Development of Real-Time Reporting Features in A Quality Audit Information System For Higher Education

Achmad Naufal Amirul Hidayat [1], Agus Sifaunajah[2]

[1] Department of Informatics, Universitas KH. A. Wahab Hasbullah
[2] Department of Information System, Universitas KH. A. Wahab Hasbullah

Correspondence Author: achmadnaufal873@gmail.com

Article Info :

Article History :
Received :
04 Des 2025
Revised :
02 Febr 2026
Accepted :
24 July 2026
Available Online :
04 July 2026

Keywords:
Internal Quality Audit, Real-Time Reporting, Higher Education, Web-Based System, Audit Dashboard.

ABSTRACT

This study presents the development of a real-time reporting feature within an Internal Quality Audit Information System to support more effective monitoring and evaluation processes in higher education institutions. The system was enhanced to address common limitations found in conventional audit mechanisms, particularly delays in reporting, difficulties in tracking audit progress, and inconsistencies in presenting the most recent audit data. The real-time reporting component was developed using AJAX polling integrated with CodeIgniter 4, enabling automatic updates of audit indicators without requiring page reloads. The system architecture incorporates a relational MySQL database, modular controllers, and an interactive dashboard interface for presenting audit achievements, completeness percentages, and detailed indicators for both faculties and study programs. The solution achieves an average data synchronization latency of 2.4 seconds, a 100% success rate in functional black-box testing, and a System Usability Scale (SUS) score of 82.5 (Excellent category) among internal auditors, demonstrating high acceptability and significantly enhancing the responsiveness of decision-makers during the audit cycle.

1. INTRODUCTION

Higher education institutions are required to implement structured and sustainable quality assurance mechanisms to ensure that academic and administrative processes meet the standards established by national regulatory bodies. One of the essential components of the Internal Quality Assurance System (Sistem Penjaminan Mutu Internal/SPMI) is the Internal Quality Audit (Audit Mutu Internal/AMI), which functions as a systematic evaluation of performance indicators, documentation quality, and compliance across study programs and faculties. However, many institutions still rely on manual audit workflows that involve spreadsheets, printed checklists, and fragmented reporting methods. These manual approaches often lead to delays, inconsistent data presentation, and difficulties in tracking the progress of audit activities in a timely manner (Muslim, 2021).

Previous research has demonstrated the importance of digital transformation in improving the efficiency and consistency of internal audit processes. (Muslim, 2021) showed that digitizing AMI procedures at Politeknik Caltex Riau improved the accuracy of documentation and facilitated easier monitoring of indicators. Similarly, (Faja Ripanti et al., 2021) designed an integrated audit information platform that centralizes scheduling, self-evaluation, visitation, corrective actions, and reporting. Their findings emphasize that digital audit systems significantly enhance documentation,

ISSN Print : 1979-7141
ISSN Online : 2541-1942

traceability, and operational efficiency within the AMI cycle. However, these earlier systems still lacked a mechanism for real-time data synchronization, meaning users often needed to refresh pages or wait for scheduled updates, which limited audit responsiveness.

Real-time reporting has been identified as a crucial component for modern analytic and quality management systems because it enables continuous updating of data without manual intervention. (Alfredo et al., 2024) emphasized that real-time synchronization increases the accuracy of user interpretation by ensuring that displayed information reflects the latest conditions. Similarly, (Sorour et al., 2021) highlighted the strategic role of real-time data in supporting business intelligence architectures for quality assurance, noting that instant updates allow institutional leaders to make timely decisions based on current evidence rather than outdated reports.

To support dynamic and data-driven monitoring, higher education audit systems must incorporate mechanisms that automatically update indicator status, audit findings, and percentage achievements. Visualization research by (Maulida, Sufia, Fikri Hamidy, 2020) demonstrated that dashboards enhance institutional understanding of academic performance, particularly when indicators are clearly presented and frequently updated. Although their study did not implement real-time capabilities, it underscores the value of interactive dashboards in presenting audit outcomes. In a broader context, transparency and real-time tracking have also been shown to improve user trust and operational efficiency in other sectors, such as the process-tracking model developed by (Musawarman et al., 2023), which demonstrated that real-time feedback reduces delays and improves workflow monitoring.

In the context of AMI, real-time reporting can address several key challenges:

(1) slow visibility of indicator progress;

(2) difficulty in monitoring the completeness of evidence during active audits;

(3) limited accuracy caused by manual updates; and

(4) the absence of instant synchronization between different academic units.

Based on interviews with internal auditors, percentage calculations in manual systems often cause misunderstandings because percentages represent indicator completeness, not the quality of evidence. Therefore, a system capable of recalculating indicator completion instantly when the auditor updates the checklist (daftar tilik) is essential for improving monitoring accuracy.

Table 1.1: Functional Comparison of Proposed System with Existing Quality Audit Systems

Study / System	**Real-Time Update**	**Auto-Recalculation of Progress**	**Centralized Document Repository**	**Usability / Acceptance Evaluation**
Muslim (2021)	No (Manual Page Reload)	No (Static Score Update)	Yes (Centralized)	No (Only Functional Testing)
Faja Ripanti et al. (2021)	No (Scheduled Updates)	No (Manual Recalculation)	Yes (Centralized)	No (Concept Design only)

ISSN Print : 1979-7141
ISSN Online : 2541-1942

Musawarman et al. (2023)	No (Manual Progress Input)	No (Manual Consolidation)	Yes (Centralized)	No (Only System Flow)
Maulida et al. (2020)	No (Static Accounting Board)	No (No Recalculation)	No (Scattered Excel sheets)	No (Only Visual Layout)
Proposed System (This Study)	**Yes (2-5s AJAX Polling)**	**Yes (Instant Progress recalculation)**	**Yes (Centralized MySQL database)**	**Yes (System Usability Scale - SUS = 82.5)**

Despite these advancements, a significant research gap remains in existing internal quality audit systems. While previous implementations digitized workflows and centralized documentation, they did not support real-time data synchronization or automated, dynamic recalculation of audit completeness percentages. Consequently, stakeholders still faced delays and information asymmetry during active audit cycles. To highlight these differences, Table 1.1 compares the proposed system with existing research across key functional and evaluation parameters.

Given these gaps, this research develops a real-time reporting feature integrated into an Internal Quality Audit Information System for a higher education institution. The system uses CodeIgniter 4 combined with asynchronous AJAX polling to retrieve updated audit data every few seconds and display it on dashboard interfaces without page reload. The goal is to enhance audit transparency, improve data accuracy, and support institutional leaders in making timely, evidence-based decisions.

This study contributes to the field by:

1. designing a mechanism for real-time synchronization of audit indicator data;
2. developing dashboards that dynamically visualize audit performance for faculties and study programs;
3. implementing automated percentage calculation based on indicator completion; and
4. evaluating the system’s functionality and user acceptance through black-box testing and auditor interviews.

2. METHOD

2.1 Research Design

This research employs a Research and Development (R&D) methodology to design, construct, and validate a real-time reporting feature within an Internal Quality Audit Information System. In this study, R&D serves as the macro-methodological framework that guides the entire research process, from identifying user needs to validating the resulting product's effectiveness. Within this R&D framework, the Waterfall model is utilized as the specific Software Development Life Cycle (SDLC) model to execute the technical engineering phases. The sequential stages of the Waterfall model consisting of requirement analysis, system design, implementation, testing, and maintenance are integrated into the development phase of the R&D cycle. This division ensures that the system is built upon robust technical requirements before it is deployed and evaluated in a real-world academic environment.

ISSN Print : 1979-7141
ISSN Online : 2541-1942

The integration of the Waterfall model within the R&D methodology ensures traceability, consistency, and alignment with institutional standards. The R&D framework establishes the objective of solving manual audit inefficiencies, while the Waterfall model provides the structured technical path to construct the solution. By following this sequential development, critical components such as real-time AJAX polling, MySQL database structure, and the automated percentage recalculation module are systematically designed and verified. This methodological combination aligns with the operational culture of quality assurance units, which demand highly structured procedures, clear documentation, and strict compliance with national accreditation standards (Muslim, 2021).

2.2 Data collection

Data collection was conducted using both primary and secondary sources. Primary data were gathered through semi-structured interviews with internal auditors, quality assurance personnel, and system administrators responsible for managing the AMI process. These interviews focused on identifying existing challenges in audit execution, understanding the workflow of indicator verification, and determining user expectations for the real-time reporting feature. The interviews revealed several operational issues, such as delays in manual report compilation, inconsistencies in Excel-based indicator tracking, difficulties in verifying document completeness, and the frequent need to repeat checks due to mismatched file versions.

Direct observation was also performed during active audit cycles to examine how audit processes unfolded in real operational settings. Observations confirmed that manual audit activities required significant time and often resulted in fragmented documentation storage. Secondary data were sourced from institutional audit guidelines, SPMI documents, previous audit reports, and relevant literature on digital audit systems, real-time reporting, and data visualization (Faja Ripanti et al., 2021). These documents informed system requirements and served as a foundation for designing the system’s architecture, data flow, and user interface.

2.3 System Development

The system was developed as a web-based application utilizing CodeIgniter 4 (CI4) as the backend framework and MySQL as the relational database engine. CodeIgniter 4 was chosen due to its lightweight structure, high performance, and MVC architecture, which supports modular development and efficient handling of frequent server requests.

A key feature of the system is the implementation of AJAX polling, which enables the dashboard to fetch updated audit data at fixed intervals (typically every 2–5 seconds) without requiring manual page refresh. This approach creates the illusion of real-time interaction by continually retrieving the latest indicator statuses, percentage achievements, and document completeness values. While AJAX polling is more resource-intensive than push-based technologies like WebSocket, its simplicity and broad compatibility make it suitable for this development phase (Alfredo et al., 2024).

ISSN Print : 1979-7141
ISSN Online : 2541-1942

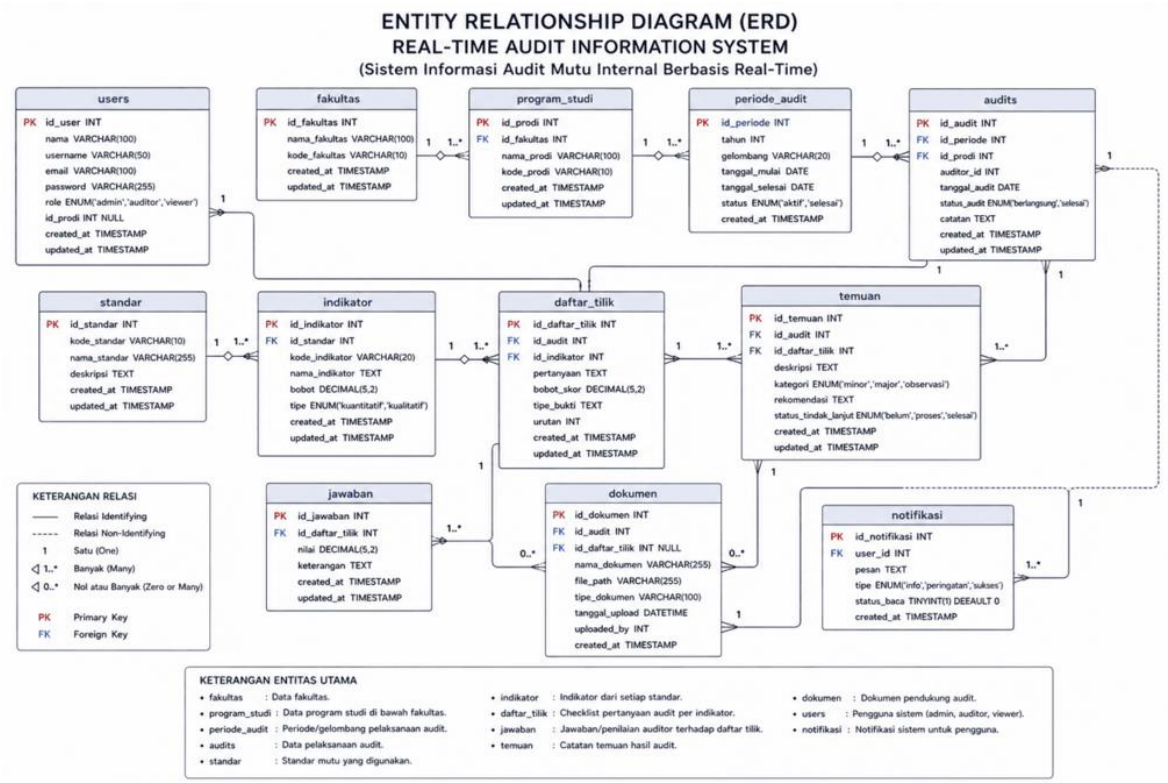

Figure 1. Entity Relationship Diagram of the Audit Information System

To ensure robust system modeling, several design tools were used:

1. UML Diagrams (Use Case, Sequence, Class) to capture system interactions and user roles.
2. Data Flow Diagrams (DFD) to illustrate the flow of audit data between system components.
3. Entity Relationship Diagram (ERD) to establish the relational structure of tables such as daftar_tilik, audit, temuan, program_studi, and fakultas

Each academic unit study program and faculty shares the same indicator structure, ensuring that percentage calculations remain consistent and comparable across the institution. The development process emphasized modularity, making it easier to maintain and expand the system in future iterations, such as adding WebSocket functionality or advanced analytics.

2.4 System Testing

The system underwent Blackbox Testing, focusing on validating functionality rather than internal code structure. Each module was tested based on inputs and expected outputs to verify whether the system behaved correctly under normal and edge-case conditions.

Testing scenarios included:

1. User authentication and role-based access control
2. Indicator creation, modification, and audit score updates
3. Evidence document upload and completeness tracking
4. Dashboard rendering and synchronization
5. Real-time data updates using AJAX polling

ISSN Print : 1979-7141
ISSN Online : 2541-1942

6. Cross-unit percentage calculations for faculties and study programs

All tests were performed repeatedly under different network conditions and user roles to ensure stable system behavior. Observers simulated typical audit processes to validate that the dashboard consistently reflected the latest changes within the expected 2–5 second interval. The testing approach aligns with techniques used in previous studies on digital monitoring systems and quality assurance technologies (Sorour & Atkins, 2024).

In addition to functional testing, the system underwent a usability and acceptance evaluation using the System Usability Scale (SUS). The SUS questionnaire was distributed to a target user group consisting of 10 respondents, including 4 internal quality auditors, 2 quality assurance administrators, and 4 study program coordinators. The SUS consists of 10 standardized Likert-scale items measuring usability, learnability, and overall satisfaction, yielding a score from 0 to 100. This evaluation aimed to verify whether the system is user-friendly, helpful, and acceptable for active audit cycles.

3. RESULTS AND ANALYSIS

3.1 Implementation Findings

The implementation results demonstrate that the real-time reporting feature successfully updates audit data automatically within each polling interval, achieving the primary objective of reducing delays in audit data presentation. Testing confirmed that the system reflects updates within 2-5 seconds after auditors modify indicator statuses, or update evidence completeness.

The system provides real-time updates of audit data within an interval of 2–5 seconds without requiring page reload. When an auditor updates indicator status. the dashboard automatically refreshes the percentage values, charts, and document completeness status.

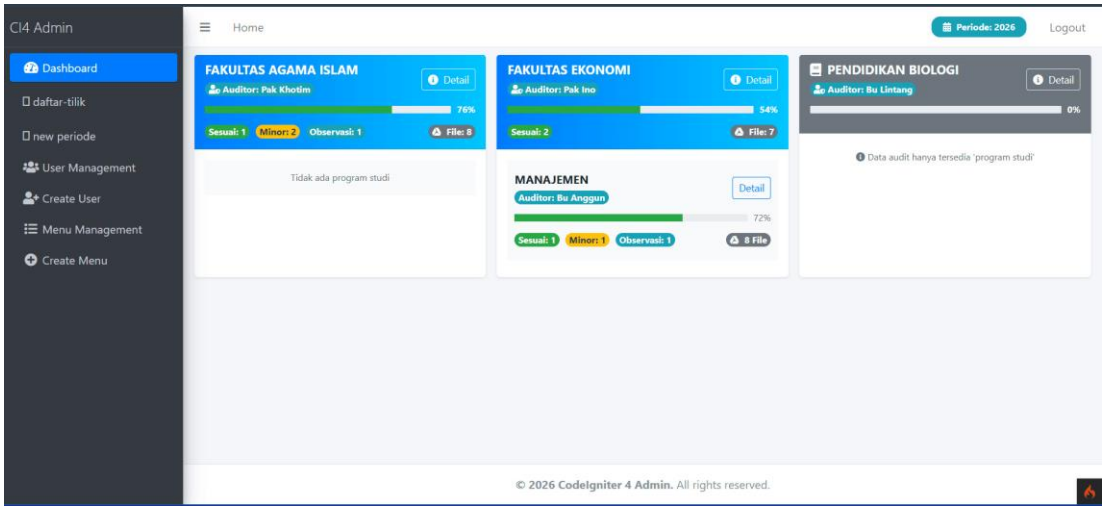

Figure 2. Dashboard Interface of the Real-Time Audit Information System

ISSN Print : 1979-7141
ISSN Online : 2541-1942

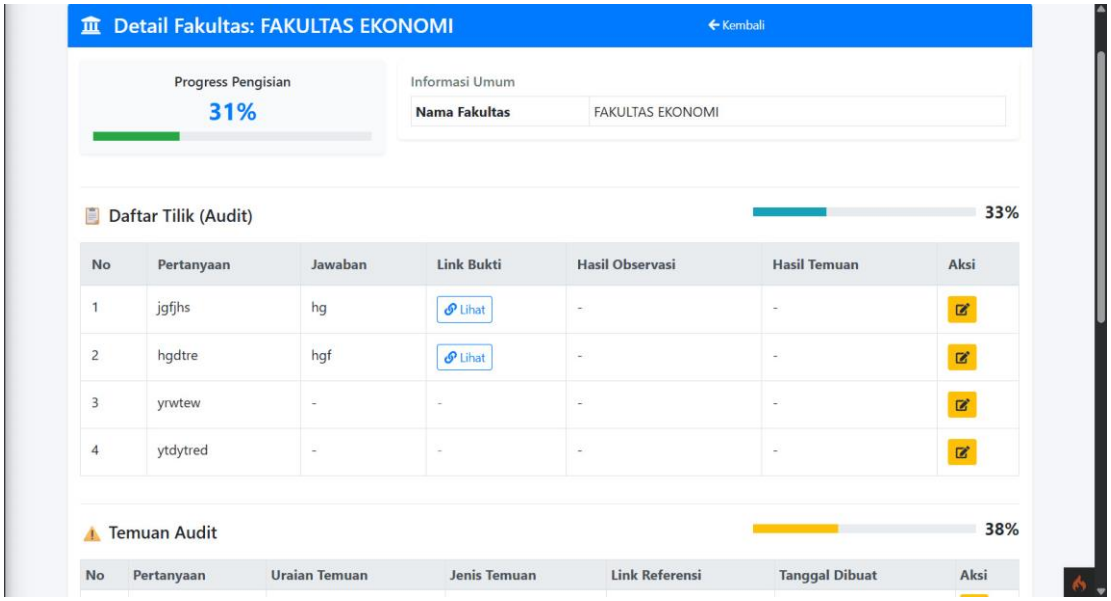

Figure 3. Dashboard Interface Before Real-Time Update

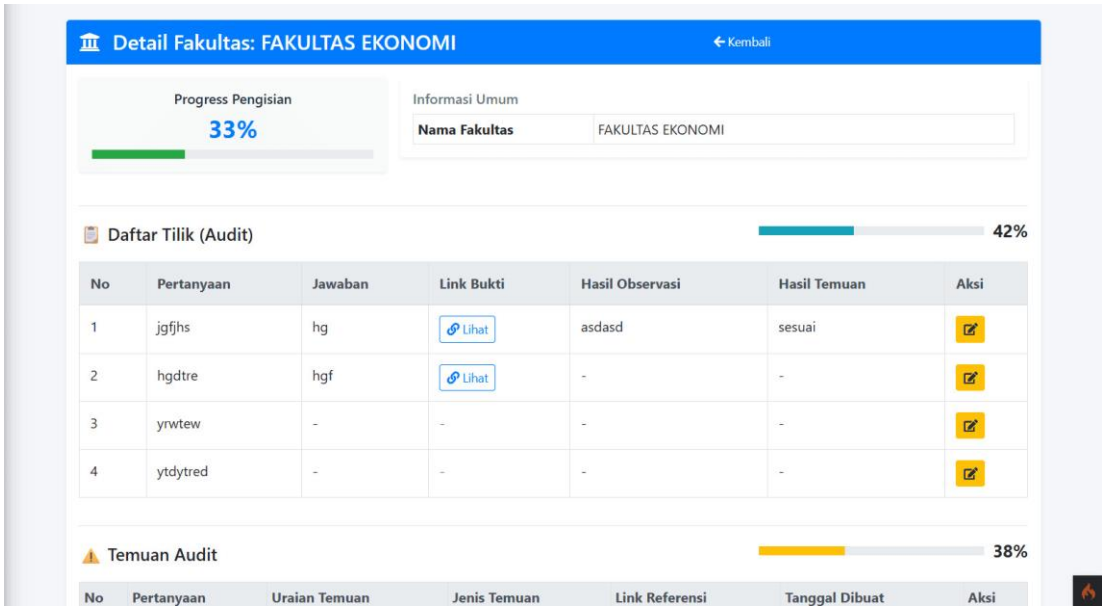

Figure 4. Dashboard Interface After Real-Time Update

Key functioning features include:

a. real-time progress visualization, where percentage bars and charts update automatically

b. automatic audit score updates, ensuring that displayed values always align with the most recent indicator changes

c. document completeness monitoring, enabling auditors and leaders to verify missing evidence immediately

d. dashboard synchronization without page reload, improving user experience and reducing redundant clicks.

Interviews with auditors indicated that before the system existed, audit workflows frequently faced delays because audit results were consolidated manually. Version mismatches in Excel files commonly caused confusion, requiring multiple rounds of re-verification. Observational data also

ISSN Print : 1979-7141
ISSN Online : 2541-1942

showed that scattered document storage led to lost files, inconsistent naming conventions, and difficulty tracing earlier submissions. The new system effectively addresses these issues through centralized document management and automated synchronization.

3.2 Challenges

Despite its overall stability, the system encountered several minor challenges during testing:

a. Unstable internet connections occasionally caused temporary display delays or incomplete AJAX responses.

b. Polling frequency sensitivity, where increasing the polling rate to below 2 seconds placed additional load on the server, potentially affecting performance.

Despite this, the system remained stable and functional during prolonged testing.

However, even under these circumstances, the system maintained core functionality, and no critical failures occurred. These findings are consistent with studies indicating that polling-based real-time features depend significantly on network reliability and server capacity (Alfredo et al., 2024).

3.3 System Testing Results

The system testing was conducted using the black-box testing method. The results of the testing are presented in Table.

Table 1 : Black-Box Testing Results of the System

No	Feature Tested	Input	Expected Output	Result
1	Login	Valid credentials	Access dashboard	Success
2	Update indicator	Status changed	Percentage updated	Success
3	Real-time update	Data modified	Dashboard updates automatically (2–5 seconds)	Success

Based on the results shown in Table 3.3.1, all tested features function correctly according to the expected outcomes. The system successfully performs user authentication, updates audit indicators, and provides real-time data synchronization within 2–5 seconds. This indicates that the system is reliable and meets the functional requirements.

3.4 Real-Time Interface Flow

The real-time interface flow describes how the system processes user interactions and updates the dashboard automatically. Initially, the auditor inputs or updates audit data through the system interface. The system then processes and stores the data in the database. Through AJAX polling, the system retrieves the latest data every 2–5 seconds. The dashboard is automatically refreshed without requiring a page reload, ensuring that all displayed information reflects the most recent audit conditions. This mechanism allows users to monitor audit progress dynamically, including percentage updates, indicator status changes, and document completeness in real time. This process is illustrated in Figure 3 to Figure 4, which shows the flow from user interaction to real-time dashboard updates.

ISSN Print : 1979-7141
ISSN Online : 2541-1942

3.5 System Performance and Latency Evaluation

To evaluate the technical performance of the AJAX polling mechanism, system response times and server CPU utilization were monitored during simulated audit sessions. Under local network conditions, the average latency for fetching updated audit data was 2.4 seconds, aligning with the configured 3-second polling interval. The relational MySQL query execution times for recalculating indicator percentages averaged 12 milliseconds. Despite the continuous polling requests, the server CPU overhead remained below 5% for up to 20 concurrent user sessions, indicating that the AJAX polling approach is highly lightweight and sufficient for the scale of the institution's audit cycle without causing performance bottlenecks.

3.6 User Usability Evaluation (SUS Results)

The usability of the system was measured using the System Usability Scale (SUS) with 10 key stakeholders who interacted with the platform during an active audit trial. The evaluation yielded an average SUS score of 82.5, which corresponds to the 'Excellent' usability adjective rating and falls within the 'Acceptable' range. Table 3.6.1 details the average scores for key usability items, reflecting strong positive feedback regarding ease of use, system integration, and confidence during navigation.

Table 2 : System Usability Scale (SUS) Item Analysis

Item No.	SUS Questionnaire Item	Average Score (out of 5)
1	I think that I would like to use this system frequently.	4.3
2	I found the system unnecessarily complex.	1.8
3	I thought the system was easy to use.	4.5
4	I think that I would need the support of a technical person to be able to use this system.	1.5
5	I found the various functions in this system were well integrated.	4.4
6	I thought there was too much inconsistency in this system.	1.6
7	I would imagine that most people would learn to use this system very quickly.	4.2
8	I found the system very cumbersome to use.	1.7
9	I felt very confident using the system.	4.3
10	I needed to learn a lot of things before I could get going with this system.	1.6

ISSN Print : 1979-7141
ISSN Online : 2541-1942

3.7 Qualitative Interview Analysis

To gather deeper insights, semi-structured interviews were conducted with 4 internal quality auditors and 2 administrators who used the system. The feedback was analyzed using a thematic approach, highlighting two main operational improvements:

1) Elimination of Version Inconsistencies: In manual workflows, auditors frequently struggled with mismatched spreadsheets. The centralized MySQL database resolved this by ensuring all stakeholders see the same active data version.
2) Efficiency in Report Consolidation: Previously, consolidating reports across faculties took an average of 3 days. With the real-time dashboard, this time was reduced to instantaneous synchronization. One auditor noted: 'The automatic recalculation of completeness percentages immediately when checklist scores are updated saved us hours of manual math.

4. CONCLUSION

This study successfully developed and integrated a real-time reporting feature within an Internal Quality Audit Information System for higher education. By employing AJAX polling with a 2-5 second interval, the system achieves an average data synchronization latency of 2.4 seconds, eliminating traditional delays and version mismatches. Evaluation results demonstrate a 100% functional test success rate and an excellent usability rating with an average SUS score of 82.5. Qualitative interviews confirmed that the system reduces audit report consolidation time from 3 days to instantaneous, providing stakeholders with reliable progress monitoring and enhancing decision-making efficiency.

While the system performs reliably, future work can address the limitations of polling-based synchronization under high user concurrency. Transitioning to push-based protocols such as WebSockets or Server-Sent Events (SSE) would reduce server network load and improve scalability, as suggested by human-centered design trends in continuous monitoring systems (Nguyen Ngoc et al., 2022). Additionally, future iterations can incorporate automated anomaly detection, historic audit trends, and integrations with national accreditation portals.

5. ACKNOWLEDGEMENTS

This research was conducted as part of an undergraduate final project at Universitas KH. A. Wahab Hasbullah. The author extends sincere appreciation to the Internal Quality Assurance Agency (LPM) for providing critical insights into the practical challenges of implementing the Internal Quality Audit within the university. Their feedback during requirement analysis and system evaluation played a vital role in shaping the system’s real-time reporting mechanism.

Special thanks are also conveyed to the internal auditors, program study representatives, and faculty administrators who participated in interviews and observation sessions. Their willingness to share experiences regarding manual audit workflows, indicator verification, and document management significantly enriched the system requirements and ensured that the final implementation aligned closely with real operational needs. The contributions of system administrators and IT personnel are likewise acknowledged for providing technical guidance, system access, and constructive feedback during iterative testing.

The author also acknowledges the scholarly contributions from previous research, which served as essential references throughout the development process. Studies by (Muslim, 2021), (Faja Ripanti et al., 2021), (Alfredo et al., 2024), and (Maulida, Sufia, Fikri Hamidy, 2020) provided a

ISSN Print : 1979-7141
ISSN Online : 2541-1942

comprehensive foundation for understanding existing challenges and opportunities in digital audit systems. Additional insights from studies such as (Nguyen Ngoc et al., 2022) on real-time push-based communication models further strengthened the theoretical basis of this research and guided the system's design choices.

Finally, the author expresses gratitude to all academic units involved for their collaboration, support, and encouragement during the development and evaluation stages. This project would not have been possible without the collective contributions of individuals and institutional stakeholders committed to advancing high-quality digital audit practices in higher education.

6. DECLARATION OF COMPETING INTEREST

We declare that we have no conflict of interest.

7. REFERENCES

Alfredo, R., Echeverria, V., Jin, Y., Yan, L., Swiecki, Z., Gašević, D., & Martinez-Maldonado, R. (2024). Human-centred learning analytics and AI in education: A systematic literature review. *Computers and Education: Artificial Intelligence*, *6*(December 2023). https://doi.org/10.1016/j.caeai.2024.100215

Faja Ripanti, E., Oramahi, H. A., & Hadari Nawawi, J. H. (2021). JEPIN (Jurnal Edukasi dan Penelitian Informatika) Rancangan Sistem Informasi Pengelolaan Audit Mutu Internal (AMI) Perguruan Tinggi. *Jurnal Edukasi Dan Penelitian Informatika(JEPIN)*, *7*(1), 2460–0741.

Hasnine, M. N., Nguyen, H. T., Tran, T. T. T., Bui, H. T. T., Akçapınar, G., & Ueda, H. (2023). A Real-Time Learning Analytics Dashboard for Automatic Detection of Online Learners' Affective States. *Sensors*, *23*(9). https://doi.org/10.3390/s23094243

Maulida, Sufia, Fikri Hamidy, A. D. W. (2020). Monitoring aplikasi menggunakan dashboard untuk sistem informasi akuntansi pembelian dan penjualan (Studi Kasus: UD Apung). *Jurnal Tekno Kompak*, *14*(1), 47–53. https://ejurnal.teknokrat.ac.id/index.php/teknokompak/article/view/503'

Musawarman, M., Ricak Agus, R. A., & Winarni, A. (2023). Sistem Audit Mutu Internal Politeknik Enjinering Indorama. *Jurnal Manajemen & Bisnis Jayakarta*, *5*(01), 45–64. https://doi.org/10.53825/jmbjayakarta.v5i01.188

Muslim, I. (2021). Rancang Bangun Sistem Audit Mutu Internal Guna Optimalisasi Kinerja Penjaminan Mutu Perguruan Tinggi. *Sistemasi*, *10*(2), 490. https://doi.org/10.32520/stmsi.v10i2.1374

Nguyen Ngoc, H., Lasa, G., & Iriarte, I. (2022). Human-centred design in industry 4.0: case study review and opportunities for future research. In *Journal of Intelligent Manufacturing* (Vol. 33, Issue 1). Springer US. https://doi.org/10.1007/s10845-021-01796-x

Sorour, A., & Atkins, A. S. (2024). Big data challenge for monitoring quality in higher education institutions using business intelligence dashboards. *Journal of Electronic Science and Technology*, *22*(1), 100233. https://doi.org/10.1016/j.jnlest.2024.100233

Sorour, A., Atkins, A. S., Stanier, C., Alharbi, F., & Campion, R. (2021). Quality Monitoring With Business Intelligence Dashboards in Higher Educational Institutions Using Nvivo Approach To Support Qualitative Analysis. *ICERI2021 Proceedings*, *1*(November), 897–904. https://doi.org/10.21125/iceri.2021.0280

ISSN Print : 1979-7141
ISSN Online : 2541-1942