

¹Aidos Mukhatayev, ²Serik Omirbayev, ¹Andrii Biloshchytskyi,
¹Aidar Ispussinov, ³Yurii Andrashko, ¹Yuliya Idiyatova, ⁴Indira Abilmazhinova*

¹ Astana IT University, Astana, Kazakhstan

² QAIRU

³ Uzhhorod National University, Uzhhorod, Ukraine

⁴ Nazarbayev University, Astana, Kazakhstan

INTEGRATION OF THE INFORMATION SYSTEM “QUALITY MANAGEMENT OF HIGHER AND POSTGRADUATE EDUCATION” WITH RELATED SYSTEMS

Abstract. This article focuses on developing and testing a model to integrate the information system “Quality Management of Higher and Postgraduate Education” with related university and government digital systems. The purpose of the study is to develop an architectural and functional model for intersystem interaction to ensure automated data exchange between internal university subsystems and the Unified Higher Education Platform. Pilot implementation confirmed the integration solution’s ability to transmit data on academic staff, students, educational programs, international activities, and research publications. Common errors in authentication, request formation, and data validation were identified, along with areas for improvement in interface design, navigation, and processing speed for large data sets. The findings show that integrating the quality management information system with interconnected digital platforms enhances the reliability, coherence, and managerial value of educational data. The practical significance of the study lies in the potential application of the proposed model as a foundation for developing a unified digital system for quality management in higher and postgraduate education.

Keywords: information system; education quality management; higher and postgraduate education; data integration; API; digital ecosystem.

Introduction

With the rise of computer technologies, specifically automated systems in education, the effort required for data collection, preparation, and results processing has greatly decreased, enabling the creation of reliable data in a structured, timely manner.

Data reliability is significantly improved through integration with university learning management systems (LMS) and related government information systems for higher education administration.

University LMS platforms support data entry and management regarding students, staff, educational program content, teaching and research infrastructure, and overall institutional resources. Particularly, student information systems store student data, organize learners by educational programs, and align them with relevant curricula. Academic performance management systems automate monitoring and evaluating student achievement based on rating-based assessment frameworks. Additionally, testing systems help assess students enrolled in universities across various disciplines included in the curriculum.

Recent studies see the integration of information systems in education not just as a technical task of data exchange between separate platforms, but as a vital part of the broader digital transformation of educational institutions. Within this perspective, integration includes connecting learning management systems (LMS), student information systems (SIS), analytical modules, learning assessment services, and administrative platforms. The literature shows that the digital transformation of universities progresses by a gradual merging of previously separate solutions; however, the sustainability of such transformation relies not only on technological infrastructure but also on process maturity, organizational coordination, and data quality (Fernández et al., 2023; Mijač et al., 2024). At the same time, a major limitation in existing research is that, while digitalization is discussed broadly, less attention is paid to the architectural mechanisms of integration, including APIs, interoperability, reference data synchronization, and master data management.

This article will attempt to address this gap by focusing on the integration of the information system “Quality Management of Higher and Postgraduate Education” with internal university and external governmental platforms. The main research question guiding the study is: How can API-based integration of

quality management information systems enhance data reliability and support managerial decision-making in higher education?

The contribution of this study is threefold. First, it proposes an architectural and functional model for intersystem interaction based on API-based data exchange. Second, the findings present a pilot implementation in real institutional contexts, specifically at two universities. Third, it identifies key challenges in authentication, data validation, and interface design, emphasizing the importance of data management and coordination.

Literature review

A significant body of research has examined the integration of student information systems (SIS) with related university services. Many studies consider SIS the foundational infrastructure of digital university governance, supporting the management of student groups, academic performance, faculty workload, and administrative tasks. Empirical studies (Çelik & Ayaz, 2022; Daim et al., 2024; Mazadu et al., 2022) show that the success of SIS implementation depends not only on system functionality but also on information quality, usability, service quality, and levels of organizational support. An important counterpoint is that even a well-designed system does not automatically generate integration impacts: without process alignment and high-quality input data, informational interconnectedness remains superficial rather than meaningfully useful.

A specific area of research examines learning management systems (LMS) as a vital part of the digital educational infrastructure. Recent studies show that LMS are increasingly seen not just as standalone e-learning platforms but as inseparable components of a larger integrated system that includes analytics, assessment, feedback tools, and administrative services. Research on the main factors influencing LMS adoption in universities highlights the significance of institutional support, technological infrastructure, and user preparedness (Alduraywish et al., 2022). Bibliometric analyses of LMS-related publications in developing countries indicate a growing interest in scalability and integrating LMS into the overall framework of digital education; however, the architectural aspects of integration remain less developed than questions of implementation and usage (Pham et al., 2022). In addition, Sayaf (2023) demonstrates that the adoption of e-learning systems in higher education relies on a combination of success factors in information systems and constructivist learning conditions, effectively linking platform integration with the quality of pedagogical design.

Contemporary literature mainly examines the integration of information systems through the lens of learning analytics. In this perspective, integration is understood as the consolidation of students' digital traces, LMS data, assessment results, course activity, and feedback into a unified analytical environment. Systematic reviews show that learning analytics is increasingly shifting from simple reporting toward decision-support ecosystems aimed at managing student engagement, self-regulation, and academic success (Bergdahl et al., 2024; Palanci et al., 2024). At the same time, studies emphasize that analytical tools yield meaningful impact only when embedded within pedagogical and organizational contexts, rather than functioning as standalone dashboards disconnected from the learning process (Drugova et al., 2024; Kleimola & Leppisaari, 2022). Thus, learning analytics can be regarded as one of the most developed mechanisms for integrating educational data, but not as a comprehensive substitute for integration itself.

Within this strand, research on learning analytics dashboards has been developing actively. A systematic review by Paulsen and Lindsay (2024) shows that contemporary dashboards are mainly focused not only on data visualization but also on supporting learning as a process. This is further supported by studies on gamified, collaborative, and AI-powered dashboards, in which the focus shifts from merely displaying indicators to interpretation, reflection, and decision support (Alam et al., 2023; Echeverria et al., 2025; Cabral et al., 2025).

At the same time, an important limitation can be identified: a substantial share of the literature focuses on the effects of already implemented solutions, while providing less detailed analysis of how these dashboards are integrated with SIS, LMS, and institutional data repositories. In other words, dashboards are often examined as pedagogical interfaces rather than as components of the university's broader enterprise information system architecture.

The findings of several studies indicate that analytical integration plays an essential role in tracking

student engagement and predicting academic success. For example, Fan et al. (2021) identify key factors influencing student engagement in the context of learning analytics in online and blended learning settings. Flanagan et al. (2022) demonstrate how behavioral data analysis can provide early warnings of risks related to declining academic performance. Kleimola et al. (2025) highlight the link between learning analytics and the development of self-regulated learning, while Candra and Jeselin (2022) expand on this research by showing that students' perceptions of digital platforms depend not just on the technical quality of the applications but also on how well they are integrated into the actual learning process. These findings are important for the study of information system integration, as they suggest that the value of data exchange between different systems lies not solely in technical connectivity but in the system's ability to support pedagogically and managerially meaningful decision-making.

Finally, broader reviews of educational technologies confirm that integrating digital tools can positively impact learning outcomes; however, the mere technological intensification of the educational environment does not, in itself, guarantee improvement. Valverde-Berrocoso et al. (2022) conclude that the effects of educational technology on student outcomes are heterogeneous and strongly context-dependent. This has important implications for research on information system integration: the central issue is not the number of platforms deployed, but the degree to which they align, interoperate, and integrate into the university's managerial cycle. The literature in this area remains unevenly developed. While topics such as adoption, user satisfaction, dashboards, and specific learning analytics interventions are relatively well explored, considerably less attention has been paid to architectural and functional models of integration that bring together LMS, SIS, quality assurance systems, and external governmental platforms into a unified digital ecosystem of education. It is precisely this gap that underscores the need for further research in this field (Mohamed et al., 2023; Fernández et al., 2023; Mijač et al., 2024).

Methods

Research Design

The study was conducted as part of applied, project-based research, including elements of design science, architectural modeling, and pilot empirical testing. This approach was chosen because the study's goal was not to describe individual digital practices but to develop, implement, and evaluate a model for integrating the information system "Quality Management of Higher and Postgraduate Education" with related university and government information systems.

The methodological framework of the study consisted of four interconnected stages. In the first stage, the problem domain was outlined, including key objects of integration and the structure of information flows. The second stage involved developing an architectural and functional model of how systems interact, focused on consolidating data from university subsystems and facilitating their transfer to the external digital platform. At the third stage, an API-based data exchange system was set up, including authentication procedures, request handling, data validation, and data updating protocols. The fourth stage involved piloting the system in real institutional settings, followed by a technical and functional assessment.

Therefore, regarding the research design, the study can be described as a pilot project centered on creating and testing an integration prototype, in which the theoretical problem was advanced to a working solution and evaluated in real institutional conditions. This increases the study's practical significance but also introduces certain limitations: the results primarily demonstrate the model's functionality and institutional relevance, rather than statistically generalizable effects for a large user base.

A key feature of the study was that the developed system was designed not as a standalone university solution but as part of a larger digital ecosystem. In this context, the pilot implementation focused on ensuring compatibility with the Unified Higher Education Platform, which serves as an external layer for national data integration. This suggests that the architectural choices in the study were intentionally made with cross-level data sharing in mind – between internal university systems and the external national higher education platform.

The focus of the study was the information system "Quality Management of Higher and Postgraduate Education," designed as an integrated core to unify fragmented data on students, academic staff, educational programs, learning outcomes, and institutional infrastructure. The system's architecture was designed to support a unified digital educational ecosystem, ensuring organized interaction between internal and external

data sources.

The technological foundation of inter-system exchange was built on an API-driven approach, allowing the automated transfer of structured data among the quality management information system, university subsystems, and the Unified Higher Education Platform. The integration process involved several consecutive steps: authenticating with access keys; creating requests with predefined parameters; sending requests to the server; server-side processing involving database interactions; generating responses; and handling results on the client side.

The pilot implementation was carried out in real operational conditions within the information systems of two universities: Astana IT University and the University of International Business. During the testing phase, both the system's technical functionality and its practical effectiveness in managing educational quality were assessed. The participating institutions used the API to send data on academic staff, students, international activities, educational programs, and research publication output. This allowed the system to be tested across various data categories that differ in structure, completeness, and update schedules.

To analyze the research results, methods such as qualitative functional analysis, descriptive technical assessment, and thematic grouping of user feedback were used. Integration errors were categorized by type, source, and API interaction phase. System activity logs were reviewed for compliance with specified performance and reliability standards. User evaluations were grouped into recurring thematic categories that highlighted both the system's strengths and areas for future improvement.

Results

II Development of the architecture of the subsystem for organizing WEB access to information resources of the software package

With the advent of computer technologies and especially with the advent of automated systems in education, the labor intensity of collecting, preparing information and processing results, obtaining reasoned data in the established manner has sharply decreased.

Increased data reliability is achieved through integration with university learning management systems (LMS) and related government higher education management systems.

The university education management system allows entering data on the students, personnel, content of educational programs, and educational, scientific, and laboratory facilities, as well as the infrastructure as a whole. Information on students is entered into the student body management system, where students are grouped by educational programs and compared with the curriculum. The academic performance system allows automating the academic performance monitoring processes based on a rating system for assessing students' knowledge. The testing system allows testing students registered at universities in the disciplines included in the curriculum.

Information about staff is available in their personal profiles, where their academic and scientific portfolio, personal achievements are entered, with a connection established with their workload and schedule.

Data on educational programs implemented in universities on a national scale are concentrated in the Register of EPs of Higher and Postgraduate Education (goal, objectives, learning outcomes and a list of disciplines with a description). More detailed information on EPs with a detailed curriculum is in the HEIs training management system with a link to individual curricula of students, the Catalog of disciplines.

Detailed information on the university infrastructure is available in the HEI management system (educational areas, laboratory facilities, computer equipment and its administration systems, technical characteristics and capacities, residential areas, etc.).

All data must be connected in the digital ecosystem of the HEI and transferred to the Unified Platform of Higher Education (UPHE). The digital ecosystem of universities is managed through different levels of data administration, where the roles and level of access of each participant in the educational process are determined.

When transferring data, their correctness is checked for each direction and their affiliation. Then the data is automatically exported to the system.

Thus, the data transfer procedure can reveal quite a lot of incorrect data that is not directly related to

its purpose. The data updating procedure involves automatic correction of some data when changes are detected in other data. To perform the data updating procedure, the IS uses a generalized repository that describes the relationships between different DBMS, programs, components and ensures the efficiency of IS management.

Transferring data via API

To ensure the integration of the information system “Quality Management of Higher and Postgraduate Education” with other systems, such as accounting systems for teachers, students and university infrastructure, an API was developed and used (<https://hedu.kz/api/swagger/#/>).

API allows you to automate data exchange processes, ensuring fast and efficient transfer of information between systems.

The process of transferring data via API includes the following steps:

- 1) Authentication: Data transfer begins with authentication in the system using API keys to ensure security and limit access to authorized users only;
- 2) request formation: the user or application creates a request to the API, specifying the necessary parameters and data to be transferred;
- 3) sending a request: the generated request is sent to the server where the API is running, using standard communication protocols;
- 4) Request processing: the API server receives the request, analyzes it and performs the necessary actions, including accessing the database or other data sources;
- 5) generating and sending a response: after processing the request, the server generates a response containing the requested data or the results of the operation and sends it back to the client;
- 6) receiving and processing the response: the user receives a response from the server and processes the received data in accordance with the application logic.

Errors and their identification

During testing of the information system using the API, the following errors were identified:

- 1) Authentication error: Incorrect transmission or use of authentication data resulted in API access being denied or incorrect data being received;
- 2) Query generation error: An incorrect query generation caused the API server to misinterpret the request and return incorrect results;
- 3) Data validation error: Incorrect or incomplete data passed through the API resulted in errors during server-side processing.

To eliminate these errors and ensure more stable and reliable operation of the information system, thorough API testing was carried out at all stages of development, and its operation was regularly monitored during operation.

Data transmitted

During the testing of the information system “Quality Management of Higher and Postgraduate Education” on experimental sites of universities in Kazakhstan, it was revealed that universities actively use API to transfer various information. Volumes of data transferred via API:

- information about teachers:

- 1) personal data of teachers (full name, contact information, qualifications and work experience);
- 2) information on the admission of teachers to advanced training courses and academic programs;
- 3) information about foreign teachers;
- 4) information on the mobility of teachers;

- information about the university's international projects:

- 1) information on current international agreements with foreign educational organizations;
- 2) information on participation in international scientific projects with foreign organizations;
- 3) information about branches/representative offices abroad and foreign universities;
- 4) information on participation in international educational projects with foreign organizations;
- 5) the number of publications in Scopus and Web Of Science by field of activity;

- information about students:

- 1) personal data of students (full name, contact information, date of birth);
- 2) information on admission, education and graduation of students;

- 3) information on the current academic performance of students, including grades and attendance;
- 4) participation of students in international exchanges and academic mobility projects.

Identified problems and suggestions

During testing, the following problems and suggestions for their elimination were identified:

- problems with data display:

1) in some reports the data was displayed incorrectly, which required additional checks and corrections;

2) the ability to manually enter and correct data in reports was added;

- delays in loading data:

1) there were periodic delays when loading large amounts of data;

2) it was recommended to optimize data processing algorithms and improve server performance;

- interface improvements:

1) users suggested adding additional filters and settings to the information display interface;

2) It was suggested to improve navigation through the system and make it more intuitive.

System activity logs

Analysis of system activity logs during the testing period showed stable operation of the IS. The average system response time was 200-300 ms, which corresponds to the stated requirements. Minor system failures were identified and promptly eliminated.

User Reviews

Users who participated in the testing provided the following feedback and suggestions:

- positive reviews:

1) convenient and intuitive interface;

2) broad opportunities for managing educational programs and monitoring the quality of education;

3) high performance and stability of the system;

- suggestions for improvement:

1) adding additional settings and filters;

2) optimization of data loading speed;

3) improved navigation through the system and simplification of some functions.

The use of advanced software technologies in the development of the Smart University system and the modern level of communications development allow for prompt access to system data via the Internet or the corporate network of universities.

When providing WEB access to an existing DB, a number of ways are possible – complexes of technological and organizational solutions. The practice of using WEB technology to access an existing DB provides a wide range of technological solutions, interconnected in different ways – overlapping, interacting, etc. The choice of specific solutions when providing access depends on the specifics of a particular DBMS and on a number of other factors, such as: the availability of specialists capable of mastering a certain branch of technological solutions with minimal costs, the existence of other DBs, WEB access to which should be carried out with minimal additional costs, etc.

The implementation of corporate information systems based on the Internet/Intranet architecture is based on the principle of "open architecture", which allows for a corporate system implementation that is quite independent of a specific manufacturer. All software for such systems is implemented in the form of applets or servlets (programs written in the JAVA language), or in the form of CGI modules (programs written, as a rule, in Perl or C, C++).

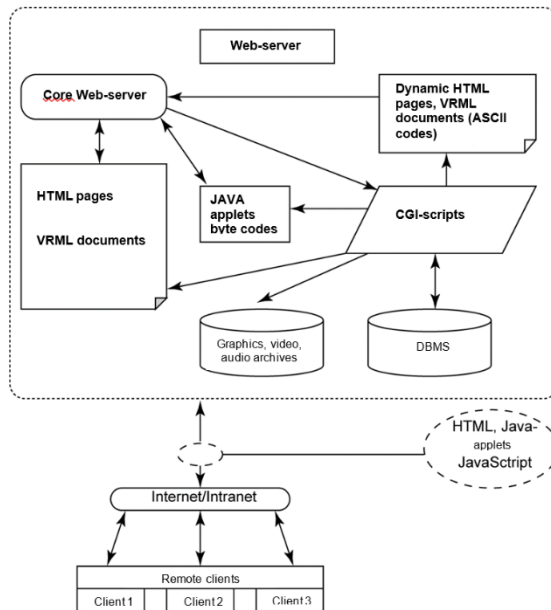
The main economic advantages of such architecture include: relatively low costs of implementation and operation; high ability to integrate existing information resources of corporations; increased level of efficiency of equipment use; availability of application software from any workstation with appropriate access rights; minimum composition of software and hardware at the workstation; minimum costs of setting up and maintaining client workstations, which allows implementing systems with a very large number of users.

The traditional client-server scheme is simple: the client sends requests, and the server responds. Because the client and server are different computers, they are able to split (or divide) the computing load, allowing work to be done more quickly. Figure 1 shows a functional diagram of user interaction in an

Internet/Intranet architecture.

Figure 1

Functional diagram of interactive user interaction in the Internet/Intranet architecture



At the client's

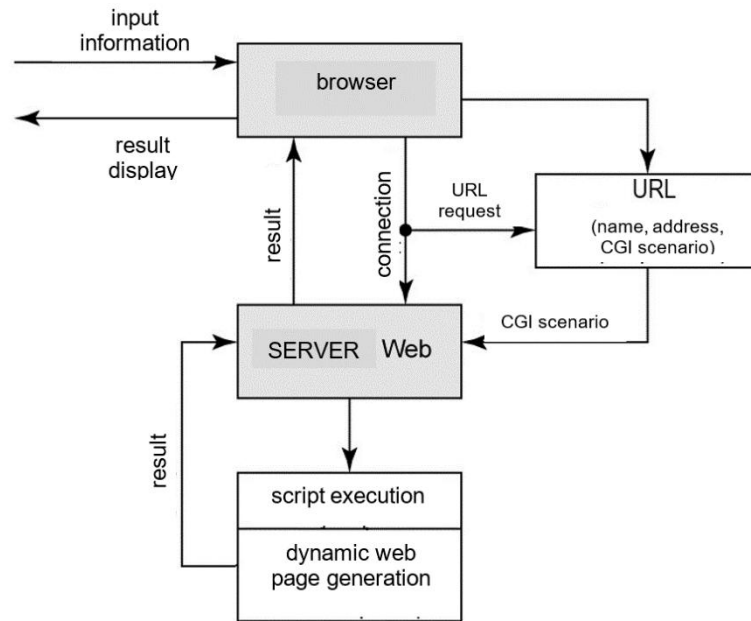
- sends ASCII codes of HTML pages (or VRML documents), JavaScript elements;
- sends images, audio, video archive files, etc.;
- sends byte codes of JAVA applets;
- receives information from the user (the result of filling out an active form or statistical information requested by a CGI script);
- fills the database;
- receives notifications from the user and regulates access to the WEB site resources;
- depending on the user's information, dynamically generates HTML pages or VRML documents, accessing, if necessary, databases and existing HTML pages and VRML documents on the WEB site. Figure 2 shows the interaction diagram between the browser, server, and CGI script.

The client, after receiving a response from the WEB server, does the following:

- visualizes an HTML page or VRML document in the browser window;
- interprets Javascript commands, modifies the image of an HTML page, etc.;
- by interpreting byte codes of Java applets, it allows loading and executing active applications;
- maintains a dialogue with the user filling out forms and creates new requests to the WEB server;
- with the help of utilities, reproduces codes of audio and video files, supports multimedia tools;
- provides virtual reality simulation by viewing VRML documents.

Figure 3

Sequence of actions during interaction between a client and a program running on a WEB server.

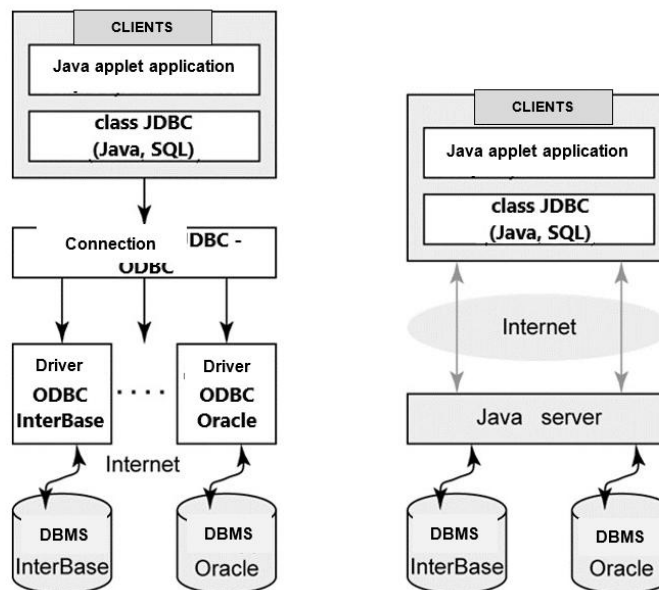


This makes clear the most important advantage - zero cost of administering client systems written in Java. It is enough to update the version of the object on the server, after which the client will automatically receive the new version, not the old one.

Being modular, applets must receive information from data storage devices, process it, and write it back for further processing by other applets. Monolithic programs can afford to have their own data processing schemes, but Java applets, which cross the boundaries of operating systems and computer networks, need to publish open data access schemes. Therefore, standard relational data access is important for Java programs, the first option of which is the JDBC (Java Database Connectivity) interface. Figure 4 shows the options for implementing JDBC communication with a database.

Figure 4

Options for implementing JDBC communication with a database



According to Internet conventions, JDBC identifies a database using a URL of the form:

jdbc:<subprotocol>:<name associated with DBMS or Protocol>

In an Internet/intranet database, "name" may contain a network URL

//<hostname>:<port>/..

<subprotocol> can be any name that implies a database. The subprotocol name "odbc" is reserved for ODBC format data sources. A typical JDBC URL for an ODBC database looks like this:

jdbc:odbc:<DNS - ODBC name>;User=<user name>;PW=<password>

II Organization of the user interface for accessing databases

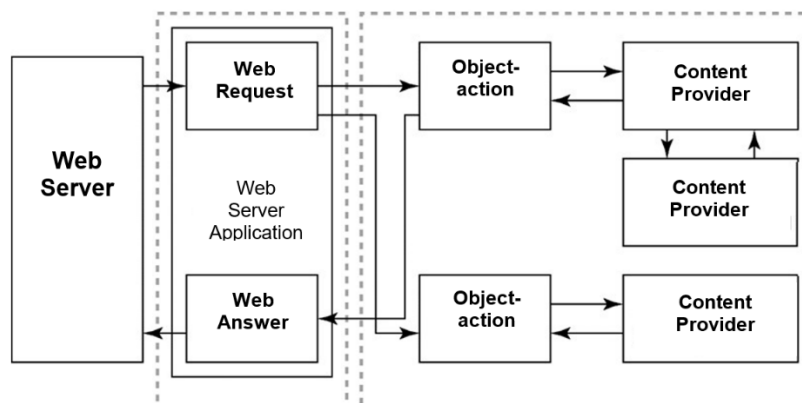
When providing WWW access to an existing DB, a number of ways are possible – complexes of technological and organizational solutions. The practice of using WWW technology to access an existing DB provides a wide range of technological solutions, interconnected in different ways. The choice of specific solutions when providing access depends on the specifics of a particular DBMS and a number of other factors, such as: the availability of specialists capable of mastering a certain branch of technological solutions with minimal costs, the existence of other DBs, WWW access to which should be carried out with minimal additional costs, etc.

In order to reduce the cost of the developed automated system of access to information in databases for Internet users, a proprietary WEB server for external access has been developed. In the Smart University system, the InterBase (Firebird) server is used to work with remote databases, where the concept of an active server core was first implemented. The functions of the InterBase active core include a patented mechanism of event signalers, stored procedures, triggers, user-defined functions (UDF - User-defined function) and BLOB filters. The joint work of these functions ensures the transfer of data processing to the server, where they are executed faster and more reliably. InterBase ensures high availability and data integrity using declarative initial integrity mechanisms, including cascade operations. Many applications (multimedia, scientific, Internet applications) require the ability to process unstructured data. InterBase is the first relational database to satisfy this requirement using BLOB. Using BLOB allows storing audio, video, graphic and binary information in the database. In modern applications, BLOB filters are used to compress and transform data. The compactness of the InterBase core saves disk space for later use by business-critical applications. InterBase also provides performance comparable to competing databases with lower memory requirements for additional savings on memory costs.

The C# programming environment adopted for the implementation of the developed system has great potential for creating your own WEB server by using, for example, WebBridge C# technology, without worrying about the choice of API (NSAPI or ISAPI). The general structure of the WEB server application is shown graphically in Figure 5.

Figure 5

General structure of WEB server application.



The task of the WEB server is to prepare the received data for the type of server application and send the user a response prepared by the application. The transmitted hypertext documents are formatted in the HTML standard – the language for describing hypertext documents. These documents are stored in static form (a set of files on a disk) and are dynamically compiled depending on the parameters of the request by

special software. For dynamic compilation of HTML documents, the WWW server uses specially designed CGI programs.

This solution is effective for large databases with a complex structure and when it is necessary to support search operations. The disadvantages of this method include a long time for query processing, the need for constant access to the main database, and additional loading of database support tools associated with processing queries from the WWW server.

To solve the problem of reducing the time of request processing, it is necessary to minimize the data transfer via the Internet and the Internet connection time between the client and the server. For example, for the network version of test control, minimization of requests to the server is solved by "downloading" the test selected for testing to a remote computer (client). After the test transfer is complete, the client no longer needs to contact the server, and it is disconnected until the moment when it needs to transfer the test results to the server.

The functioning of the server, which serves remote test subjects, consists of:

- launching the server and preliminary preparation of tests for their transmission to remote clients.

The server analyzes the registered profiles of the test subjects and selects the tests necessary for testing. The selected tests are then compressed for transmission and placed in a special "transport" folder on the disk;

- saving the results of the tests carried out in the main results database and terminating the server.

For a client wishing to undergo testing at a great distance, the following is performed:

- user identification by the server and receiving the file with the test. If the client with the entered password is registered, the server will transfer the file with the test to him;

- unpacking the test and testing, disconnecting from the server, the testing process;

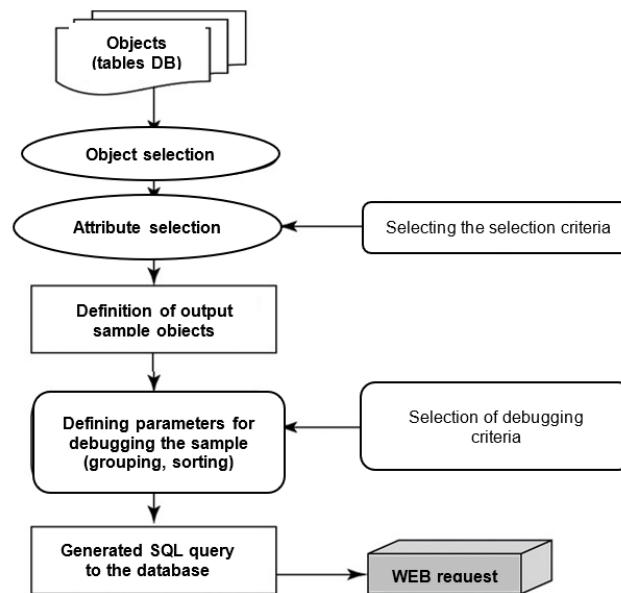
- connection to the server, transfer of the results of the testing to the server and completion of work by disconnecting from the server.

For working with databases in the application of WEB servers, the C# programming environment adopted for the development of the system under consideration provides a number of components: TQueryTableProducer (for providing the result of an SQL query to the database in the form of an HTML table), TDataSetTableProducer (similarly forms a table in the HTML language, into which it places all the records of the TDataSet object), TDataSetPageProducer (replaces templates in the HTML document template with the value of the corresponding database fields from the current record).

When receiving a request to execute, it creates TWebRequest and TWebResponse objects to place the HTTP request and response of the server in them. These objects are transferred to the WEB dispatcher (the dispatcher built into the WEB module or TWebDispatcher component), which controls the operation of the server application, supporting a set of action objects, TWebActionitem class objects that process various types of requests. If after completing the request processing, the dispatcher has not found a suitable action object, then the server message for sending to the client is not sent and the server breaks the connection with the client. This technology is also used when executing remote client requests for reference information about data in the database of the system in question.

For this purpose, the work proposes a "query constructor" that allows constructing various queries to the database, selecting the database objects needed for analysis from the available ones and setting the selection parameters based on their inherent attributes. The scheme of the "query constructor" operation is shown in Figure 6.

Figure 6
Scheme of the "Query Designer" operation



The results of generated SQL queries to the database can be presented to the user as HTML pages or as tables in the browser window.

Testing of the test version of the Information System "Quality Management of Higher and Postgraduate Education" in pilot mode showed that the system is generally operational, data exchange is carried out in accordance with the technical specifications. The technical parameters of the information system are met.

Bugs (errors) discovered during testing were subsequently corrected, and appropriate changes were made to the nodes.

Thus, it should be noted that the version of the Information System "Quality Management of Higher and Postgraduate Education" has passed testing and is ready for integration with other modules of the Unified Platform for Higher Education.

Discussion

The findings confirm that the integration of the information system "Quality Management of Higher and Postgraduate Education" with university-level and external national systems should not be viewed merely as a localized technical task of data exchange. Rather, it should be considered as a foundation for building a unified digital framework for managing the quality of higher and postgraduate education.

The pilot implementation at Astana IT University and the University of International Business demonstrated that API-based integration makes it possible to consolidate diverse categories of institutional data, including information on students, academic staff, educational programs, international activities, and research publications, within a single management system. The subsequent transmission of these data to the Unified Higher Education Platform shows the potential of such integration to reduce fragmentation, improve data consistency, and support more systematic quality assurance processes.

These results are consistent with contemporary research (Çelik & Ayaz, 2022; Daim et al., 2024; Mazadu et al., 2022), which emphasizes that the value of digital transformation is determined not by the number of implemented digital platforms, but by their interoperability, alignment with institutional processes, and integration into the university management cycle.

At the same time, the present study offers a more practice-oriented perspective compared to much of the reviewed literature. While many studies focus primarily on LMS adoption, user satisfaction, learning analytics dashboards, or the effectiveness of individual digital tools, this research proposes an architectural and functional model of intersystem interaction. This is one of the key strengths of the study, since the literature still provides limited examples of models that integrate LMS, SIS, quality assurance subsystems,

and external governmental platforms into a unified digital ecosystem.

In this regard, the proposed solution partially addresses the identified research gap by demonstrating how system integration can be implemented through specific procedures, including authentication, request formation, data validation, data updating, and transmission to external platforms. The study therefore contributes not only to the theoretical discussion of digital transformation in higher education, but also to the practical development of institutional quality management mechanisms.

Importantly, the pilot implementation revealed both the advantages and the limitations of the integration approach. The errors identified in authentication, request formation, and data validation indicate that the main risks of such systems are related not only to software implementation, but also to the quality of source data, the accuracy of exchange rules, and the reliability of intersystem interaction protocols. In other words, the existence of a functioning API does not automatically guarantee reliable, complete, or managerially meaningful integration.

Therefore, the further development of such systems requires special attention to master data management, unified reference data, formalized validation protocols, and continuous monitoring of data exchange quality. Without these elements, digital integration may remain a technical procedure rather than becoming a meaningful instrument for quality assurance and evidence-based decision-making.

Positive user evaluations of the interface, system performance, and quality-monitoring capabilities indicate the functional suitability of the proposed solution in real institutional contexts. However, these results should be interpreted with caution. First, the study was conducted as a pilot and was limited to two universities, which does not allow for broad generalization to the entire higher education system. Second, the evaluation focused mainly on the technical and organizational functionality of the model, rather than on its long-term effects on the quality of managerial decision-making, institutional performance, or educational outcomes.

Conclusion

The study demonstrated that the integration of quality management information systems with university and external national systems has significant potential for improving the management of higher and postgraduate education. The proposed API-based model enables the consolidation, validation, updating, and transmission of institutional data, thereby creating the basis for a more coherent and transparent digital quality assurance ecosystem.

The pilot implementation confirmed that such integration can improve the availability and consistency of data related to students, academic staff, educational programs, international activities, and research outputs. It also showed that intersystem interaction can support institutional monitoring and reporting processes, while reducing duplication and fragmentation of information flows.

At the same time, the study revealed that successful digital integration depends not only on technical connectivity, but also on the quality of data governance. The reliability of such systems requires standardized data structures, unified reference directories, clear validation rules, and continuous monitoring of data exchange. These factors are essential for transforming integration from a simple data transmission mechanism into a practical tool for supporting quality management.

The results suggest three main directions for further development. First, pilot implementation should be expanded to include a larger number of universities with different organizational and digital maturity levels. Second, data quality standards and validation procedures should be strengthened at both institutional and national levels. Third, analytical decision-support modules should be added to allow universities and authorized bodies to use the collected data not only for reporting, but also for strategic planning, risk identification, and evidence-based quality improvement.

Thus, the integration of quality management information systems can become an important element of the digital transformation of higher education. However, its full value will be achieved only when technological interoperability is combined with organizational regulation, reliable data governance, and analytical maturity. This combination can transform the system from a technical platform for data exchange into a full-fledged mechanism for managing the quality of higher and postgraduate education.

Conflicts of Interest Statement.

The authors declare no potential conflicts of interest regarding the research, authorship, or publication of this article.

Author Contributions

Aidos Mukhatayev: Conceptualization, Methodology, Supervision, Project Administration, Writing – Review and Editing. Serik Omirbayev: Conceptualization, Strategic Guidance, Validation, Supervision, Writing – Review and Editing. Andrii Biloshchyskyi: Methodology, Formal Analysis, Technical Expertise, Validation, Writing – Review and Editing. Aidar Ispussinov: Resources, Digital Infrastructure Analysis, Data Collection, Visualization, Writing – Review and Editing. Yurii Andrashko: Methodology, Statistical Analysis, Validation, Visualization, Writing – Review and Editing. Yuliya Idiyatova: Literature Review, Data Collection and Curation, Formal Analysis, Writing – Original Draft, Final Editing. Indira Abilmazhinova: Investigation, Data Analysis, Regulatory and Institutional Review, Writing – Original Draft, Writing – Review and Editing.

References

- Alduraywish, Y., Patsavellas, J., & Salonitis, K. (2022). Critical success factors for improving learning management systems diffusion in KSA HEIs: An ISM approach. *Education and Information Technologies*, 27, 1105–1131. <https://doi.org/10.1007/s10639-021-10621-0>
- Alam, M. I., Malone, L., Nadolny, L., Brown, M., & Cervato, C. (2023). Investigating the impact of a gamified learning analytics dashboard: Student experiences and academic achievement. *Journal of Computer Assisted Learning*, 39(5), 1436–1449. <https://doi.org/10.1111/jcal.12853>
- Bergdahl, N., Bond, M., Sjöberg, J., Dougherty, M., & Oxley, E. (2024). Unpacking student engagement in higher education learning analytics: A systematic review. *International Journal of Educational Technology in Higher Education*, 21, 63. <https://doi.org/10.1186/s41239-024-00493-y>
- Cabral, L., Pinto, R., & Gonçalves, G. (2025). AI-powered learning analytics dashboards: A systematic review of applications, techniques, and research gaps. *Discover Education*, 4, 525. <https://doi.org/10.1007/s44217-025-00964-y>
- Candra, S., & Jeselin, F. S. (2022). Students' perspectives on using e-learning applications and technology during the COVID-19 pandemic in Indonesian higher education. *Journal of Science and Technology Policy Management*, 15(2). <https://doi.org/10.1108/JSTPM-12-2021-0185>
- Çelik, K., & Ayaz, A. (2022). Validation of the Delone and McLean information systems success model: A study on student information system. *Education and Information Technologies*, 27, 4709–4727. <https://doi.org/10.1007/s10639-021-10798-4>
- Daim, T., Ozdemir Gungor, D., Basoglu, N., Yarga, A., & VanDerSchaaf, H. (2024). Exploring student information management system adoption post pandemic: Case of Turkish higher education. *Technology in Society*, 77, 102557. <https://doi.org/10.1016/j.techsoc.2024.102557>
- Drugova, E., Zhuravleva, I., Zakharova, U., & Latipov, A. (2024). Learning analytics driven improvements in learning design in higher education: A systematic literature review. *Journal of Computer Assisted Learning*, 40(2), 510–524. <https://doi.org/10.1111/jcal.12894>
- Echeverria, V., Fernandez-Nieto, G., Zhao, L., Palominos, E., Srivastava, N., Gašević, D., Pammer-Schindler, V., & Martinez-Maldonado, R. (2025). A learning analytics dashboard to support students' reflection on collaboration. *Journal of Computer Assisted Learning*, 41(1), e13088. <https://doi.org/10.1111/jcal.13088>
- Fan, S., Chen, L., Nair, M., Garg, S., Yeom, S., Kregor, G., Yang, Y., & Wang, Y. (2021). Revealing impact factors on student engagement: Learning analytics adoption in online and blended courses in higher education. *Education Sciences*, 11(10), 608. <https://doi.org/10.3390/educsci11100608>
- Fernández, A., Peralta, D., Herrera-Viedma, E., & Benítez, J. M. (2023). Digital transformation initiatives in higher education institutions: A multivocal literature review. *Education and Information Technologies*, 28, 12351–12382. <https://doi.org/10.1007/s10639-022-11544-0>
- Flanagan, B., Majumdar, R., & Ogata, H. (2022). Early-warning prediction of student performance and engagement in open book assessment by reading behavior analysis. *International Journal of Educational Technology in Higher Education*, 19, 41. <https://doi.org/10.1186/s41239-022-00348-4>
- Kleimola, R., Hirsto, L., & Ruokamo, H. (2025). Promoting higher education students' self-regulated

- learning through learning analytics: A qualitative study. *Education and Information Technologies*, 30(4), 4959–4986. <https://doi.org/10.1007/s10639-024-12978-4>
- Kleimola, R., & Leppisaari, I. (2022). Learning analytics to develop future competences in higher education: A case study. *International Journal of Educational Technology in Higher Education*, 19, 17. <https://doi.org/10.1186/s41239-022-00318-w>
- Mazadu, U. H., Ibrahim, M. M., Ibrahim, A. S., & Mansur, M. S. (2022). Examining the instructor management benefits of student information system: An empirical investigation. *Social Sciences & Humanities Open*, 6(1), 100322. <https://doi.org/10.1016/j.ssaho.2022.100322>
- Mijač, T., Jadrić, M., & Ćukušić, M. (2024). Measuring the success of information systems in higher education: A systematic review. *Education and Information Technologies*, 29, 18323–18360. <https://doi.org/10.1007/s10639-024-12564-8>
- Mohamed, N., Mahadi, B., Miskon, S., & Haghshenas, H. (2023). Determinants and influences of information systems integration in a public higher education context. *International Journal of Asian Business and Information Management*, 14(1). <https://doi.org/10.4018/IJABIM.330987>
- Palanci, A., Yılmaz, R. M., & Turan, Z. (2024). Learning analytics in distance education: A systematic review study. *Education and Information Technologies*, 29, 22629–22650. <https://doi.org/10.1007/s10639-024-12737-5>
- Paulsen, L., & Lindsay, E. (2024). Learning analytics dashboards are increasingly becoming about learning and not just analytics: A systematic review. *Education and Information Technologies*, 29(11), 14279–14308. <https://doi.org/10.1007/s10639-023-12401-4>
- Pham, P.-T., Lien, D. T. H., Kien, H. C., Chi, N. H., Tinh, P. T., Do, T., Nguyen, L. C., & Nguyen, T. (2022). Learning management system in developing countries: A bibliometric analysis between 2005 and 2020. *European Journal of Educational Research*, 11(3), 1363–1377. <https://doi.org/10.12973/eu-er.11.3.1363>
- Sayaf, A. M. (2023). Adoption of e-learning systems: An integration of ISSM and constructivism theories in higher education. *Heliyon*, 9(2), e13014. <https://doi.org/10.1016/j.heliyon.2023.e13014>

Information about authors

Aidos Mukhatayev - Candidate of Pedagogical Sciences, Professor, School of General Education Disciplines, Astana IT University, Astana, Kazakhstan, e-mail: mukhatayev.aidos@gmail.com, ORCID ID: 0000-0002-8667-3200.

Serik Omirbayev - Doctor of Economic Sciences, Professor, QAIRU, Astana, Kazakhstan, e-mail: serik.omirbayev@gmail.com, ORCID ID: 0000-0001-7643-3513.

Andrii Biloshchytskyi - Doctor of Technical Sciences, Professor, Vice-Rector for Science and Innovation, Astana IT University, Astana, Kazakhstan, e-mail: a.b@astanait.edu.kz, ORCID ID: 0000-0001-9548-1959.

Aidar Ispussinov - Vice-Rector for Digitalization, Astana IT University, Astana, Kazakhstan, e-mail: a.manas@astanait.edu.kz, ORCID ID: 0009-0001-4134-776X.

Yuliya Idiyatova - Department of Quality Assurance, Astana IT University, Astana, Kazakhstan, e-mail: yulia.idiyatova@gmail.com, ORCID ID: 0000-0002-1677-1745.

Indira Abilmazhinova - PhD student, Nazarbayev University, Astana, Kazakhstan, Astana, Kazakhstan, e-mail: Indira.Abilmazhinova@nu.edu.kz, ORCID ID: 0009-0003-9745-7196 (*corresponding author*)

Yurii Andrashko - Candidate of Technical Sciences, Associate Professor, Department of System Analysis and Optimization Theory, Uzhhorod National University, Uzhhorod, Ukraine, e-mail: yurii.andrashko@uzhnu.edu.ua, ORCID ID: 0000-0003-2306-8377.

Received: 09.04.2026

Revised: 29.05.2026

Accepted: 10.06.2026

Published: 30.06.2026