Reflections on the Functional Development of University Academic Management Systems: Optimization and Innovation Driven by Demand

Xinglei Xu

Zhejiang Ocean University, Zhejiang Province, China

DOI: https://dx.doi.org/10.47772/IJRISS.2025.905000200

Received: 23 April 2025; Accepted: 09 May 2025; Published: 06 June 2025

ABSTRACT

With the continuous advancement of information technology in higher education, academic management systems have gradually become key tools to enhance teaching quality, optimize management processes, and improve work efficiency. However, existing university academic management systems continue to suffer from several critical challenges, including inflexible functionalities, suboptimal user experience, and inadequate levels of intelligence, all of which necessitate urgent optimization and innovation to meet the evolving demands of modern educational environments. This paper analyzes the current status and existing problems of university academic management systems from a "demand-driven" perspective, offering optimization suggestions for modules including course selection, scheduling, grade management, academic warnings, and teaching feedback. Additionally, it explores feasible paths for innovative design of academic management systems in conjunction with intelligent technologies and mobile applications, aiming to provide theoretical support and practical guidance for the upgrading and transformation of university academic management systems.

Keywords: University academic management systems; Functional optimization; Intelligent design; Teaching management; System innovation

INTRODUCTION

With the rapid development of information technology, university academic management systems have gradually transitioned from traditional offline management methods to digital, networked, and intelligent management platforms [1, 2]. An efficient, stable, and well-functioning academic management system can not only enhance the regularity and scientific nature of teaching operations but also effectively meet the diverse needs of teachers and students during the teaching process [3]. However, many university academic management systems still suffer from unreasonable functionality, poor user experience, and low responsiveness, making it difficult to meet the increasingly complex and personalized demands of teaching management [4, 5]. Particularly in key modules such as course selection management, teaching scheduling, grade inquiry, and academic warnings, some system designs deviate from user needs, leading to frequent communication costs and operational barriers during usage.

Currently, most academic management systems are designed and developed with a "functionality implementation" focus, neglecting in-depth exploration and feedback mechanisms regarding the actual usage needs of system users (i.e., teachers, students, and academic management personnel) [6]. This "system-centered"

development model limits the adaptability and scalability of the system and reduces user satisfaction and work efficiency. Consequently, it has become a critical issue in the development of information technology in higher education to re-optimize the functional architecture and service logic of academic management systems, with a focus on a 'demand-driven' approach.

This paper focuses on university academic management systems, examining the typical challenges revealed during their practical implementation. Through a comprehensive analysis that integrates user research with evaluations of current functionalities, the paper identifies key deficiencies and areas for improvement within these systems. Based on these insights, the paper proposes demand-driven optimization strategies and innovative functional design concepts, aiming to enhance the operational efficiency, intelligence, and personalization of these systems. Specifically, the study provides both theoretical foundations and practical solutions for the ongoing optimization of academic management systems, with a particular emphasis on advancing the informatization, refinement, and intelligent development of educational administration. By integrating intelligent technologies, such as AI-driven decision support and mobile optimization, the proposed strategies contribute to the development of smarter, more efficient systems that can better meet the evolving needs of students, faculty, and administrators.

Current Status and Analysis of Problems in University Academic Management Systems

As the complexity of educational management in universities continues to grow, academic management systems have become essential in supporting teaching operations, information management, and service delivery [7]. Currently, many universities have established or introduced comprehensive academic management platforms that integrate functionalities such as course management, teaching schedules, grade management, student status management, and course selection systems [8]. These systems offer substantial support for both teaching activities and student course selection, thereby effectively advancing the informatization of academic management in higher education.

However, the pervasive adoption of academic management systems has also brought to light significant issues that demand critical attention. On one hand, numerous systems remain grounded in legacy design paradigms, resulting in obsolete functional architectures that lack the adaptability required to address evolving pedagogical complexities [9]. On the other hand, system development processes predominantly adhere to a 'function-centric' rather than a 'needs-driven' approach, often overlooking user experience and personalized requirements. Consequently, this misalignment exacerbates the disparity between system capabilities and end-user expectations [10, 11].

Specifically, the current major problems in university academic management systems can be summarized into the following four aspects (Fig. 1):

1) Inefficient functionality design and limited flexibility: Different user groups in academic environments—students, instructors, and academic staff—require the academic management system to support adaptive and streamlined operations across interconnected tasks. For example, students expect that changes in their academic status (such as entering academic probation, taking a leave of absence, or changing majors) will be automatically reflected in their course enrollment eligibility and schedules. However, current systems lack real-time synchronization across modules, requiring academic advisors to manually verify and adjust course access. This gap between user expectations and system behavior not only increases administrative workload but also leads to delays in academic planning. Instructors and administrators likewise demand flexible tools that reduce repetitive procedures in routine academic tasks. Yet the system’s rigid module-based architecture and low degree of cross-functional integration hinder operational efficiency. For instance, teaching resource allocation (such as reserving classrooms or laboratories) often involves unnecessarily complex approval workflows, and

the grade submission interface lacks clear, task-oriented guidance. These issues arise from a functional design approach that prioritizes static process control over dynamic user support, leading to diminished system effectiveness in real-world academic scenarios.

2) Suboptimal user experience due to inefficient interaction design: Users of academic management systems—particularly students and instructors—expect smooth, responsive, and user-friendly interfaces that support efficient task completion without cognitive overload. For students, this includes the ability to complete critical tasks such as course registration or transcript access without encountering delays or system instability. However, current systems often exhibit poor interface responsiveness, especially during peak usage periods (e.g., course selection), resulting in slowdowns or crashes that directly compromise the user experience. Instructors, on the other hand, require stable and streamlined workflows for tasks such as grade entry, syllabus uploads, and attendance tracking. Yet, the interface design is often unintuitive, requiring repetitive manual input with minimal automation support. For example, academic staff are frequently forced to process individual tasks one by one—such as course approvals or schedule changes—due to the system's lack of batch processing capabilities. Additionally, the absence of responsive web design and inadequate mobile optimization severely limits the system's usability across diverse usage contexts, including remote teaching or on-the-go access. These problems stem from a lack of user-centered interaction design and failure to consider the operational habits, pain points, and expectations of different user groups. As a result, the system not only reduces user satisfaction but also undermines efficiency across high-volume academic processes.

3) Inefficient data utilization due to inadequate intelligent support: With the increasing emphasis on data-driven management in higher education, users across all roles expect academic systems not only to store data but also to assist in meaningful decision-making. For example, academic staff and administrators need tools that can analyze patterns in course selection, student performance, and teaching activities to support curriculum planning and academic intervention. However, current systems remain confined to basic data entry and retrieval functions, offering little in terms of integrated analytics or predictive insights. Although large volumes of valuable data are generated through routine system usage, these datasets are underutilized due to the absence of intelligent processing modules. For instance, there is no functionality to automatically identify at-risk students based on early academic indicators, nor are there dashboards to help optimize the allocation of teaching resources based on historical usage trends. As a result, the system fails to support proactive decision-making and timely pedagogical adjustments.

4) Absence of user feedback mechanisms: Users—whether students, faculty, or academic staff—expect that their actual usage experiences will inform the system's ongoing development, ensuring that updates are responsive to real needs. However, most academic management systems lack structured channels for collecting, analyzing, and incorporating user feedback. As a result, system iterations are often driven by technical feasibility rather than by pedagogical or operational priorities. This misalignment manifests in two key ways: First, new features are frequently introduced that fail to address users' core workflow challenges. Second, long-standing pain points—such as inefficient navigation paths or missing batch operation tools—persist unresolved across multiple update cycles. The absence of mechanisms like embedded feedback prompts, structured satisfaction surveys, or participatory co-design sessions means that valuable experiential data from end-users is systematically overlooked. Ultimately, this disconnects results in a system that evolves in isolation from its users, undermining both satisfaction and long-term adoption.

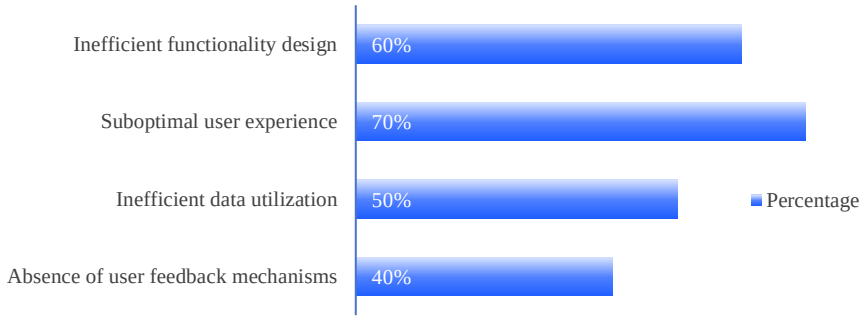

Fig. 1 Major Problems in the Current University Academic Affairs System

In summary, while current university academic management systems achieve high functional completeness, they still suffer from deficiencies in architectural rationality, functional practicality, user experience, and intelligent services. The fundamental cause of these issues stems from non-user-centric system development, coupled with insufficient continuous optimization and dynamic adaptation mechanisms. Consequently, re-engineering university academic management systems through a demand-driven approach holds significant practical importance and application value.

Demand-Driven System Optimization Ideas and Methods

As academic management systems continue to expand their functionalities, improving system practicality and user satisfaction has emerged as a critical challenge for higher education digitalization initiatives [12]. Compared to traditional "function-driven" development models, demand-driven optimization concepts place a greater emphasis on user participation, real situational analysis, and dynamic adaptability [13, 14]. They propose that system development should prioritize user operational efficiency and comfort to ensure close alignment between functional design and real-world usage [15].

A. Core Concept of Demand Driven

"Demand-driven" does not merely refer to fulfilling individual user requests, but rather involves identifying the commonalities and specific needs of different user groups through systematic research and analysis, and subsequently constructing a functional system that is both extensible and adaptable [16]. This concept establishes users' central role in system development, advocating for teacher-student co-design participation to improve perceived usefulness and user satisfaction.

B. User Demand Research Methods

To holistically capture user experiences and requirements in academic management system usage, multidimensional investigations should be implemented during the initial development phase (Table 1). Primary data collection methods include:

1) Questionnaire surveys: Structured questionnaires will be designed for distinct stakeholder groups (e.g., teachers, students, and academic staff) to collect data on usage patterns, satisfaction levels, and improvement suggestions across typical usage scenarios.

2) In-depth interviews: Purposive sampling will be employed to recruit representative users for in-depth, semi-structured interviews, focusing on pain points and unmet needs identified during their interaction with [the system/technology].

3) System usage log analysis: System log analysis will be performed to mine operational records from the backend, identifying high-frequency usage paths and exit points. These data will be analyzed to infer potential usability issues in the interface design.

4) Feedback closure mechanism: A dedicated user feedback interface will be implemented within the system, enabling real-time submission of usability issues and improvement suggestions. This feature will be coupled with an automated ticketing system to ensure prompt response tracking and systematic resolution of reported concerns.

TABLE 1 COMPARISON OF DATA COLLECTION TECHNIQUES FOR USER DEMAND RESEARCH

Data Collection Method	**Advantages**	**Disadvantages**	**Suitable For**
Questionnaire Surveys	Easy to distribute, large sample size	Limited depth of responses	Gathering general user opinions
In-depth Interviews	Provides detailed insights	Time-consuming, small sample size	Exploring specific user challenges
System Usage Log Analysis	Objectively reflects user behavior	Requires backend data, may not cover all issues	Identifying usage patterns
Feedback Closure Mechanism	Real-time user feedback, direct interaction	May lead to incomplete data if not well promoted	Immediate problem resolution

By integrating insights from surveys, interviews, system logs, and real-time feedback, a dynamic and evolving user requirements database can be established. This comprehensive dataset will provide a solid empirical foundation to inform iterative design improvements and guide future feature development with greater user alignment.

C. Analysis of Differentiated User Group Needs

Academic management systems serve diverse user groups with distinct needs and priorities. To ensure the system design is responsive to these differences, we recommend a structured assessment strategy that combines multiple data collection methods—including questionnaire surveys, in-depth interviews, system usage log analysis, and real-time feedback mechanisms (see Table 1). This approach enables a more comprehensive understanding of role-specific requirements during the early stages of system development.

For students, questionnaire surveys can be used to gather general feedback on course selection, grade inquiry, and overall system satisfaction, while micro-feedback prompts embedded in key processes (e.g., after course enrollment) can help capture immediate reactions. In addition, system logs should be analyzed to evaluate real usage behaviors, such as course selection completion times and drop-out rates, which may reveal hidden usability issues.

For teachers, in-depth interviews are especially valuable in uncovering workflow-related challenges, such as difficulties in submitting grades or managing course evaluations. These interviews can be supplemented by simulated task walkthroughs to better understand where friction points occur in actual operations. System logs can also provide insights into frequently used functions and possible inefficiencies in instructional management.

For administrative staff, a combination of structured feedback mechanisms and usage data can support the prioritization of system features—such as curriculum planning, scheduling, and reporting. Workshops or focus group discussions may be used to gather consensus on which functions require improvement, while heuristic evaluations can assess whether the current system interfaces meet standard usability benchmarks.

By integrating findings from these methods, designers can build role-specific requirement matrices to guide modular interface design. This enables the system to streamline interfaces for each user type (e.g., simplifying student views, tailoring administrative dashboards) and reduce unnecessary complexity. Continuous evaluation, such as quarterly user feedback reviews or diary studies, is recommended to adapt the system to evolving institutional demands over time.

D. Translating Requirements into System Functions

The effective translation of user requirements into system functionalities necessitates both rigorous logical analysis and methodical design processes (Fig. 2). Critical components include:

1. **Requirement Collection:** Analyze user needs based on typical scenarios and create operational flowcharts to identify key interaction points and workflows.

2. **Functional priority evaluation***:* Score and rank functions based on dimensions such as frequency of user feedback and functional impact scope, then prioritize development pace accordingly.

3. **Modular design***:* Categorize system functionalities into basic, extended, and personalized modules to ensure core requirements are met while maintaining flexibility for future scalability.

4. **User participatory testing***:* Engage selected users in prototype testing and usability trials throughout system development and iterative cycles, collecting feedback to refine design details in real time.

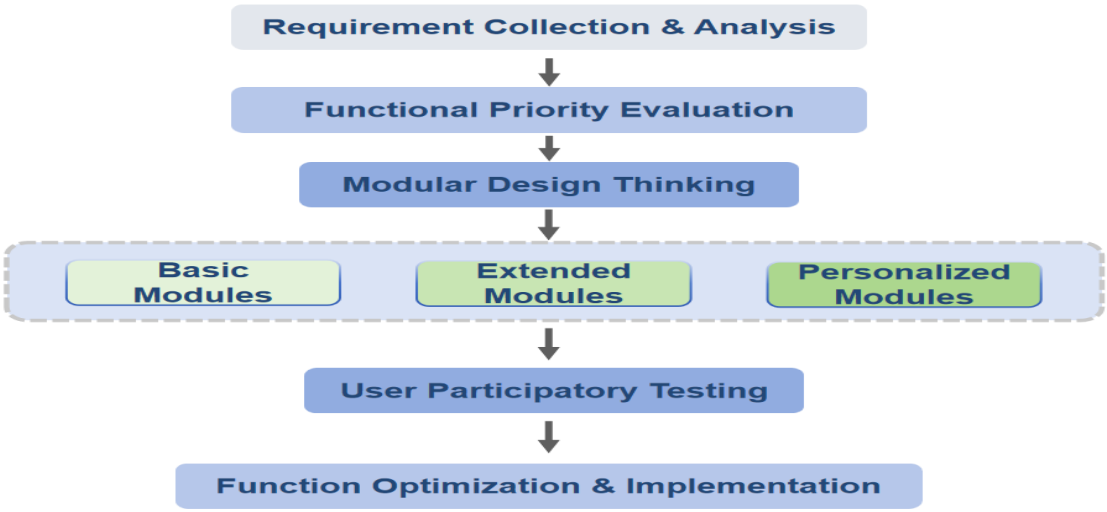

Fig. 2 Process of Translating User Requirements into System Functions with Modular Design

E. Establishing a Mechanism for Continuous Optimization

Demand-driven functional development in academic management systems should not be a one-time initiative but rather an institutionalized practice in system construction. Universities should institutionalize systematic feedback mechanisms for academic management systems, deploy specialized support teams, and conduct periodic evaluations to optimize operational performance and user experience. Furthermore, leveraging data mining and artificial intelligence to analyze and predict user behavior enables self-adaptive system optimization, thereby facilitating intelligent evolution in academic management systems.

Functional Optimization and Innovative Design of Academic Management Systems

A comprehensive analysis of current university academic management systems reveals that while existing platforms adequately support core administrative functions, significant potential remains for enhancing intelligent services, improving system flexibility, and developing advanced data analytics capabilities [17]. Consequently, adopting a demand-driven approach necessitates both systematic optimization of core academic management system functionalities and exploration of innovative features (Table 2). This dual focus will better address diverse user requirements while advancing the digital transformation of university academic administration. [18-20].

A. Optimization Design Of course Selection and Scheduling Modules

As the most actively utilized component of academic management systems, the course selection module generates the highest volume of user feedback regarding operational challenges. To mitigate persistent issues including information disparities and system latency, several optimization strategies can be proposed (Fig. 3):.

1. **Timed Course Selection:** Assign specific course selection time slots based on grade levels or subject categories. This strategy helps distribute server load and enhance overall system stability.

2. **Course Recommendation System:** Leverage historical course performance data alongside students' majors to intelligently recommend relevant courses, thereby improving the efficiency of course selection.

3. **Dynamic Course Capacity Indicators:** Provide real-time updates on course availability to reduce the need for students to repeatedly refresh the system.

4. **Visualized Class Schedule Previews**: Display potential time conflicts between already selected and newly chosen courses during the selection process to improve usability. For example, the system can prevent students from enrolling in courses that overlap with their existing schedule.

The scheduling system, as a core component of academic management, plays a crucial role in optimizing course arrangements and resource utilization. However, it also encounters technical limitations and challenges in adaptability. Several optimization strategies are proposed to address these issues (Fig. 3):

1. **Manual Scheduling:** Provide a drag-and-drop, what-you-see-is-what-you-get (WYSIWYG) interface that allows users to rearrange courses easily within and across weekly schedules.

2. **Automated Scheduling:** Utilize scheduling algorithms to automatically generate course timetables. The system should also support fine-tuned scheduling by considering variables such as teaching tasks, personnel, class types, time slots, and classroom availability.

3. **AI Scheduling Algorithms:** The integration of artificial intelligence-driven scheduling algorithms can optimize course timetabling by simultaneously analyzing multiple parameters including classroom capacity, instructor availability, and historical enrollment patterns. This intelligent scheduling approach achieves three critical objectives: (1) balanced course distribution across available timeslots, (2) prevention of classroom overutilization, and (3) reduced wear on shared multimedia resources, thereby extending their operational lifecycle while improving overall resource allocation efficiency.

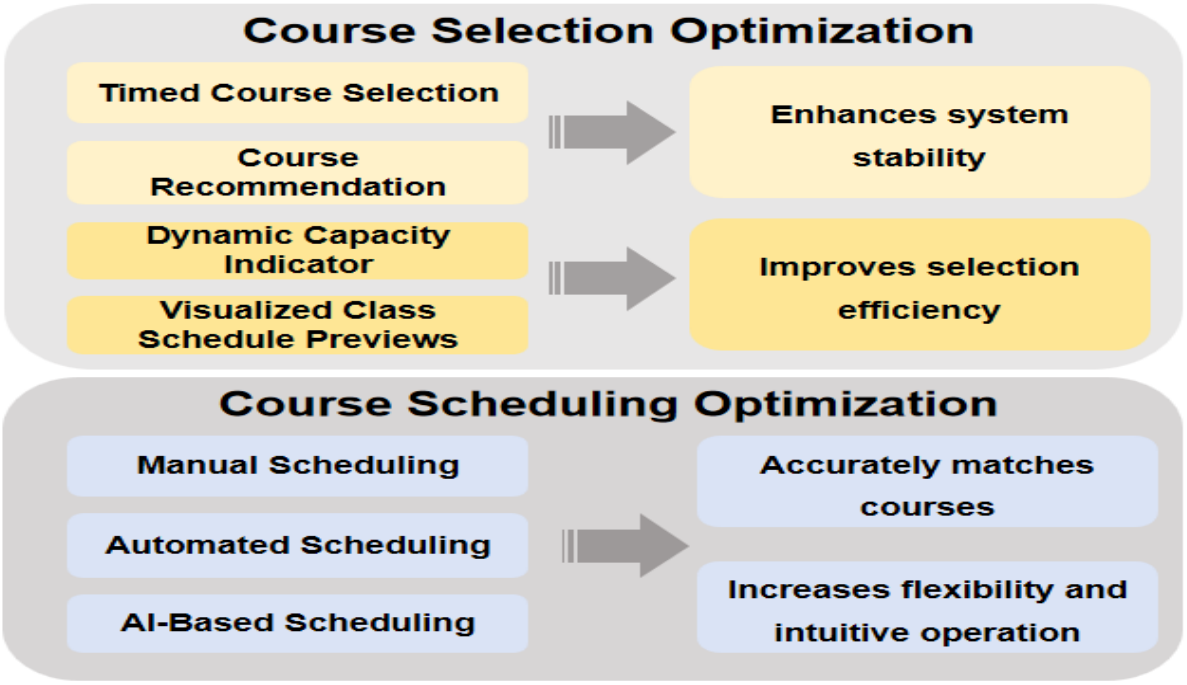

Fig. 3 Schematic diagram of course selection and scheduling optimization

B. Expansion of Grade Management and Academic Warning Functionality

Grade management constitutes a critically sensitive component of academic administration systems, with direct implications for student welfare and institutional quality assurance. To enhance this core functionality, optimization efforts should prioritize the following dimensions:

1. **Multi-dimensional grade input and evaluation:** Enable instructors to configure flexible grading components (e.g., continuous assessment, class participation, and final examinations) with customizable weight allocation templates, ensuring comprehensive evaluation of student performance.

2. **Clarification of grade review processes:** Implement a blockchain-based digital signature system to authenticate all grade-related transactions (entry, submission, and modification), establishing an immutable audit trail for academic integrity verification.

3. **Academic warning system**: Develop predictive analytics models leveraging historical performance data, course failure patterns, and enrollment load metrics to automatically identify at-risk students and trigger tiered notifications to relevant stakeholders (faculty advisors, academic counselors, and guardians).

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume IX Issue V May 2025

C. Design of Teaching Feedback and Quality Management Systems

Traditional teaching evaluation systems predominantly utilize standardized end-of-semester surveys, resulting in two critical limitations: (1) substantial time delays in feedback delivery, and (2) insufficient granularity for meaningful pedagogical improvement. To overcome these constraints, the academic management system should integrate the following innovative features:

1. Whole process teaching evaluation mechanism: Allow students to submit course experience feedback in real-time throughout the semester, with the system automatically categorizing and counting responses for timely teaching adjustments.

2. Bi-directional evaluation system: Allow students to evaluate teachers while also enabling teachers to provide feedback on students' classroom performances, promoting teaching interactions.

3. Automatic generation of teaching evaluation reports: Integrate evaluation results, teaching data, and student feedback to automatically generate reports on teaching quality for review by teaching and research departments or academic supervisors.

TABLE 2 ACADEMIC MANAGEMENT SYSTEM OPTIMIZATION HIGHLIGHTS

MODULE	CORE INNOVATIONS	KEY TECH/BENEFITS
Course Selection	- Staggered enrollment slots - AI course recommendations - Real-time capacity visualization	Reduces server load Personalizes student pathways
Scheduling	- Drag-and-drop WYSIWYG editor - AI-driven resource balancing	Prevents classroom overload Optimizes teacher availability
Grade Management	- Customizable grading templates - Blockchain-audited review process	Ensures transparency Supports diverse evaluation methods
Academic Warnings	- Predictive risk models - multi-stakeholder alerts (parents/staff)	Early intervention Data-driven decision support
Teaching Quality	- Real-time feedback analytics - Automated report generation	Enables agile teaching Reduces administrative workload
Mobile Experience	- Role-based homepage - Smart calendar sync & conflict detection	24/7 access Context-aware notifications

D. Support for Mobile and Personalized Functions

To meet the evolving user expectations and institutional demands, future academic management systems must embrace both intelligent technologies and mobile platforms. This integration will significantly improve service

accessibility and broaden institutional coverage across diverse educational contexts. Specifically, academic institutions should prioritize the development of mobile-optimized systems that cater to a wide range of users and offer personalized, data-driven features.

1. Development of academic mini-programs/apps: Enable mobile access to core functions, such as course registration alerts, grade notifications, and class schedule queries. By leveraging intelligent mobile applications, students and faculty can receive real-time updates and interact with the system anytime, anywhere, improving user engagement and system efficiency.

2. Personalized homepage customization: The system should automatically display content based on user roles and frequency of use. For example, students would have quick access to course and grade information, while teachers would see scheduling and submission reminders. By utilizing AI-driven data analytics, personalized recommendations and alerts can be provided to users, making the system more intuitive and responsive to their needs.

3. Scheduling and Reminder System Integration: Automatically synchronize students' timetables, exam schedules, and course registration periods into a unified calendar interface. This system should incorporate smart notification features powered by AI, ensuring timely reminders for critical deadlines and events. A mobile-based academic assistant, leveraging natural language interaction, can further guide students through complex academic processes, such as credit auditing or tracking graduation requirements.

By integrating intelligent technologies, such as AI-powered data analytics and mobile-responsive interfaces, academic management systems will shift from being passive administrative tools to proactive, intelligent service platforms. These innovations not only address current usability gaps but also pave the way for more personalized and efficient management of academic services.

CONCLUSION

The optimization of university academic management systems represents both a technological advancement and a paradigm shift in educational administration - from control-oriented approaches to service-oriented models. Moving forward, institutions must adopt a user-centric approach, harness intelligent technologies, and develop next-generation systems characterized by efficiency, flexibility, and intelligence. These systems will provide robust support for enhancing the quality of higher education.

REFERENCES

1. Magno-Tan, M.J., et al., (2014). Cloud-Based College Management Information System for Universities. International Journal of Information and Education Technology. 4(6):508.
2. Sanyal, B.C., The use of computerized information systems to increase efficiency in university management. 1995: Unesco, International Institute for Educational Planning.
3. Coates, H., R. James, and G. Baldwin. (2005). A critical examination of the effects of learning management systems on university teaching and learning. Tertiary education and management. 11(1):19-36.
4. Wawak, T., Current Problems of University Management. 2014: Wydawnictwo UJ.
5. Avdeeva, T.I., et al., (2017). Problems and prospects of higher education system development in modern society.
6. Joshi, L.M., (2015). A research paper on college management system. International Journal of Computer

Applications. 122(11):32-44.
7. Abugre, J.B., (2018). Institutional governance and management systems in Sub-Saharan Africa higher education: Developments and challenges in a Ghanaian Research University. Higher Education. 75:323-339.
8. Falvo, D.A. and B.F. Johnson. (2007). The use of learning management systems in the United States. TechTrends: Linking Research & Practice to Improve Learning. 51(2).
9. Eludire, A., (2011). The Design and Implementation of Student Academic Record Management System. Research Journal of Applied Sciences, Engineering and Technology. 3(8):707-712.
10. Li, L., L. Shi, and Y. Zhang. Design and implementation of universities' Academic affairs management system. in 2012 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet). 2012. IEEE.
11. Keller, G., Higher education management: Challenges and strategies, in international handbook of higher education. 2006, Springer. p. 229-242.
12. Pandey, S.R. and S. Pandey. Developing a more effective and flexible learning management system (LMS) for the academic institutions using Moodle. in International Conference on Academic Libraries. 2009.
13. Gardner, L., D. Sheridan, and D. White. (2002). A web-based learning and assessment system to support flexible education. Journal of Computer Assisted Learning. 18(2):125-136.
14. Shoaib, M., et al., (2024). AI student success predictor: Enhancing personalized learning in campus management systems. Computers in Human Behavior. 158:108301.
15. Zaharias, P. and C. Pappas. (2016). Quality management of learning management systems: A user experience perspective. Current Issues in Emerging eLearning. 3(1):5.
16. Fan, H. and M.S. Poole. (2006). What is personalization? Perspectives on the design and implementation of personalization in information systems. Journal of Organizational Computing and Electronic Commerce. 16(3-4):179-202.
17. Vershitskaya, E.R., et al., (2020). Present-day management of universities in Russia: Prospects and challenges of e-learning. Education and Information Technologies. 25:611-621.
18. Iasechko, S., et al., (2022). Artificial intelligence in the modern educational space: problems and prospects. International Journal of Computer Science & Network Security. 22(6):25-32.
19. Stoykova, S. and N. Shakev. (2023). Artificial intelligence for management information systems: Opportunities, challenges, and future directions. Algorithms. 16(8):357.
20. Dahlstrom, E., D.C. Brooks, and J. Bichsel. (2014). The current ecosystem of learning management systems in higher education: Student, faculty, and IT perspectives.