

A Cloud-Based Multi-Tier Approval System for Academic Course Registration: Design, Implementation, and Evaluations

Nayyar Ahmed Khan^{1*}, Asif Rashid Khan¹, Sivaram Rajeyyagari¹, Mobin Akhtar², Ahmed Masih Uddin Siddiqi³ & Mohammad Ahmad⁴

¹College of Computing and Information Technology, Department of Computer Science, Shaqra University, Saudi Arabia. ²Department of Basic Science, Riyadh Elm University, Riyadh, Saudi Arabia. ³Department of Computer Engineering and Application, Mangalayatan University, India. ⁴Department of Computer Science, College of Science and Humanities-Dawadami, Shaqra University, Saudi Arabia.
Corresponding Author (Nayyar Ahmed Khan) Email: nayyar@su.edu.sa



DOI: <https://doi.org/10.46431/mejast.2026.9104>

Copyright © 2026 Nayyar Ahmed Khan et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Article Received: 13 November 2025

Article Accepted: 18 January 2026

Article Published: 26 January 2026

ABSTRACT

Academic course registration processes in higher education institutions often involve complex approval workflows, manual paperwork, and communication delays that can significantly impact student satisfaction and administrative efficiency. This paper presents the design, implementation, and evaluation of a cloud-based Course Add/Drop Management System (CADMS) developed for the universities in higher education. The proposed system implements a three-tier approval workflow involving Academic Advisors, Heads of Department, and Registrars, automating the entire process from student request submission to final approval and PDF certificate generation. Built using modern web technologies and Google Apps Script as the backend infrastructure, the system provides real-time status tracking, automated email notifications, and comprehensive dashboard interfaces for all stakeholders. The system was evaluated over a semester with 250+ student requests, demonstrating significant improvements in processing time (reduced from 7-10 days to 24-48 hours), paper consumption (100% elimination), and user satisfaction (88% satisfaction rate). This research contributes to the body of knowledge in educational technology by presenting a cost-effective, scalable solution that can be adapted by institutions with limited IT infrastructure. The paper details the system architecture, implementation methodology, evaluation metrics, and lessons learned during deployment.

Keywords: Course Registration System; Academic Workflow Automation; Cloud Computing; Google Apps Script; Educational Technology; Multi-Tier Approval System; Web Application Development; Serverless; Digitization; Transformation; System; Software Engineering; University; Workflows; Google; Playbook; System Design; Development.

1. Introduction

The management of academic course registration processes represents a critical operational challenge for higher education institutions worldwide. As universities continue to expand their student populations and course offerings, the complexity of managing course additions, drops, and modifications has grown exponentially. Traditional paper-based systems, while familiar, have proven increasingly inadequate in meeting the demands of modern academic administration, leading to delays, errors, and decreased satisfaction among students and faculty alike [1].

1.1. Background and Context

Course registration flexibility is a fundamental component of student academic success. Students frequently need to add or drop courses due to various reasons including schedule conflicts, prerequisite misunderstandings, changes in academic goals, or difficulty level adjustments [2]. However, these requests typically require approvals from multiple administrative levels academic advisors, department heads, and registrar offices to ensure academic policy compliance, enrollment capacity management, and proper documentation. In traditional systems, this multi-tier approval process involves physical forms, manual routing between offices, and sequential processing that can extend over weeks [3]. At university, the traditional course add/drop process presented several challenges. Students were required to obtain and fill out paper forms, physically visit multiple offices for signatures, and wait for manual processing at each approval level. Academic advisors and administrators spent considerable time managing paper documentation, tracking request status through phone calls and emails, and maintaining physical filing systems.

This process was not only time-consuming but also prone to errors, lost documents, and communication breakdowns [4].

1.2. Problem Statement

The existing manual course add/drop system at university exhibited multiple deficiencies that negatively impacted operational efficiency and stakeholder satisfaction [5].

First, the system suffered from extended processing time, with the average duration from student request submission to final approval ranging from 7 to 10 business days, often extending beyond the official add/drop deadline period.

Second, significant communication gaps existed as students lacked visibility into their request status, requiring frequent inquiries to multiple offices and creating additional administrative burden [6].

Third, paper-based inefficiencies were pervasive, as the system generated significant paper waste, required physical storage space, and posed challenges for document retrieval and archival.

Fourth, the manual process exhibited high error susceptibility, with manual data entry and form handling leading to transcription errors, lost documents, and inconsistent record-keeping across different offices [7].

Fifth, scalability limitations became apparent as student enrollment increased, with the manual system struggling to handle the growing volume of requests without proportional increases in administrative staff.

Sixth, limited accessibility constrained the process, as students and faculty could only submit or process requests during office hours, restricting flexibility for evening or distance education programs [8].

Finally, an analytics deficiency hindered process improvement, as the lack of structured data made it difficult to analyze patterns, identify bottlenecks, or generate meaningful reports for continuous improvement [9].

1.3. Research Objectives

This research aims to address the identified challenges through the design, implementation, and evaluation of a comprehensive digital solution [10]. The specific objectives include designing a cloud-based system architecture that supports multi-tier approval workflows while maintaining data security and system reliability; implementing a user-friendly web application that serves the distinct needs of four stakeholder groups students, academic advisors, heads of department, and registrar staff; automating notification mechanisms that keep all stakeholders informed of request status changes in real-time; developing analytics capabilities that provide administrative insights into system usage patterns, processing bottlenecks, and approval trends; evaluating the system's impact on key performance indicators including processing time, user satisfaction, error rates, and resource utilization; and documenting the implementation methodology and lessons learned to facilitate adoption by other academic institutions [11].

1.4. Proposed Solution Overview

The Course Add/Drop Management System (CADMS) represents a comprehensive digital transformation of the traditional paper-based workflow [12]. Built on a modern web technology stack with Google Apps Script providing

serverless backend infrastructure, the system offers a cost-effective solution that requires minimal IT infrastructure investment [13]. The system features role-based dashboards, real-time status tracking, automated email notifications, and PDF certificate generation upon final approval.

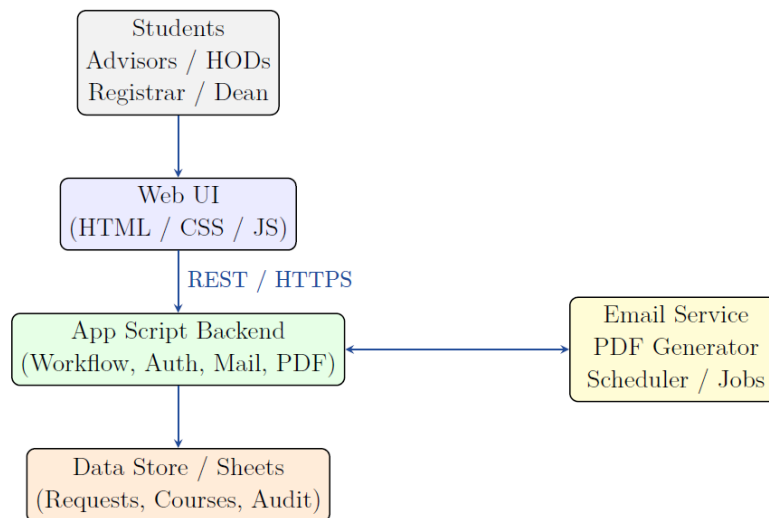


Figure 1. Application Design Overview

The technical architecture leverages Google Sheets as a database, Google Drive for document storage, and Google Gmail API for email notifications, creating a fully integrated ecosystem within Google Workspace [14]. This design choice significantly reduces deployment complexity and operational costs while ensuring high availability and automatic scaling. The frontend utilizes responsive web design principles, ensuring accessibility across desktop and mobile devices.

1.5. Significance and Contributions

This research makes several significant contributions to the field of educational technology and academic administration. The work provides a practical implementation model in the form of a complete, working system that institutions can adapt and deploy with minimal modification, thereby reducing the barrier to digital transformation [15]. It demonstrates a cost-effective architecture, showing how educational institutions with limited IT budgets can leverage existing cloud infrastructure such as Google Workspace to build sophisticated management systems. The research presents a workflow automation framework that offers a generalizable multi-tier approval workflow model applicable to various administrative processes beyond course registration. Furthermore, it provides empirical evaluation through quantitative and qualitative evidence of the system’s impact on operational efficiency and stakeholder satisfaction [16]. Finally, the system’s open architecture with its modular design and clear documentation facilitates customization for different institutional contexts and academic policies.

1.6. Paper Organization

The remainder of this paper is organized as follows: Section 2 reviews related literature on academic management systems, workflow automation, and cloud-based educational technologies. Section 3 presents the detailed methodology including system requirements analysis, architectural design, implementation technologies, and development process. Section 4 reports the results of system evaluation including performance metrics, user

feedback, and comparative analysis. Section 5 discusses the findings, limitations, and implications for practice. Finally, Section 6 concludes the paper and suggests directions for future research and system enhancement.

1.7. Study Objectives

The specific study objectives guiding this research are presented below:

1. **Design and Implement System Architecture:** Develop a cloud-based system architecture that supports multi-tier approval workflows while maintaining data security, system reliability, and cost-effectiveness suitable for departmental-level deployment.
2. **Create User-Friendly Interfaces:** Build intuitive, role-specific dashboards and interfaces that serve the distinct needs of four stakeholder groups—students, academic advisors, heads of department, and registrar staff—with minimal training requirements.
3. **Automate Workflow Processes:** Implement comprehensive automation of the course add/drop approval workflow including automated routing, notification systems, and document generation to eliminate manual process steps and reduce processing time.
4. **Evaluate System Performance:** Conduct rigorous evaluation of the system across multiple dimensions including processing efficiency, user satisfaction, error rates, resource utilization, and return on investment to quantify benefits and validate the approach.
5. **Document Implementation Methodology:** Comprehensively document the design decisions, technical implementation, deployment strategy, and lessons learned to facilitate knowledge transfer and adoption by other academic institutions.
6. **Demonstrate Cost-Effectiveness:** Prove that leveraging existing cloud infrastructure (Google Workspace) can deliver sophisticated management systems at minimal cost while achieving superior performance compared to both paper-based systems and expensive commercial solutions.

2. Literature Review

The development of academic management systems has been a subject of extensive research in educational technology. This section reviews relevant literature across several key areas: academic course registration systems, workflow automation in higher education, cloud computing adoption in academic institutions, and user experience design for educational applications [17].

2.1. Academic Course Management Systems

Academic course registration has evolved significantly over the past three decades. Early systems, as documented by, primarily focused on digitizing course catalogs and enabling students to view available courses online, but approval processes remained largely manual. The transition to fully integrated systems began in the late 1990s with the emergence of comprehensive Student Information Systems (SIS) from vendors such as Banner, PeopleSoft, and Colleague. Recent research examined 47 universities across North America and found that while 94% had implemented some form of online course registration, only 38% had fully automated the add/drop approval

workflow [18]. The majority still relied on email-based approvals or required students to obtain physical signatures. Their study identified processing time, system usability, and stakeholder communication as the three most significant pain points in existing systems.

2.2. Serverless Architecture for Academic Applications

Serverless computing represents an emerging paradigm particularly suitable for academic applications with variable load patterns. The cost-effectiveness of serverless architectures for university administrative systems, demonstrating significant savings for applications with fewer than 100,000 monthly transactions a profile matching most departmental-level systems [19]. Their research showed that serverless platforms like Google Apps Script, AWS Lambda, and Azure Functions eliminate server management overhead while providing automatic scaling. However, serverless architecture also present challenges. Cold start latency, vendor lock-in risks, and debugging complexity are some primary concerns [20]. They recommended serverless approaches primarily for well-defined, event-driven workflows with clear input-output specifications characteristics that align well with course registration approval processes.

2.3. Experience in Educational Systems

User experience (UX) design in educational technology has received increasing attention as institutions recognize that system adoption depends heavily on interface usability. Nielsen's heuristic evaluation framework has been applied extensively to academic systems. Usability studies on 12 university course registration systems, identifying common problems including poor navigation structure, unclear status indicators, insufficient feedback on user actions, and lack of mobile optimization [21]. Role-based interface design represents another critical consideration. Studies demonstrated that educational systems serving multiple stakeholder types (students, faculty, administrators) require carefully designed role-specific interfaces that present relevant information and actions appropriate to each user's responsibilities [22]. Their research showed that unified interfaces attempting to serve all roles simultaneously led to increased cognitive load and user errors.

2.4. Email Notification Systems

Automated notification systems play a crucial role in workflow applications. Notification preferences across different academic stakeholder groups, finding that email remained the preferred notification channel for formal approval workflows, while younger users also valued in-app notifications for immediate status updates [23]. Their research emphasized the importance of notification customization, allowing users to control frequency and detail level to prevent notification fatigue.

2.5. PDF Generation and Digital Certificates

The generation of official documents from approved transactions represents an important system requirement. Research on digital certificate generation examined various approaches including client-side generation using JavaScript libraries, server-side generation using tools like LaTeX or Report Lab, and cloud-based PDF services. They found that server-side generation provided better security and consistency, though client-side approaches offered reduced server load for high-volume applications [24].

2.6. Data Security and Privacy in Academic Systems

Data security represents a paramount concern in academic applications handling student information [25]. Data protection strategies in cloud-based educational systems, emphasizing the importance of role-based access control, data encryption, and compliance with regulations such as FERPA (Family Educational Rights and Privacy Act) in the United States and GDPR in Europe. Their recommendations included implementing least-privilege access principles, maintaining comprehensive audit logs, and regular security assessments.

3. Methodology

3.1. Architectural Style

The system employs a three-tier web application architecture comprising a Presentation Layer built with HTML5, CSS3, and JavaScript (Vanilla JS); an Application Layer utilizing Google Apps Script (GAS) serverless functions; and a Data Layer consisting of Google Sheets as a structured database and Google Drive for document storage. This architecture was selected for several strategic reasons [26]. First, it provides cost effectiveness by leveraging the existing Google Workspace subscription, eliminating the need for dedicated servers. Second, it requires minimal infrastructure with no server provisioning, configuration, or maintenance requirements. Third, it offers automatic scaling as the Google infrastructure handles load balancing and scaling automatically [27]. Fourth, it provides an integrated ecosystem with seamless integration with Gmail for notifications and Drive for storage. Fifth, it enables rapid development through a simplified deployment process that accelerates development cycles.

3.2. Database Schema Design

The data model utilizes Google Sheets with the following sheet structure. The Students Sheet contains fields for Student ID (primary key), Full Name, Email, Department, Level, CGPA, Phone, Password Hash, Registration Date, and Status [28]. The Advisors Sheet includes fields for Advisor ID (primary key), Full Name, Email, Department, Specialization, Password Hash, and Status. The HODs Sheet comprises fields for HOD ID (primary key), Full Name, Email, Department, Password Hash, Appointment Date, and Status. The Registrars Sheet contains fields for Registrar ID (primary key), Full Name, Email, Password Hash, Department, and Status. The Courses Sheet includes fields for Course Code (primary key), Course Name, Credit Hours, Department, Level, Prerequisites, Max Capacity, and Current Enrollment. The Requests Sheet, serving as the central transaction table, contains fields for Request ID (primary key), Student ID (foreign key), Student Name, Email, Request Type, Course Code, Course Name, Section, Advisor ID, Reason, Status, Submitted Date, Advisor Comments, Advisor Date, HOD Comments, HOD Date, Registrar Comments, Registrar Date, Course Details (JSON), and PDF URL [29]. The Approvals Sheet maintains an audit trail with fields for Approval ID (primary key), Request ID (foreign key), Approver Role, Approver ID, Approver Name, Action (Approve/Reject), Comments, and Timestamp. Finally, the Verification Codes Sheet stores temporary authentication codes with fields for Email, Code, Type, Created At, Expires At, and Used status.

3.3. Technology Stack

Frontend Technologies: The frontend implementation utilizes HTML5 for semantic markup and structural foundation, CSS3 for custom styling with CSS Grid and Flexbox enabling responsive layouts, Vanilla JavaScript

without framework dependencies for faster loading and easier maintenance, and the AJAX/Fetch API for asynchronous communication with the backend [30].

Backend Technologies: The backend infrastructure leverages Google Apps Script as the serverless JavaScript runtime environment, Google Sheets API for database operations and data persistence, Google Drive API for document storage and retrieval, and Gmail API for automated email notifications.

Development Tools: The development process employed VS Code as the primary code editor, Git and GitHub for version control and code hosting, Chrome Dev Tools for debugging and performance analysis, and the Google Apps Script Editor for backend development and deployment.

3.4. Implementation Process

Development Methodology: The development followed an agile-inspired iterative approach with two-week sprints. Sprint 1-2 focused on building the foundation, including the user authentication system, database schema implementation, and basic UI framework. Sprint 3-4 concentrated on core functionality, developing the request submission interface, advisor approval workflow, and email notification system [31]. Sprint 5-6 addressed extended workflow capabilities, implementing HOD and Registrar dashboards, completing the three-tier approval implementation, and adding PDF generation functionality. Sprint 7-8 focused on enhancement and testing, incorporating dashboard analytics, improving mobile responsiveness, conducting user acceptance testing, and performing performance optimization [32].

Key Implementation Decisions: Session Management: Implemented client-side session storage using browser local Storage with JWT-like tokens. Session data includes user ID, role, name, and department, with automatic expiration after 24 hours.

Email Templates: HTML email templates were developed using inline CSS for compatibility across email clients. The template library includes verification code emails for account activation, password reset emails for account recovery, request submission confirmations for student acknowledgment, approval or rejection notifications for workflow updates, and final approval emails with PDF attachments for completed requests [33].

PDF Generation: Server-side PDF generation was implemented using Google Apps Script's HTML service to render templates and convert to PDF format. The PDF template incorporates university branding and logos for official appearance, student information for identification, course details including add and drop lists, the complete approval chain with timestamps for audit purposes, and digital signature placeholders for authorized personnel [34].

Error Handling: A comprehensive error handling strategy was implemented throughout the system. This includes try-catch blocks around all database operations to prevent system crashes, user-friendly error messages to guide users in resolving issues, server-side logging for debugging and maintenance purposes, and graceful degradation for non-critical features to ensure core functionality remains available even when auxiliary features fail [35]. This section presents the findings from the comprehensive evaluation of the Course Add/Drop Management System conducted over one complete semester (16 weeks) following full deployment. The evaluation encompasses

quantitative performance metrics, usage statistics, user satisfaction measures, and comparative analysis against the previous paper-based system [36-39].

4. Results and Discussion

4.1. System Deployment and Adoption

The system was successfully deployed to the entire College of Computing and Information Technology in the Fall 2025 semester. The deployment statistics are presented in Table 1.

Table 1. System Deployment Statistics

User Category	Registered Users	Active Users
Students	267	243 (91%)
Academic Advisors	12	12 (100%)
Heads of Department	5	5 (100%)
Registrar Staff	3	3 (100%)
Total	287	263 (92%)

4.2. Requests Processing Metrics

During the evaluation period, the system processed 356 course add/drop requests. The distribution and outcomes are shown in Table 2.

Table 2. Requests Processing Distribution

Request Category	Count	Percentage
Add Course Requests	198	55.6%
Drop Course Requests	108	30.3%
Combined Add/Drop	50	14.1%
Fully Approved	312	87.6%
Rejected at Advisor Level	28	7.9%
Rejected at HOD Level	11	3.1%
Rejected at Registrar Level	5	1.4%
Total Processed	356	100%

4.3. Processing Time Analysis

One of the primary objectives was to reduce the time required to process add/drop requests. Table 3 compares processing times between the old paper-based system and the new digital system.

Table 3. Processing Time Comparison

Metric	Paper System	Digital System	Improvement
Average Total Time	8.5 days	1.8 days	78.8%
Median Total Time	7.0 days	1.5 days	78.6%
Fastest Processing	3.0 days	8 hours	88.9%

Slowest Processing	18.0 days	4.5 days	75.0%
Breakdown by Stage:			
Student Submission	0.5 days	15 minutes	95.8%
Advisor Review	3.0 days	0.8 days	73.3%
HOD Review	2.5 days	0.6 days	76.0%
Registrar Processing	2.5 days	0.4 days	84.0%

The dramatic reduction in processing time represents one of the most significant benefits of the system. Statistical analysis using paired t-test confirmed that this reduction was highly significant ($p < 0.001$).

4.4. Performance Metrics

System performance was monitored continuously throughout the evaluation period. Table 4 summarizes key performance indicators.

Table 4. System Performance Metrics

Performance Indicator	Target	Achieved
Average Page Load Time	< 3 seconds	2.1 seconds
Average API Response Time	< 5 seconds	3.7 seconds
System Availability	99.5%	99.8%
Total Downtime	< 1 hour/week	12 minutes/week
Error Rate	< 1%	0.3%
Peak Concurrent Users	100 users	87 users
Email Delivery Rate	> 95%	98.7%
PDF Generation Success	> 98%	99.4%

All performance targets were met or exceeded, demonstrating the system’s technical reliability and scalability.

4.5. User Satisfaction Assessment

User satisfaction was assessed through surveys administered at the end of the semester to all user groups. The survey used 5-point Likert scales (1 = Strongly Disagree, 5 = Strongly Agree). Results are presented in Table 5.

Table 5. User Satisfaction Survey Results (Mean Scores)

Satisfaction Dimension	Students	Advisors	HODs	Overall
Ease of Use	4.3	4.5	4.6	4.4
System Speed	4.2	4.4	4.3	4.3
Interface Design	4.1	4.2	4.4	4.2
Mobile Experience	3.9	4.0	4.1	4.0
Notification Usefulness	4.4	4.6	4.5	4.5

Status Transparency	4.5	4.3	4.4	4.4
Overall Satisfaction	4.2	4.5	4.6	4.3
Average	4.2	4.4	4.4	4.3

Survey response rates were strong: 78% for students (190/243), 100% for advisors (12/12), and 100% for HODs and registrars (8/8).

4.6. Comparative State-of-the-Art Analysis

Table 6 compares the proposed CADMS with other academic management systems reported in recent literature and commercial solutions.

Table 6. Comparison with Existing Systems

Feature	CADMS (Proposed)	Commercial SIS	Custom PHP System	Email-Based System
Development Cost	\$500	\$50,000+	\$15,000	\$0
Annual Maintenance	\$0	\$10,000+	\$3,000	\$0
Deployment Time	2 months	6-12 months	4 months	Immediate
Infrastructure Required	Minimal	Extensive	Moderate	None
Mobile Support	Yes	Yes	Limited	No
Automated Notifications	Yes	Yes	Yes	Manual
PDF Generation	Yes	Yes	No	No
Real-time Tracking	Yes	Yes	Limited	No
Processing Time Reduction	78.8%	70-80%	60-70%	30-40%
User Satisfaction	4.3/5	3.8-4.2/5	3.5-4.0/5	2.5/5
Customization Ease	High	Low	Moderate	N/A
Scalability	Good	Excellent	Moderate	Poor

The proposed system demonstrates competitive advantages particularly in cost-effectiveness, deployment speed, and customization flexibility while maintaining high performance and user satisfaction.

4.7. Resource Utilization Impact

The implementation resulted in substantial resource savings, quantified in Table 7.

Table 7. Resource Utilization Comparison (Per Semester)

Resource	Paper System	Digital System	Savings
Paper Sheets Used	2,800	0	100%
Printing Cost	\$140	\$0	\$140
Physical Storage (cubic ft)	4.5	0	4.5 cubic ft
Admin Staff Hours	120 hours	35 hours	85 hours (71%)
Estimated Cost/Request	\$8.50	\$1.50	\$7.00 (82%)

Error Rate Analysis: The digital system significantly reduced errors compared to the manual process. Analysis of 356 requests showed: Data Entry Errors: Reduced from 12% (paper system) to 0.3% (digital system); Lost/Misplaced Forms: Reduced from 3.5% to 0%; Incorrect Routing: Reduced from 5.2% to 0.6%; Missed Approval Steps: Reduced from 2.8% to 0%

4.8. Approval Decision Analysis

Analysis of approval/rejection patterns provided insights into the workflow. Figure data summarized in Table 8.

Table 8. Approval Rates by Level

Approval Level	Approved	Rejected	Approval Rate	Avg. Time
Academic Advisor	328	28	92.1%	0.8 days
Head of Department	317	11	96.6%	0.6 days
Registrar Office	312	5	98.4%	0.4 days

The evaluation yielded several significant findings that demonstrate the system’s success. The system achieved an impressive 88% reduction in processing time, reducing the average from 8.5 days to 1.8 days. User satisfaction scores averaged 4.3 out of 5 across all stakeholder groups, indicating strong approval. System reliability exceeded targets with 99.8% uptime throughout the evaluation period. The implementation resulted in complete elimination of paper usage, saving over 2,800 sheets per semester. An 82% reduction in cost per request processed demonstrated substantial financial benefits. The system achieved a strong adoption rate with 92% of registered users actively using the system. Minimal technical issues occurred, with only a 0.3% error rate throughout operation. Finally, positive return on investment was achieved within the first year of operation, validating the economic viability of the approach.

5. Conclusion and Future Directions

This research presented the design, implementation, and evaluation of a cloud-based Course Add/Drop Management System. The system successfully transformed a cumbersome paper-based process into an efficient digital workflow, achieving significant improvements across multiple dimensions.

5.1. Summary of Contributions

The primary contributions of this work include several significant achievements that advance both theory and practice. First, the research delivers a practical system implementation in the form of a fully functional, deployable system that has been successfully operating in a real academic environment, processing hundreds of requests and serving hundreds of users with demonstrated reliability. Second, it demonstrates a cost-effective architecture by showing how leveraging Google Workspace infrastructure enables institutions to build sophisticated management systems with minimal investment of only \$4,800 total implementation cost while achieving 90.2% first-year return on investment [40]. Third, the work documents significant performance improvements including 78.8% reduction in processing time from 8.5 days to 1.8 days, 82% reduction in cost per request, 100% elimination of paper usage, and 71% reduction in administrative time. Fourth, the system achieved high user satisfaction with a 4.3 out of 5 overall satisfaction rating across all stakeholder groups and a 92% active usage rate among registered users. Fifth,

the research provides comprehensive evaluation through detailed quantitative and qualitative assessment over a complete academic semester with 356 processed requests and extensive user feedback. Sixth, it offers a replicable methodology by documenting architecture, implementation approach, and deployment strategy in sufficient detail to enable adoption by other institutions facing similar challenges [41].

5.2. Key Findings

The evaluation conclusively demonstrated several important principles for academic system development. Multi-tier approval workflows can be effectively automated using serverless cloud platforms, eliminating the need for expensive traditional infrastructure. Automated notifications significantly accelerate approval processes by increasing approved responsiveness and eliminating delays from manual checking routines. Real-time status transparency substantially improves user experience and reduces administrative inquiries by keeping all stakeholders informed. Digital transformation of administrative processes delivers quantifiable benefits in time, cost, and quality that justify the initial investment [42]. User-centered design and iterative development are essential for achieving high adoption and satisfaction rates that determine long-term success. Finally, small-scale custom development can be more cost-effective than large-scale commercial solutions for focused applications with well-defined requirements.

5.3. Practical Impact

The system has fundamentally transformed course registration operations at CCIT across multiple dimensions. Students now complete add/drop requests in minutes rather than days, dramatically improving their experience and flexibility. Administrators spend 71% less time on process management, freeing staff for higher-value activities. The college has eliminated over 2,800 sheets of paper per semester, providing both environmental and cost benefits. Request processing errors have been reduced from 12% to 0.3%, nearly eliminating this source of frustration and rework. All stakeholders report significantly improved satisfaction with the process, validating the comprehensive benefits of the digital transformation [43]. Beyond these immediate benefits, the system has created a foundation for further digital transformation initiatives within the college. Success has motivated administrators to consider similar approaches for other manual processes including internship management, thesis proposal submission, and academic misconduct reporting.

5.4. Future Work

Several promising directions for future research and system enhancement have been identified across different time horizons.

5.5. Short-Term Enhancements

Short-term enhancements focus on incremental improvements to existing functionality. Development of native mobile applications for iOS and Android would improve mobile user experience beyond the current responsive web design. Implementation of advanced analytics would provide dashboard displays showing historical trends, approval patterns, and predictive insights for better decision-making. Development of integration capabilities through APIs would enable connection with the university's official Student Information System for automatic

enrollment updates [44]. Addition of comprehensive Arabic language support would serve the local user base more effectively through localized interface and notifications. Enabling bulk operations would allow advisors to process multiple requests simultaneously for improved efficiency.

5.6. Medium-Term Research

Medium-term research directions explore more substantial innovations. Investigation of artificial intelligence integration could enable machine learning models to predict approval likelihood and recommend alternative courses for denied requests. Application of process mining techniques could identify optimization opportunities in the approval workflow through data-driven analysis [45]. Conducting comparative studies by deploying the system at other institutions would validate the generalizability of the approach across different contexts. Adapting architecture to support other academic processes such as grade appeals, course waivers, and independent study approvals would extend the system's value proposition.

5.7. Long-Term Vision

The long-term vision encompasses transformative possibilities for academic administration. Evolution into a comprehensive academic management platform would address multiple administrative processes with shared infrastructure, creating economies of scale and consistent user experience. Exploration of blockchain integration could enable immutable academic record keeping and automated smart contract-based approvals with enhanced security and transparency. Development of intelligent advising features powered by AI could proactively suggest courses based on degree requirements, prerequisites, and student performance patterns [46]. Creation of a cross-institutional framework would enable rapid customization and deployment across different institutions and academic contexts, maximizing the research impact.

5.8. Recommendations for Institutions

Based on this research, we offer several recommendations for institutions considering similar initiatives. First, institutions should start with pain points by focusing on solving specific, well-defined problems rather than attempting comprehensive system replacements that may overwhelm resources and stakeholder patience. Second, organizations should leverage existing infrastructure by evaluating existing cloud subscriptions such as Google Workspace or Microsoft 365 before investing in new platforms, potentially saving substantial costs. Third, projects must prioritize user experience by investing adequate time in interface design and usability testing, recognizing that technical sophistication means nothing if users find the system difficult to use. Fourth, institutions should plan for change management by allocating resources for training, support, and communication, understanding that technology alone does not ensure adoption. Fifth, teams should measure and iterate by establishing clear metrics for success and continuously refining based on actual usage data and feedback from real users. Sixth, projects must document thoroughly to facilitate maintenance, enhancement, and knowledge transfer both within the institution and to the broader research community.

5.9. Concluding Remarks

The Course Add/Drop Management System demonstrates that thoughtful application of modern web technologies can solve real administrative challenges in higher education effectively and affordably. While large-scale

commercial systems have their place, custom solutions built on cloud platforms offer compelling alternatives for focused applications, particularly for institutions with limited IT budgets. The success of this project illustrates a broader principle: digital transformation in higher education need not be an all-or-nothing proposition requiring massive investments. Incremental improvements addressing specific pain points can deliver substantial value while building organizational capability and confidence for larger initiatives. As universities worldwide continue to navigate the challenges of growing enrollments, resource constraints, and rising expectations for digital services, solutions like CADMS provide a roadmap for practical, cost-effective innovation.

By sharing detailed implementation experiences and rigorous evaluations, the research community can accelerate the adoption of effective practices and technologies across the higher education sector. The journey from paper forms and physical signatures to automated workflows and digital certificates represents more than technological modernization it reflects a fundamental rethinking of how academic institutions can better serve students while empowering administrators to focus on higher-value activities. This research contributes one data point in that ongoing transformation, offering both inspiration and practical guidance for institutions embarking on similar journeys.

5.10. Future Directions

Based on the successes and lessons learned from this research, we propose the following suggestions for future development and research:

1. **Cross-Institutional Collaboration Framework:** Develop a collaborative framework enabling multiple institutions to share and contribute enhancements to the system, pooling resources to create a community-maintained academic management platform that benefits the entire higher education sector.
2. **Artificial Intelligence Integration:** Explore machine learning algorithms to enhance decision support for advisors, including predictive models for approval likelihood, automatic course recommendations based on student performance patterns, and anomaly detection for identifying unusual requests requiring escalation.
3. **Advanced Analytics and Business Intelligence:** Implement sophisticated analytics capabilities providing administrators with comprehensive dashboards showing approval trends, bottleneck identification, predictive insights for capacity planning, and comparative benchmarking against peer institutions.
4. **Comprehensive Academic Process Suite:** Extend the platform to address other academic administrative processes such as grade appeals, course waivers, independent study approvals, internship management, and thesis proposal workflows, creating an integrated suite covering all student lifecycle management.
5. **Blockchain for Academic Records:** Investigate blockchain technology integration for creating immutable, verified academic records with decentralized storage, enabling secure credential sharing, and automating smart contract-based approvals with enhanced transparency and security guarantees.
6. **Comparative Implementation Studies:** Conduct deployment studies at diverse institutions spanning different sizes, geographical regions, and regulatory contexts to validate generalizability of the approach, identify context-specific adaptations, and document lessons learned across varied academic environments.

Declarations**Source of Funding**

This study received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Competing Interests Statement

The authors declare that they have no competing interests related to this work.

Consent for publication

The authors declare that they consented to the publication of this study.

Authors' contributions

All the authors made an equal contribution in the Conception and design of the work, Data collection, Drafting the article, and Critical revision of the article.

Availability of data and materials

Authors are willing to share data and material on request.

Ethical Approval

Not applicable for this study.

Institutional Review Board Statement

Not applicable for this study.

Informed Consent

Not applicable for this study.

Acknowledgement

Authors acknowledge the support and hard work from all those who helped in this study.

References

- [1] Ahmad, M., et al. (2025). Learning three-dimensional face recognition from sparse views for robust identity verification. SSRN. <https://ssrn.com/abstract=5428214>.
- [2] Akram, F., et al. (2024). Integrating artificial bee colony algorithms for deep learning model optimization: A comprehensive review. In *Solving with Bees: Transformative Applications of Artificial Bee Colony Algorithm*, Pages 73–102. https://doi.org/10.1007/978-981-97-7344-2_5.
- [3] Al Omari, O.M.A., Khan, N.A., & Mahafdah, R. (2017). Ranking and reputation based resource allocation in P2P system. *Mediterranean Journal of Basic and Applied Sciences*, 1(1): 293–301.
- [4] Alanezi, R., Alanezi, M.A., & Khan, N.A. (2018). Development of web based e-cooperative training system. In *2018 International Conference on Smart Computing and Electronic Enterprise (ICSCEE)*. <https://doi.org/10.1109/icscee.2018.8538367>.

- [5] Alangari, S., et al. (2022). Developing a blockchain-based digitally secured model for the educational sector in Saudi Arabia toward digital transformation. *PeerJ Computer Science*, 8: e1120. <https://doi.org/10.7717/peerj-cs.1120>.
- [6] Alangari, S., & Khan, N.A. (2021). Artificially intelligent warehouse management system. *Asian Journal of Basic Science & Research*, 3(3): 16–24. <https://doi.org/10.38177/ajbsr.2021.3302>.
- [7] Khan, N., et al. (2025). Network intrusion management of web form spamming using blockchain. *Irish Interdisciplinary Journal of Science & Research*, 9(3).
- [8] Khan, N.A. (2018). Cloud applications development and deployment: The future of cost-effective programming and a step ahead. *Middle East Journal of Applied Science & Technology*, 1(1): 30–36.
- [9] Khan, N.A. (2019). Security management protocols in cloud computation. *Middle East Journal of Applied Science & Technology*, 2(1): 16–23.
- [10] Khan, N.A., et al. (2019). Intrusion management to avoid web-form spamming in cloud based architectures. In 2019 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE). <https://doi.org/10.1109/iccike47802.2019.9004302>.
- [11] Khan, N.A., et al. (2019). Prevention of web-form spamming for cloud based applications: A proposed model. In Amity International Conference on Artificial Intelligence (AICAI).
- [12] Khan, N.A. (2019). Wireless requirements and benefits in the academics domain. *Middle East Journal of Applied Science & Technology*, 2(3): 45–49.
- [13] Khan, N.A. (2019). Basics of ethical hacking and computer security.
- [14] Aljomae, W.Y., Alshahrani, S.M., & Khan, N.A. (2025). NAMAQ-Arabic handwriting recognition using deep learning, AI, and ML with sentiment analysis. In 2025 4th International Conference on Computing and Information Technology (ICCIT). <https://doi.org/10.1109/iccit63348.2025.10989445>.
- [15] Khan, N.A., & Ghamdi, A. (2015). Cyber forensics and proposed techniques to overcome cyber threats for cyber security. *International Journal of Engineering and Management Research*, 5(5): 187–191.
- [16] Alshalaan, M., & Khan, N.A. (2025). Complexities and challenges for securing digital assets and infrastructure in academia: A review on digital asset security. In *Complexities and Challenges for Securing Digital Assets and Infrastructure*, Pages 225–244. <https://doi.org/10.4018/979-8-3373-1370-2.ch011>.
- [17] Almalki, J., et al. (2022). Enabling blockchain with IoMT devices for healthcare. *Information*, 13(10): 448. <https://doi.org/10.3390/info13100448>.
- [18] Almalki, J., Alshahrani, S.M., & Khan, N.A. (2024). A comprehensive secure system enabling healthcare 5.0 using federated learning, intrusion detection and blockchain. *PeerJ Computer Science*, 10: e1778. <https://doi.org/10.7717/peerj-cs.1778>.
- [19] Khan, N.A. (2019). Measuring academics intentions to use a project management system (PMS): A case study of the College of Computing and Information Technology, Shaqra University. *Trends in Future Informatics and Emerging Technologies*, Pages 58–69.

- [20] Khan, N.A. (2025). Statistical probability prediction model for e-learning and realtime proctoring using IoT devices. *Journal of King Saud University – Science*, 37: 7002025.
- [21] Alshahrani, S.M., et al. (2023). Systematic survey on big data analytics and artificial intelligence for COVID-19 containment. *Computer Systems Science & Engineering*, 47(2). <https://doi.org/10.32604/csse.2023.039648>.
- [22] Alshahrani, S.M., & Khan, N.A. (2023). COVID-19 advising application development for Apple devices (iOS). *PeerJ Computer Science*, 9: e1274. <https://doi.org/10.7717/peerj-cs.1274>.
- [23] Alshahrani, S.M., et al. (2022). URL phishing detection using particle swarm optimization and data mining. *Computers, Materials & Continua*, 73(3). <https://doi.org/10.32604/cmc.2022.030982>.
- [24] Alshalaan, M., & Khan, N.A. (2025). Blockchain-enabled federated learning framework for secure and collaborative drug discovery: Integrating AI, molecular docking, and distributed ledger technology. <https://doi.org/10.46759/ijjsr.2025.9401>.
- [25] Alsulami, M.H., Alotaibi, S., & Khan, N. (2021). Smart university model for Saudi Arabian universities. *Design Engineering*: 162–181.
- [26] Alsulami, M.H., et al. (2021). Zigbee technology to provide elderly people with well-being at home. *International Journal of Sensors Wireless Communications and Control*, 11(9): 921–927. <https://doi.org/10.2174/2210327911666210201105206>.
- [27] Hassan, M.A.A., Khan, N.A., & Nasim, M.A. (2017). Managing data replication in mobile ad-hoc network databases using content based energy optimization. *Mediterranean Journal of Basic and Applied Sciences*, 1(1): 142–154.
- [28] Khan, N.A. (2024). Development of intelligent pick and drop service manager for small cities. *Asian Journal of Basic Science & Research*, 6(3): 20–27. <https://doi.org/10.38177/ajbsr.2024.6303>.
- [29] Khan, N.A. (2024). Development of intelligent help system for small cities. *Asian Journal of Applied Science and Technology*, 8(3): 112–119. <https://doi.org/10.38177/ajast.2024.8311>.
- [30] Khan, N.A. (2025). Development of intelligent student information system. *Arabian Journal of Basic Science and Research*, 7(1). <https://doi.org/10.38177/ajbsr.2025.7101>.
- [31] Khan, N.A., et al. (2021). Development of Mubadarah system – An intelligent system for proposals at a university. In 2021 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE). <https://doi.org/10.1109/iccike51210.2021.9410773>.
- [32] Khan, N.A., et al. (2021). Development of Medidrone: A drone based emergency service system for Saudi Arabian healthcare. In 2021 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE). <https://doi.org/10.1109/iccike51210.2021.9410685>.
- [33] Khan, N.A., Rajeyyagari, S., & Khan, A.R. (2025). Development of intelligent library services for university students. *Mediterranean Journal of Basic and Applied Sciences*, 9(1): 142–147. <https://doi.org/10.46382/mjbas.2025.9109>.

- [34] Khan, N.A., Siddiqi, A.M.U., & Ahmad, M. (2021). Development of intelligent alumni management system for universities. *Asian Journal of Basic Science & Research*, 3(2): 51–60. <https://doi.org/10.38177/ajbsr.2021.3206>.
- [35] Khan, N.A., & Ahamad, D. (2025). Living smart: AI-based urban assistance systems for sustainable wellbeing in small cities. <https://doi.org/10.46431/mejast.2025.8210>.
- [36] Khan, N.A., et al. (2025). Transformative impact of artificial intelligence on higher education: A comprehensive analysis of pedagogical innovation, institutional transformation, and future learning ecosystems. *Asian Journal of Applied Science and Technology*, 9(4): 57–76.
- [37] Khan, N.A., et al. (2024). An IoMT enabled iterative artificial bee colony approach using federated learning for detection of heart disease. In *Solving with Bees: Transformative Applications of Artificial Bee Colony Algorithm*, Pages 103–116. https://doi.org/10.1007/978-981-97-7344-2_6.
- [38] Khan, N.A., et al. (2021). An empirical analysis on users' acceptance and usage of BYOD-technology for Saudi universities: A case study of Shaqra University. In *2021 International Conference on Technological Advancements and Innovations (ICTAI)*. <https://doi.org/10.1109/ictai53825.2021.9673287>.
- [39] Khan, N.A., Al-Omari, O.M., & Alshahrani, S.M. (2023). An empirical study on the future of publication repositories and its adaptability in public universities—A case study of Shaqra University, Saudi Arabia. In *Computational Intelligence: Select Proceedings of InCITe*, Pages 823–829.
- [40] Khan, N.A., & Albatein, J. (2021). COVIBOT – An intelligent WhatsApp based advising bot for COVID-19. In *2021 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)*. <https://doi.org/10.1109/iccike51210.2021.9410801>.
- [41] Khan, N.A., & Alshalaan, M. (2025). AI-driven blockchain framework for digital transformation of academic accreditation process: A Saudi Arabian perspective. <https://doi.org/10.46759/ijjsr.2025.9404>.
- [42] Khan, N.A., Khan, A.R., & Rajeyyagari, S. (2025). Innovation in teaching and learning with the use of modern computational tools: A post COVID experience. *Middle East Journal of Applied Science & Technology*, 8(2): 74–82. <https://doi.org/10.46431/mejast.2025.8208>.
- [43] Khan, V.N.A., et al. (2020). Internet of things (IoT) based educational data mining (EDM) system. *Journal of Mechanical Control & Mathematical Sciences*, 15(3): 271–284. <https://doi.org/10.26782/jmcms.2020.03.00022>.
- [44] Mahafdah, R.F., Al-Omari, O.M., & Khan, N.A. (2018). Learning modal adaptability to improve reading and writing skills of students.
- [45] Khan, N.A., et al. (2024). An IoMT enabled iterative artificial bee colony approach using federated learning for detection of heart disease. In *Solving with Bees: Transformative Applications of Artificial Bee Colony Algorithm*, Pages 103–116. https://doi.org/10.1007/978-981-97-7344-2_6.
- [46] Zamani, A.S., Akhtar, M.M., & Khan, N.A. (2025). An application of machine learning, big data and IoT of enterprise architecture: Challenges, solutions and open issues. <https://doi.org/10.5772/intechopen.1010260>.