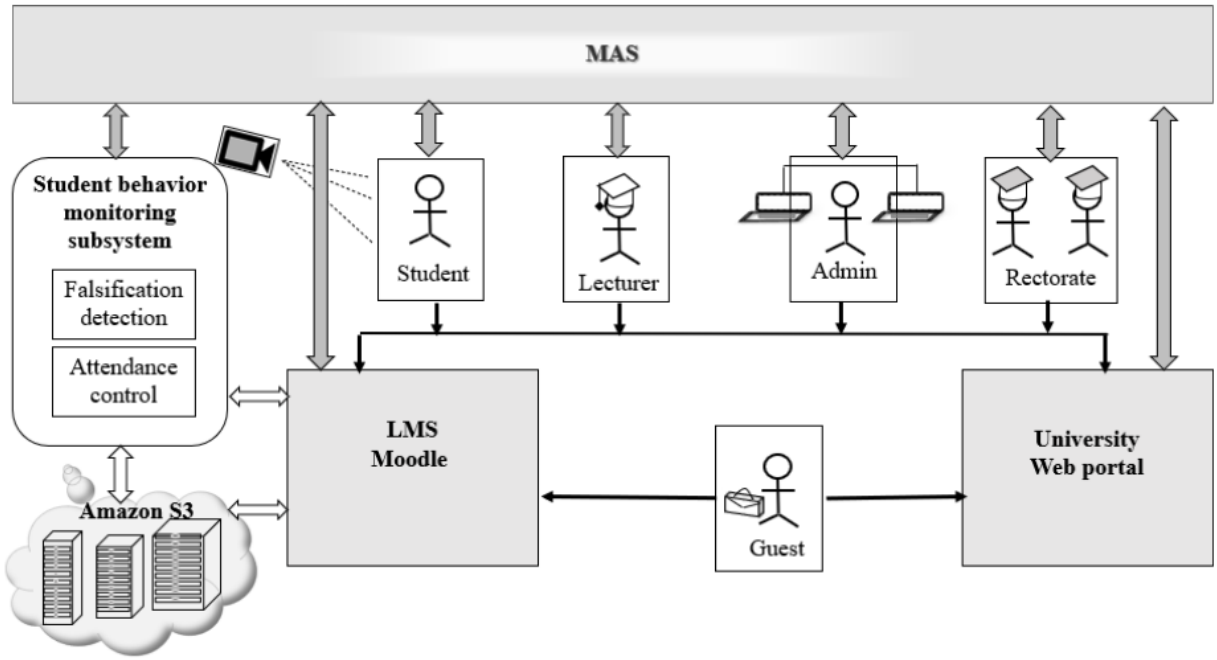

Figure 1: Architecture of the agent-oriented learning platform

In Figure 1, user access to ALP components is indicated by black arrows; the connection of ALP components for monitoring student behavior is indicated by white arrows; the messaging relationship is indicated by the gray arrows.

The multi-agent system is designed for modeling the process of communication and coordination of participants in the educational process. This approach helps to solve urgent problems of each of the participants thanks to the monitoring of information located on Moodle and on the university's Web portal.

The multi-agent system is presented as follows

$$MAS = \{Ag, A, E\}, \quad (1)$$

where $Ag = \{Ag_{\text{Student}}, Ag_{\text{Lecturer}}, Ag_{\text{Admin}}, Ag_{\text{Rectorate}}, Ag_{\text{Guest}}\}$ – a set of agents operating in the environment E; $A = \{A_{\text{Student}}, A_{\text{Lecturer}}, A_{\text{Admin}}, A_{\text{Rectorate}}, A_{\text{Guest}}\}$ – a set of agent actions ($A_i = \{a_1^i, a_2^i, \dots, a_n^i\}, i = \overline{(Student, Lecturer, Admin, Rectorate, Guest)}$); $E = \{E_{\text{Monitoring}}, E_{\text{Moodle}}, E_{\text{University}}\}$ – a set of states of the environment of Student behavior monitoring subsystem ($E_{\text{Monitoring}}$), the environment of LMS Moodle (E_{Moodle}) and the environment of University Web portal ($E_{\text{University}}$), ($E_j = \{e_1^j, e_2^j, \dots, e_m^j\}, j = \overline{(Monitoring, Moodle, University)}$).

The interaction of agents and the environment is the execution of actions:

$$f_j: e_0^j \xrightarrow{a_0^i} e_1^j \xrightarrow{a_1^i} e_2^j \xrightarrow{a_2^i} \dots \xrightarrow{a_{u-1}^i} e_u^j \xrightarrow{a_u^i} \dots \quad (2)$$

The behavior of the environment is modeled as a function of the state converter

$$\Upsilon_j: f_j\left(ended\ with\ a_k^i\right) \to \rho(E),\ (k = \overline{1,n}), \quad (3)$$

This function represents the set of environmental states $\Upsilon_j(f_j)$ that can occur as a result of action a_n^i in state e_m^j.

Due to the fact that learning resources are decentralized, the use of only one stationary agent will not be sufficient. Therefore, mobile agents are needed to receive information from various resources (LMS Moodle, University Web portal, Amazon S3). A society of agents was created to develop a multi-agent system (Fig. 2).

Each agent is assigned a specific role. MAS functionality is available to agents with the following roles:

- "Student" - students, graduate students and doctoral students;
- "Lecturer" - teachers and researchers;
- "Admin" - site administrator;
- "Rectorate" - university administration (rector, vice-rectors, dean, etc.);
- "Guest" only if the role of the guest is performed by the student's parents.

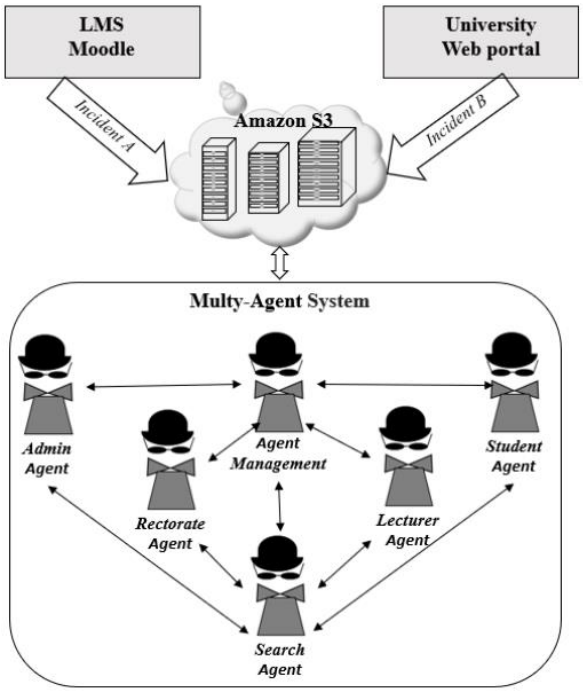

Figure 2: The main actors of the multi-agent system

Student behavior monitoring subsystem allows:

- the teacher to make a decision regarding the assessment of the student's knowledge in case of detection of falsifications during testing or exams;
- the teacher to increase the student's activity by adjusting the total score for attending classes;
- determine the duration of students' attendance at classes in accordance with the established schedule in the educational institution for parental control.

All users of the proposed ALP - employees, admin, students, postgraduates, applicants, guests, etc., can have access to various components of the ALP depending on the role performed and receive information and all kinds of help regarding the educational process at the institution of higher education.

For example, "Student" can:

- have access to Moodle blocks related to store, communicate and collaborate;
- automatically receive up-to-date information from the university's Web portal, from the dean's office and teachers;
- get a certificate that he is a student of this institution (with enrollment order number, term of study), individual study plan of the applicant (acquired credits and grades);

- participate in surveys (internal and external);
- learn about scientific conferences, internships, scientific circles, Olympiads, hackathons, etc.;
- to be invited to participate in scientific works with publications indexed in various scientometric databases;
- to keep certificates, diplomas, awards for academic success.

"Lecturer" can:

- have access to Moodle blocks related to store, communicate, evaluate and collaborate;
- to receive information about the duration of students' attendance at classes and notification of detection of falsifications during testing or exams;
- automatically receive up-to-date information from the university's Web portal, from the dean's office and teachers;
- a certificate of employment;
- to be invited to participate in scientific works with publications indexed in various scientometric databases;
- information about scientific conferences, internships, etc.

“Admin” can:

- performs the necessary actions to support the operation of the Moodle platform;
- receives up-to-date information for the effective operation of the Moodle platform and the university's Web portal;
- is responsible for maintaining the functionality of the university's Web portal and ensuring network security, manages the placement, updating, and moderation of content.

“Rectorate” can:

- have access to Moodle blocks;
- provide information for the university's Web portal.

“Guest” can:

- have access to open information on the university's Web portal;
- receive information about the implementation of the student's individual study plan, if the role of "guest" is performed by parents.

Agents are represented as functions

$$Ag: f_{\text{j}}\left(ended\ with\ e_u^j\right) \to A,\ \ (u = \overline{1,m}). \tag{4}$$

The behavior of an agent in an environment is the set of all runs

$$F(Ag, Env), \qquad (Env = \{E, e_0^{\text{j}}, \Upsilon_j\}). \tag{5}$$

Moodle LMS is a learning platform used to provide teachers and students with a personalized learning environment. This software is deployed on its own web server. Moodle is free and open source software under the GNU General Public License. Anyone can adapt, extend or modify Moodle for both commercial and non-commercial projects without any licensing fees.

There are many tasks in the educational process. At the initial stage of the development of an agent-oriented educational platform, the authors limited themselves to the implementation of only one task - the assessment of the knowledge acquired by the student. This functionality provides important information related to a student's progress and allows you to individually highlight students' weaknesses or strengths and inform parents of their child's academic progress. In order to achieve this goal, there is attendance monitoring and ongoing control, which is carried out during the semester during classroom classes (lectures, laboratory and practical classes, etc.) To do this, according to the established schedule of classes, "Student Agent" detects negative aspects:

- absence of a student in classes;
- sum of points obtained according to the results of the final/semester test of success in accordance with the schedule of the educational process is less than the minimum permissible.

In case of negative aspects, "Student Agent" sends a message to parents that their child is in the risk zone.

Figure 3 shows a usage diagram showing the main functionality of the agents.

Initialization occurs as the first event handled by the agent, and deinitialization occurs after the application terminates. Deinitialization and initialization are performed only once.

Receiving messages from other agents is one of the main functions of an agent. An agent can receive messages from other agents or information from the network. Messages are sent to inform other agents or to give an indication of what action to take.

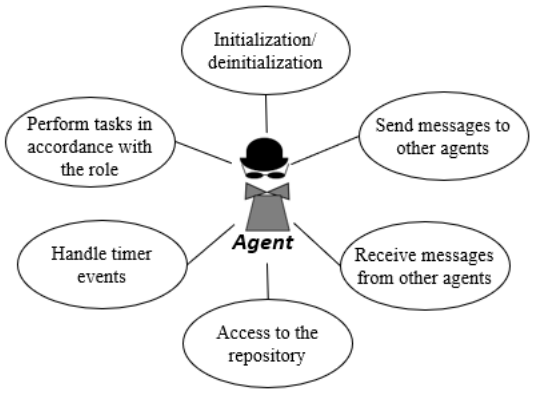

Figure 3: Agent program usage chart

Receiving messages from other agents is one of the main functions of an agent. An agent can receive messages from other agents or information from the network. Messages are sent to inform other agents or to give an indication of what action to take.

Timer event handling is used to periodically check any resource on the network or to check the status of any task performed by another agent.

Each agent is equipped with a knowledge base consisting of scenarios to follow in order to achieve a given goal.

A multi-agent system is based on several agents that are designed to communicate and exchange data in order to provide users with relevant information, for example, to notify a student and his parents that he is at risk due to poor academic performance. Within the framework of the proposed model, the agent receives messages from other agents and external notifications about the activity of the learning process from the environment.

Agent Management provides agent management, allows you to create and delete agents, publish information about the services provided and find agents providing the necessary services and enter into negotiations with them.

The diagram of the interaction performed by each agent from the moment of launch to the end of its existence in the environment is presented in fig. 4.

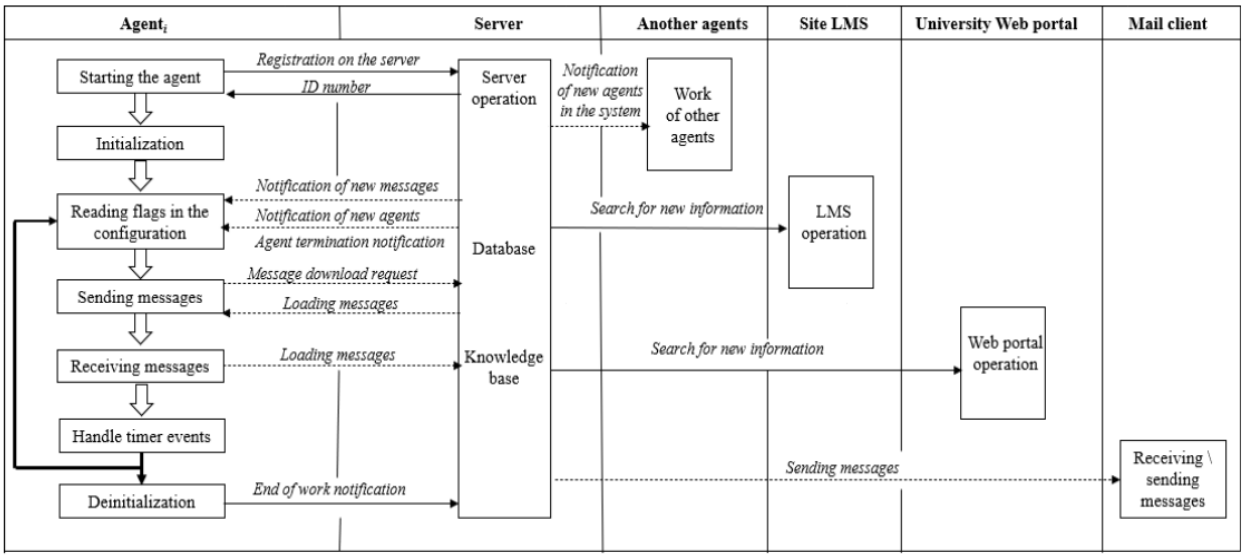

Figure 4: The agent activity (cooperation) diagram

Actions performed by participants, who are involved in ALP, are marked with rectangles. The sequence of actions performed by agents is indicated by vertical arrows, and messages exchanged by agents are shown by horizontal arrows. A solid arrow indicates that this message is sent in any case, and a dashed one means that sending the message depends on some condition.

Viewing the chart starts with the agent running. At this point, a request to register on the server is broadcast. The server, upon receiving such a request, adds information about this agent - the current location and its type (Agent Management, Admin Agent, Rectorate Agent, Lecturer Agent, Search Agent

and Student Agent) to its database. Based on this data, it generates a unique identification code for each agent. This code is used when sending messages between agents and the server. After registering on the server, the agent executes a script specific to it.

Messages are received and sent if the corresponding script is executed for this agent. The agent sends a request to receive data, the server sends it to the agent, and then the script is launched (Fig. 5). The downloaded message is stored in a file on the drive, and its handle is passed to the script.

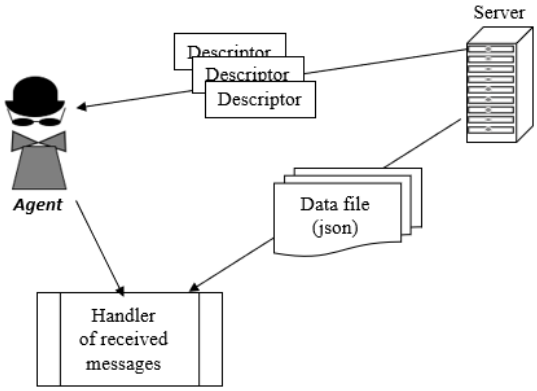

Figure 5: Messaging mechanism

When sending a message, the corresponding script is launched, which is designed to process the data that will be sent. Then a descriptor is formed, which specifies the size of the message, the way to place the message on the disk, the identifiers of the agents that should receive it, and a checksum to verify the integrity of the message. This handle is sent to the server, which downloads this message at the specified address. It then parses the handle and sends a new message notification to all agents specified in the handle.

Thus, the developed system gives ALP users the opportunity to maintain updated information that helps to improve learning outcomes.

4. Modeling the agent's function of detecting negative incidents in the educational process

In this paper, the authors emphasized the implementation of the learning progress control function. For this purpose, the agent, in case of detecting negative aspects (student's absence from classes is more than 20%, or the student received the sum of points based on the results of the final/semester academic performance check in accordance with the schedule of the educational process is less than the minimum allowable), sends a message that this student is at risk.

An agent is created by adding a corresponding component to the program workspace. Creating an agent includes two actions: programming the agent and linking the created agent to other agents. Linking occurs by connecting two agents in the workspace. Communication between agents can be uni- or bi-directional. In the first case, messages can be transmitted only in one direction and in the other - agents have the ability to exchange messages in both directions

Programming agents and organizing their interaction is done with the help of basic HTML, CSS, JavaScript, PHP, and MySQL capabilities.

The monitoring subsystem uses a web camera to track the presence of students in class [28]. Based on the monitoring results, the detection time (in minutes) of the student's presence during the entire class is recorded. This data is stored on the server. According to the set timer parameters (Fig. 4), Student Agent checks the attendance data stored on the server and builds a table (Fig. 6).

The data is stored in a table that consists of the following elements:

- unique_id - a unique record identifier that is generated from the time the record is created and encoded in md5 format;
- data - information about student attendance, which is stored in json format;
- create_at - date of creation of the attendance record in the usual date format.

The data in the table is updated every week of the training plan. Each record contains information about the attendance of one week.

unique_id	data	created_at
827ccb0eea8a706c4c34a16891f84e7b	{ "status": "OK", "result": { "827...	2023-02-04
01cfcd4f6b8770febfb40cb906715822	{ "status": "OK", "result": { "...	2023-02-11
dcdcce4af533fdf8581777e1c954a072	{ "status": "OK", "result": { "...	2023-02-18
715a097049577be8cf37147e127de682	{ "status": "OK", "result": { "...	2023-02-25

Figure 6: Table of student attendance at classes

The results of the analysis of students' success from the course at the time of the inspection are shown in Table 1 (for the period 2023-03-13 – 2023-03-13) and the results of the general analysis of students' success according to the educational plan at the time of the inspection are shown in Table 2, where Th_L – the total hours of lectures; H_L – the hours of the student's attendance at lectures, Th_{lp} – the total hours of laboratory/ practical works; H_{lp} – the hours of the student's attendance at laboratory/ practical works; P_{Min} – the minimum allowable points per semester for the course; S_p – the student's accumulation of points for the course; N_u – the number of uncompleted laboratory/practical works.

Table 1
The results of the analysis of students' success from the course

Username	Group	Th_{lp} / H_{lp}	P_{Min} / S_p
Albert Joref	KIT-22-1	2/2	20/25
Kirill Petrtenko	KIT-22-1	2/1,8	20/20
Pavlo Sydorenko	KIT-22-1	2/1,5	20/0
Denis Vashenko	KIT-22-1	2/0	20/0

Analysis is performed weekly.

Table 2
The results of the general analysis of students' success according to the educational plan

Username	Group	Th_L / H_L	Th_{lp} / H_{lp}	N_u
Albert Joref	KIT-22-1	50/48	28/26	0
Kirill Petrtenko	KIT-22-1	50/41,8	28/24	1
Pavlo Sydorenko	KIT-22-1	50/36,5	28/20	2
Denis Vashenko	KIT-22-1	50/1,5	28/0	7

Analyzing the diagram in Figure 7 helps to identify the links between different indicators and draw conclusions about student performance and student achievement. For example, the relationship between attendance and academic performance: if students who attend more lectures and labs also have fewer

incomplete assignments, we can conclude that regular attendance can affect a student's academic performance.

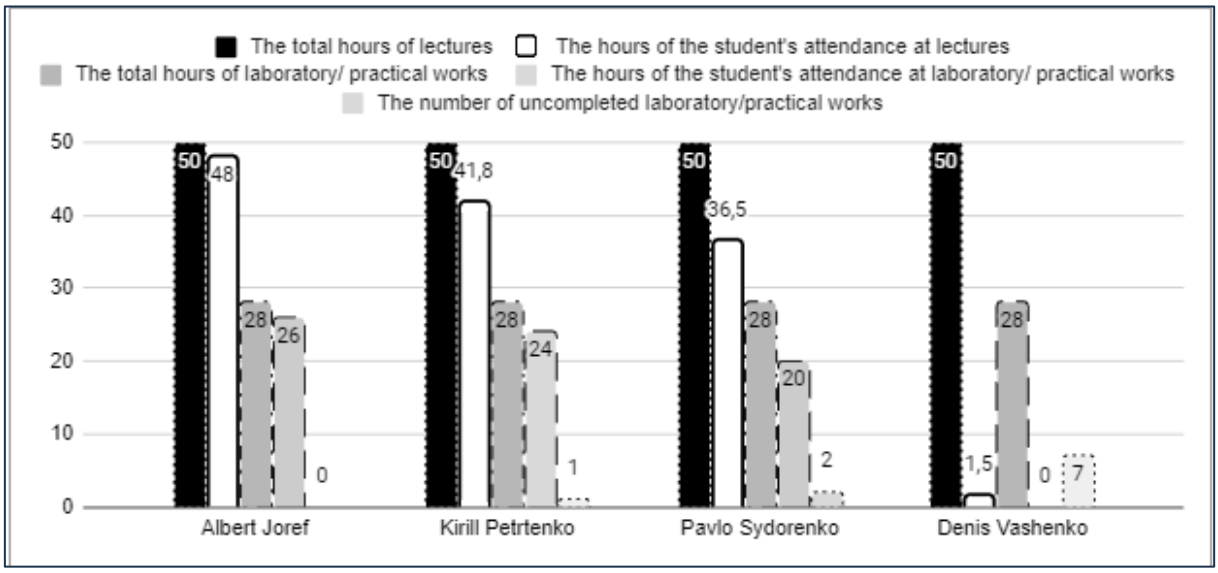

Figure 7: The diagram of student success for the semester

If you click on a row, you can see more detailed information in a separate window (Fig. 8):

- "status" - is an automatic check of the json file for correctness of formation;
- "result" - information about the created record: unique key; date of creation; period for which the data was collected; identifier of the study group; information about each student of this group (their attendance at the disciplines that are mandatory in the curriculum).


```
"status": "OK",
"result": {
    "827ccb0eea8a706c4c34a16891f84e7b": {
        "id": "827ccb0eea8a706c4c34a16891f84e7b",
        "date_generate": "2023-03-19 21:00:01",
        "info_from": "2023-03-13 to 2023-03-19",
        "group": "CITm-22-1",

        "students_info": {
            "class_attendance": {
                "Tatarnykov Andrii": {
                    "visits_minutes": "357/360",
                    "subj_best_attendance": "Multi-agent systems",
                    "minutes_best_attendance": "175/180",
                    "missed_classes": "0",
                    "subj_worst_attendance": "Intelligent data analysis",
                    "minutes_worst_attendance": "115/120"
                },
                "Petrenko Kirill": {
                    "visits_minutes": "201/360",
                    "subj_best_attendance": "Multi-agent systems",
                    "minutes_best_attendance": "110/120",
                    "missed_classes": "1",
                    "subj_worst_attendance": "Features of modern scientific communication",
                    "minutes_worst_attendance": "0/60"
                },
                "Sydorenko Pavlo": {
                    "visits_minutes": "140/360",
                    "subj_best_attendance": "Multi-agent systems",
                    "minutes_best_attendance": "90/120",
                    "missed_classes": "3",
                    "subj_worst_attendance": "Intelligent data analysis",
                    "minutes_worst_attendance": "0/120"
```


Figure 8: Information on student attendance in JSON data exchange format

The json (JavaScript Object Notation) format is independent of the implementation language, so it is convenient to use it for data exchange.

Depending on the debugging parameters of the developed system, the agent sends a message to e-mail (Fig. 9, 10).

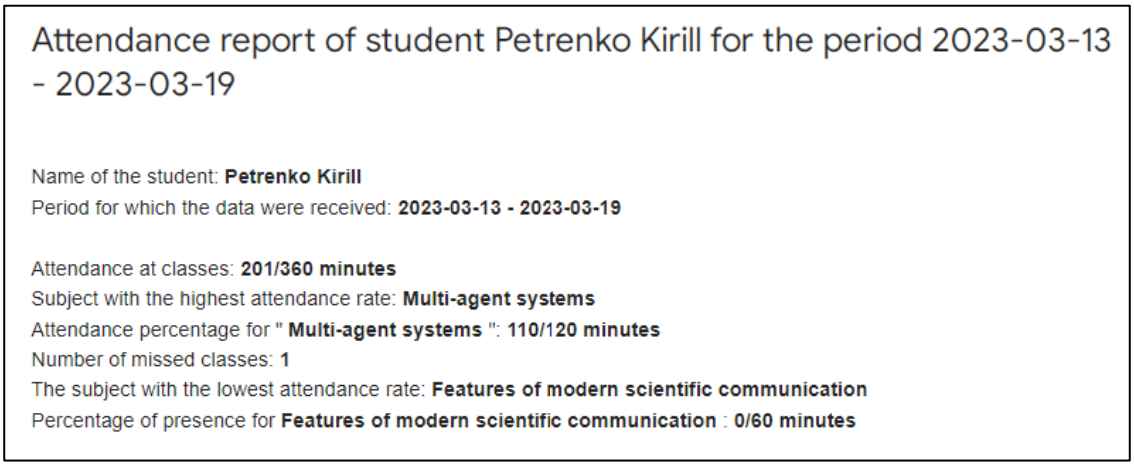

Figure 9: Mail notification of successful training

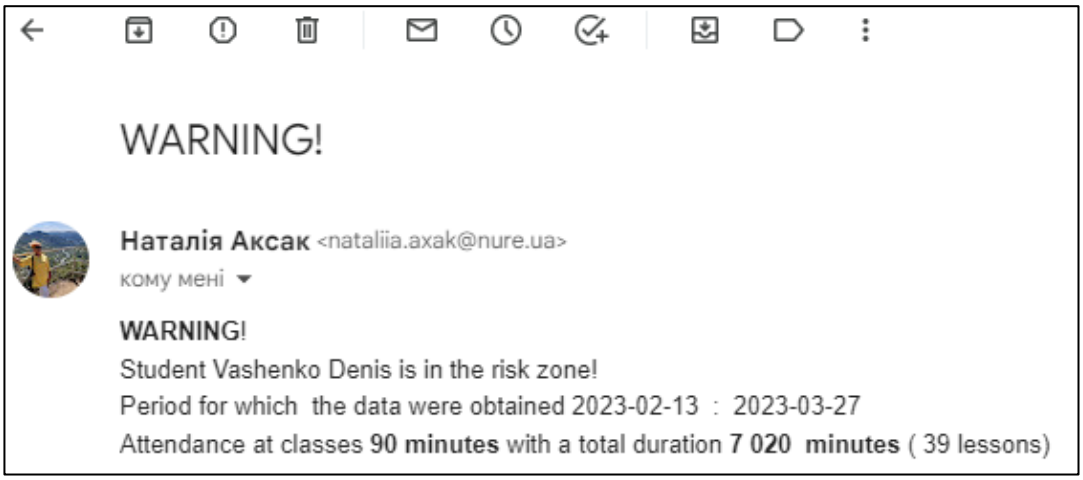

Figure 10: Mail notification in case of a negative aspect

Depending on the settings, the notification can be sent weekly in any case (Fig. 8), or the notification will be sent only if negative aspects occur. Figure 9 shows an example when a student Vashenko Denis attended classes for only 90 minutes from the beginning of the semester 2023-02-13 to the current date 2023-03-27, although during this period 39 lessons were held for a total of 7020 minutes. Therefore, this student is at risk of being expelled.

5. Discussion

The approach proposed on the basis of the agent-based learning platform model allows solving the complex problem of timely provision of knowledge together with monitoring of student performance by both teachers and parents. This helps to increase the effectiveness of distance learning. The chosen learning strategy allows us to detect the lack of student progress in a timely manner by monitoring the information available on Moodle and the university's web portal. In the event of negative aspects (prolonged absence of a student from classes, the amount of points received based on the results of the final/semester academic performance check in accordance with the schedule of the educational process is less than the minimum allowable), the system notifies the student, his/her parents and the teacher that such a student is at risk. Thus, the student has the opportunity to control his or her academic performance

and, if necessary, to liquidate his or her debts in a timely manner, i.e., to successfully complete the course and not be expelled.

6. Conclusion

The paper solves an urgent problem that arises during distance learning in an e-learning environment, namely, analyzing and monitoring student performance.

To effectively manage the internal learning processes, we propose a model of an agent-based learning platform that can provide services to university management, teachers, students, and their parents. The novelty of the proposed model lies in taking into account the interconnection of all its components: a multi-agent system, a university web portal, a Moodle learning management system, a cloud data storage, and a subsystem for monitoring student behavior. The multi-agent system is designed to model the process of communication and coordination of participants in the educational process. The role of each actor of the multiagent system is defined.

The use of the JSON data exchange format made it possible to realize communication between agents.

Modeling the identification of negative aspects in the learning process with the help of agents confirmed the feasibility of the proposed approach.

7. References

[1] M. U. Bokhari, S. Ahmad, Multi-agent Based E-learning Systems: A Comparative Study, in: Proceedings of the 2014 International Conference on Information and Communication Technology for Competitive Strategies, 2014, pp. 1-6. doi.org/10.1145/2677855.2677875.

[2] S. Mariani, A. Omicini, Special Issue “Advances in Multi-Agent Systems”: Editorial. Appl. Sci., 13 2023, 3027. doi.org/10.3390/app13053027.

[3] J. Serrano, D. Jimenez, L.M. Contreras, F. Alvarez, A Deep Reinforcement Learning Quality Optimization Framework for Multimedia Streaming over 5G Networks, Appl. Sci, 12 (2022) 10343. https://doi.org/10.3390/app122010343.

[4] H. A. Abu-Alsaad, Agent Applications in E-learning Systems and Current Development and Challenges of Adaptive e-Learning Systems, in: 2019 11th International Conference on Electronics, Computers and Artificial Intelligence (ECAI), 2019, pp. 1-6. doi: 10.1109/ECAI46879.2019.9042015.

[5] I. Vuković, K. Kuk, P. Čisar, M. Banđur, Đ. Banđur, N. Milić, B. Popović, Multi-Agent System Observer: Intelligent Support for Engaged E-Learningб Electronics, 10 12 (2021) 1370. doi.org/10.3390/electronics10121370.

[6] F. Elghibarı, R. Elouahbi, F. El Khoukhi, Dynamic Multi Agent System for Revising E-learning Content Materia, The Turkish Online Journal of Distance Education (TOJDE), 20 1 (2019) 131-144. doi.org/10.17718/tojde.522434.

[7] N. Viswanathan, S. Meacham, F. F. Adedoyin, Enhancement of Online Education System by Using a Multi-agent Approach. Computers and Education: Artificial Intelligence, 3 (2022) 100057. doi.org/10.1016/j.caeai.2022.100057.

[8] M. Amane, K. Aissaoui, M. Berrada, A Multi-agent and Content-Based Course Recommender System for University E-learning Platforms, in: Motahhir, S., Bossoufi, B. (eds) Digital Technologies and Applications. ICDTA 2021. Lecture Notes in Networks and Systems, Springer, Cham, vol 211, 2021, pp. 663-672. doi.org/10.1007/978-3-030-73882-2_60.

[9] A. B. Mourão, J. F. M. Netto, SIMROAA Multi-Agent Recommendation System for Recommending Accessible Learning Objects, in: 2019 IEEE Frontiers in Education Conference (FIE), Covington, KY, USA, 2019, pp. 1-9. doi: 10.1109/FIE43999.2019.9028504.

[10] S. Munawar, S. Khalil Toor, M. Aslam, E. Aimeur, PACA-ITS: A Multi-Agent System for Intelligent Virtual Laboratory Courses. Applied Sciences, 9 23 (2019) 5084. doi.org/10.3390/app9235084.

[11] Z. Houhamdi, B. Athamena, R. Abuzaineddin, M. Muhairat, A Multi-Agent System for Course Timetable Generation, TEM Journal, 8 1 (2019) 211-221.

[12] T. Ivanova, V. Terzieva, K. Todorova, An Agent-Oriented Architecture for Strategy-Based Personalized E-Learning, in: 2021 Big Data, Knowledge and Control Systems Engineering (BdKCSE), Sofia, Bulgaria, 2021, pp. 1-8, doi: 10.1109/BdKCSE53180.2021.9627260.

[13] M. L. Morales-Rodríguez, J. A. Ramírez-Saldivar, J. P. Sánchez-Solís, A. Hernández-Ramírez, Design of an Intelligent Agent for Personalization of Moodle's Contents, Res. Comput. Sci., 56 (2012) 11-17.
[14] M. M. Al-Tarabily, R. F. Abdel-Kader, G. A. Azeem, M. I. Marie, Optimizing Dynamic Multi-Agent Performance in E-Learning Environment, in: IEEE Access, vol. 6, 2018, pp. 35631-35645. doi: 10.1109/ACCESS.2018.2847334.
[15] T. O. Almeida, J. F. De Magalhães Netto, A. M. M. Lopes, Multi-Agent System for Recommending Learning Objects in E-Learning Environments, in 2021 IEEE Frontiers in Education Conference (FIE), Lincoln, NE, USA, 2021, pp. 1-5, doi: 10.1109/FIE49875.2021.9637258.
[16] A. Alexandru, E. Tirziu, E. Tudora, O. Bica, Enhanced Education by Using Intelligent Agents in Multi-Agent Adaptive e-Learning Systems, Studies in Informatics and Control, 24 1 (2015) 13-22.
[17] O. M. Salazar, D. A. Ovalle, N. D. Duque, Adaptive and Personalized Educational Ubiquitous Multi-Agent System Using Context-Awareness Services and Mobile Devices, in: Zaphiris, P., Ioannou, A. (eds) Learning and Collaboration Technologies. LCT 2015. Lecture Notes in Computer Science, Springer, Cham, vol 9192, 2015, pp. 301-312. doi.org/10.1007/978-3-319-20609-7_29.
[18] H. El Fazazi, M. Elgarej, M. Qbadou, K. Mansouri, Design of an Adaptive E-Learning System based on Multi-Agent Approach and Reinforcement Learning, 11 1 (2021) 6637–6644. doi.org/10.48084/etasr.3905.
[19] S. Hammami, H. Mathkour, Adaptive E-learning System Based on Agents and Object Petri Nets (AELS-A/OPN), Computer Applications in Engineering Education, 23 2 (2015) 170-190. doi 10.1002/cae.21587.
[20] N. Tahiri, M. El Alami, An Intelligent E-Learning System for Autistic Children: Multi-Agent Architecture, in: Ezziyyani, M. (eds) Advanced Intelligent Systems for Sustainable Development (AI2SD'2019). AI2SD 2019. Advances in Intelligent Systems and Computing, Springer, Cham, vol 1102, 2020, pp. 83-90. doi.org/10.1007/978-3-030-36653-7_8.
[21] F. Z. Lhafra, O. Abdoun, Integration of Evolutionary Algorithm in an Agent-oriented Approach for an Adaptive E-learning, International Journal of Electrical and Computer Engineering, 13 2 (2023) 1964-1978. doi:10.11591/ijece.v13i2.pp1964-1978.
[22] M. Amer, H. Aldesoky, Building an E-Learning Application Using Multi-agents and Fuzzy Rules. Electronic Journal of e-Learning, 19 3 (2021) 199-208. doi.org/10.34190/ejel.19.3.2308.
[23] N. E. Ghouch, M. Kouissi, E. M. En-naimi, Multi-agent Adaptive Learning System Based on Incremental Hybrid Case-based Reasoning (IHCBR), in Proceedings of the 4th International Conference on Smart City Applications, 2019, pp. 1-6. doi.org/10.1145/3368756.3369030.
[24] J. H. Ko, C. Hur, H. Kim, A Personalized Mobile Learning System Using Multi-agent, in: Dean, M., et al. Web Information Systems Engineering – WISE 2005 Workshops. WISE 2005. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, vol 3807, 2005, pp. 144-151. doi.org/10.1007/11581116_15.
[25] D. G. De La Iglesia, J. F. Calderón, D. Weyns, M. Milrad, M. Nussbaum, A Self-Adaptive Multi-Agent System Approach for Collaborative Mobile Learning, in IEEE Transactions on Learning Technologies, vol. 8 (2), 2015, pp. 158-172. doi: 10.1109/TLT.2014.2367493.
[26] C. N. Udanor, O. U. Oparaku, A Performance Evaluation of a Multi-Agent Mobile Learning System, International Journal of Interactive Mobile Technologies, 10 2 (2016) 10-18. doi.org/10.3991/ijim.v10i2.4873.
[27] K. Chouchane, O. Kazar, A New Cloud-Based Adaptive Mobile Learning System Using Multi-Agent Paradigm. International Journal of Organizational and Collective Intelligence (IJOCI), 12 3 (2022) 1-9. doi: 10.4018/IJOCI.306694.
[28] N. Axak, A. Tatarnykov, The Behavior Model of the Computer User, in: IEEE 17th International Conference on Computer Sciences and Information Technologies (CSIT), 2022, pp. 458-461. doi: 10.1109/CSIT56902.2022.10000499.