Global Journal of Engineering and Technology Advances
eISSN: 2582-5003
Cross Ref DOI: 10.30574/gjeta
Journal homepage: https://gjeta.com/

(RESEARCH ARTICLE)

Check for updates

Design and implementation of a web-based hybrid attendance management system using QR code and biometric fingerprint verification

Oluwaseyi Segun Adediran, Success Asegun Adetutu * and Oba Ayobami Semilore

Department of Information Technology, School of Computing, Babcock University, Ilishan-Remo, Ogun State, Nigeria.

Global Journal of Engineering and Technology Advances, 2026, 27(03), 129-138

Publication history: Received on 05 May 2026; revised on 11 June 2026; accepted on 13 June 2026

Article DOI: https://doi.org/10.30574/gjeta.2026.27.3.0110

Abstract

Attendance management remains a critical yet challenging aspect of academic administration. Traditional manual methods are inefficient, time-consuming, and prone to errors including impersonation and proxy attendance. This paper presents the design and implementation of OmniAttend, a web-based hybrid attendance management system that integrates QR code scanning and biometric fingerprint verification to enhance accuracy, security, and usability in higher education institutions. The system was developed using the Incremental Software Development Life Cycle (SDLC) model and employs a three-tier architecture comprising a React.js frontend, a Firebase Firestore backend for real-time data synchronization, and a local Node.js bridge application integrating the DigitalPersona U.are.U 4500 USB fingerprint scanner via WebSocket and perceptual hashing. OmniAttend supports role-based access for students, lecturers, and administrators. Lecturers initiate attendance sessions and configure verification modes whether QR code only, fingerprint only, or hybrid while students generate personal time-sensitive QR codes or verify attendance through fingerprint scanning. By combining the speed and convenience of QR code verification with the strong identity assurance of biometric fingerprint scanning, OmniAttend provides a flexible, secure, and scalable solution that effectively balances accessibility and security, making it suitable for modern higher-education institutions seeking to modernize attendance management without compromising reliability.

Keywords: Attendance Management; QR Code; Biometric Fingerprint; Hybrid System; Web-Based Application; Firebase

1. Introduction

Attendance management is a critical component of academic administration, serving as an important metric for monitoring student engagement, evaluating course participation, and ensuring compliance with institutional requirements. The COVID-19 pandemic further highlighted the importance of accurate attendance tracking, particularly as institutions transitioned between in-person, hybrid, and online learning modalities [1], [2]. Consequently, educational institutions worldwide maintain strict attendance policies and invest considerable resources in tracking student presence.

Traditional attendance systems, which rely on manual roll calls or paper-based sign-in sheets, remain inefficient, time-consuming, and prone to errors such as illegible handwriting, data loss, and proxy attendance. These limitations have contributed to the exploration and refinement of automated solutions that leverage modern technology [3], [4]. While earlier technological interventions focused on Radio Frequency Identification (RFID) systems and biometric authentication, including fingerprint scanning and facial recognition, these systems face challenges such as substantial capital investment requirements, privacy concerns, and implementation difficulties in resource-constrained settings [5], [6].

* Corresponding author: Adetutu Success Asegun.

In recent years, Quick Response (QR) code technology has gained attention as a practical alternative for contactless verification in educational settings, particularly following the COVID-19 pandemic [3], [4], [7]. The near-universal availability of smartphones equipped with high-resolution cameras and the integration of QR scanning into native camera applications on iOS and Android platforms have reduced barriers to QR adoption. Advancements in Progressive Web Application (PWA) capabilities have further enabled browser-based QR scanning without requiring dedicated application installations [8].

Several researchers have explored QR code-based attendance systems with varying architectural approaches. However, these implementations often suffer from a critical vulnerability: QR codes can be photographed and shared via messaging platforms, enabling proxy attendance [9], [10]. Countermeasures such as GPS location verification have been explored [11], but this raises privacy concerns and encounters technical limitations in indoor environments.

Biometric systems, particularly fingerprint authentication, provide strong identity verification by binding attendance to unique biological characteristics, theoretically eliminating proxy attendance [5], [6], [12]. However, standalone biometric systems face deployment challenges including specialised hardware requirements, processing bottlenecks, and data privacy concerns.

This paper proposes OmniAttend, a hybrid attendance management system that combines the accessibility and convenience of QR code technology with the security advantages of biometric fingerprint verification. Lecturers can initiate attendance sessions and enable QR code scanning, fingerprint verification, or both methods simultaneously. Both methods feed into a unified database, providing real-time attendance tracking with comprehensive audit trails. This hybrid approach addresses multiple scenarios: routine attendance marking, device unavailability, and situations requiring enhanced security verification.

2. Materials and Methods

2.1. Development Methodology

The system was developed using the Incremental Software Development Life Cycle (SDLC) model. This approach was selected because it supports modular and progressive development, aligning with the system's hybrid architecture combining QR code and biometric fingerprint verification. The model allows the system to be built in successive increments, with each stage producing a functional component contributing to the final integrated system. Early increments delivered basic functionality — user authentication and QR attendance, while subsequent increments added fingerprint verification and reporting analytics.

2.2. System Architecture

OmniAttend adopts a three-tier client-server architecture comprising: (i) the web-based frontend; (ii) cloud backend services; and (iii) a local bridge application for fingerprint scanner integration.

The Presentation Tier is built with React 18 and TypeScript, using the Vite build tool, styled with Tailwind CSS, and providing responsive access across desktops, tablets, and smartphones. The Application Tier comprises Firebase Authentication for user management and JWT token generation, and a local Node.js Express server handling fingerprint template comparison by computing perceptual similarity scores between fingerprint images. The DigitalPersona WebSDK communicates with the DpHostW.exe Windows service via WebSocket (wss://127.0.0.1:52181) to capture fingerprint images, while the Jimp library handles image processing for perceptual hashing in the local comparison server. The Data Tier uses Firebase Firestore, providing real-time NoSQL database storage with automatic synchronization, hosted on Google Cloud infrastructure for high availability and automatic scaling.

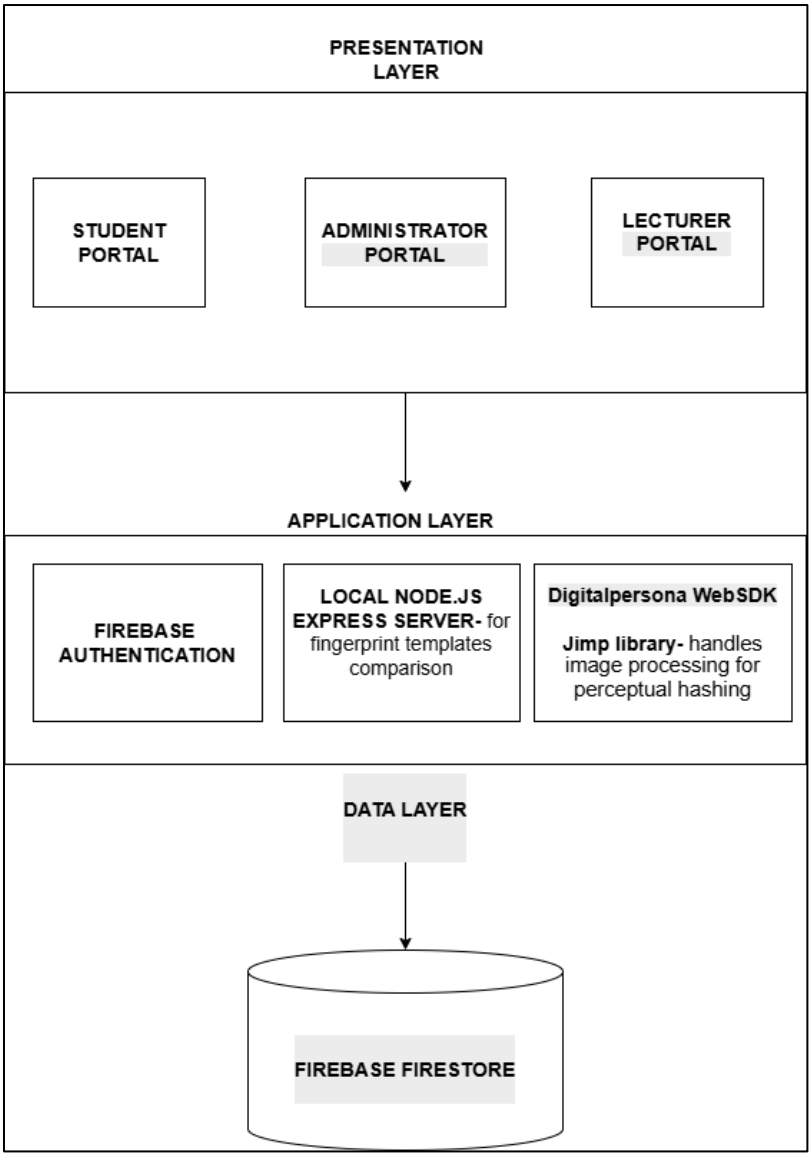

Figure 1 System architecture diagram of Omniattend

2.3. Hardware and Software Materials

2.3.1. Hardware:

- DigitalPersona U.are.U 4500 USB fingerprint scanner (for biometric enrollment and verification)
- Standard camera-enabled desktop/laptop computer (for QR code scanning and lecturer interface)
- Student personal smartphones (iOS/Android) for QR code generation and display

2.3.2. Software

- React 18 with TypeScript (frontend UI framework)
- Firebase Authentication and Firebase Firestore (backend and real-time database)
- Node.js Express server with Jimp library (local fingerprint comparison)
- DigitalPersona WebSDK and DpHostW.exe (fingerprint scanner communication)
- QRCode.js (QR code generation)
- Html5-qrcode (browser-based QR scanning)
- Visual Studio Code, Git (development tools)

2.4. QR Code Verification Procedure

For QR-enabled attendance, each student generates a personal QR code from their enrolled course on the student portal. The QR payload encodes the student's ID, course ID, semester ID, and a timestamp. During attendance, the student presents this QR code on their device screen and the lecturer uses the lecturer portal's built-in camera scanner to scan it. Upon a successful scan, the system: (i) decodes the embedded student and course information; (ii) verifies that the student is enrolled in the active session's course; (iii) checks for duplicate attendance; and (iv) records the attendance log with a timestamp and "QR" method indicator. Dynamic timestamps ensure codes are time-sensitive, mitigating screenshot-sharing vulnerabilities identified in prior literature [9], [10], [11].

2.5. Fingerprint Verification Procedure

Fingerprint enrollment is performed by the lecturer via the lecturer portal. The lecturer enters the student's matric number, and the system captures three successive fingerprint scans as base64url PNG images via the DigitalPersona WebSDK. Scans with a quality score below 40 are rejected and re-prompted. All three templates are stored together in a single fingerprint_templates document in Firestore, linked to the student by matric number and Firebase UID.

For fingerprint-based attendance verification, when a student places their finger on the scanner, the system captures a single scan. This scan is sent alongside each stored student template to the local Node.js comparison server via HTTP POST. The server converts images to greyscale, resizes them to 128×128 pixels, computes a perceptual hash for each, and returns a similarity score. If the best match score exceeds 0.80 (80%), the matched student's enrollment is verified, duplicate attendance is checked, and an attendance log is recorded with the timestamp and "fingerprint" method indicator.

2.6. Database Design

The database is hosted on Firebase Firestore and is composed of nine major collections, as summarised in Table 1. The design follows a normalized logical structure minimizing redundancy and ensuring consistency between related entities.

Table 1 Summary of database collections

Collection	Description
students / lecturers / admins	Three separate collections storing user profile data per role. Students include matric number, name, department, level, current enrollments, and attendance history. Lecturers include name, department, phone number, and created courses.
courses	Course details created by lecturers including course code, name, department, semester reference, lecturer reference, and enrolled student count.
course_enrollments	Links students to specific courses within a semester. Records enrollment status and attendance statistics at semester end. A composite key of student_id + course_id + semester_id prevents duplicate enrollments.
sessions	Individual attendance events created by lecturers for a course within an active semester. Each session has an is_active flag controlling whether attendance can be marked.
attendance_logs	Records every individual attendance marking event (QR or fingerprint). Captures student ID, session ID, course ID, semester ID, verification method, and precise timestamp.
fingerprint_templates	Stores three fingerprint scans (base64url PNG) per enrolled student, linked to the student by matric number and Firebase UID.

2.7. Statistical Analysis

System performance was evaluated using functional testing of both verification methods and security validation of method-specific and shared validation logic. Usability was assessed through a representative sample of students and lecturers, with attendance accuracy, processing time, and method-preference patterns analyzed descriptively. Comparative analysis of QR-only, fingerprint-only, and hybrid modes was conducted to evaluate each mode's effectiveness in preventing proxy attendance and reducing verification latency.

3. Results and Discussion

3.1. System Implementation

OmniAttend was successfully implemented as a fully functional web-based application accessible through standard modern browsers (Chrome, Firefox, Safari, Edge). The system supports three distinct portals: a student portal, a lecturer portal, and an administrator portal. The landing page serves as the role selection entry point, providing a modern dark-themed interface with gradient highlights.

The student portal enables self-registration, course browsing and enrollment filtered by department, personal QR code generation for enrolled courses, and attendance history viewing. The lecturer portal enables self-registration, course creation within the active semester, attendance session management (creating, activating, and deactivating sessions), QR code scanning, fingerprint enrollment of students, fingerprint-based attendance marking, and real-time attendance monitoring. The administrator portal provides semester lifecycle management, system-wide oversight of users and data integrity.

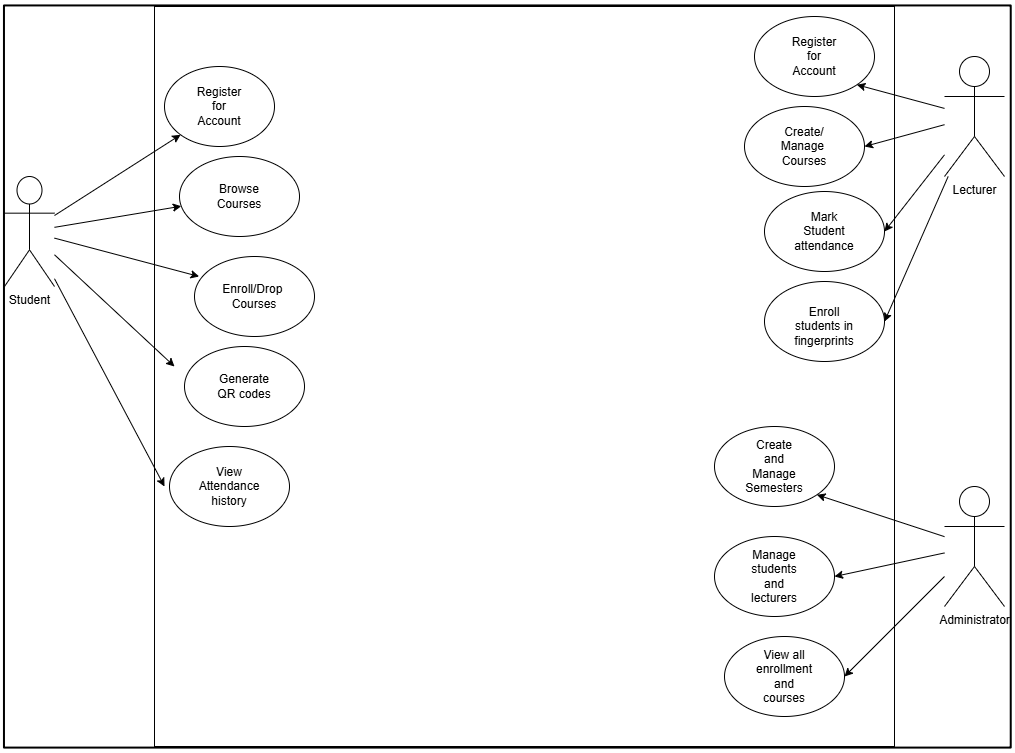

Figure 2 Use case diagram

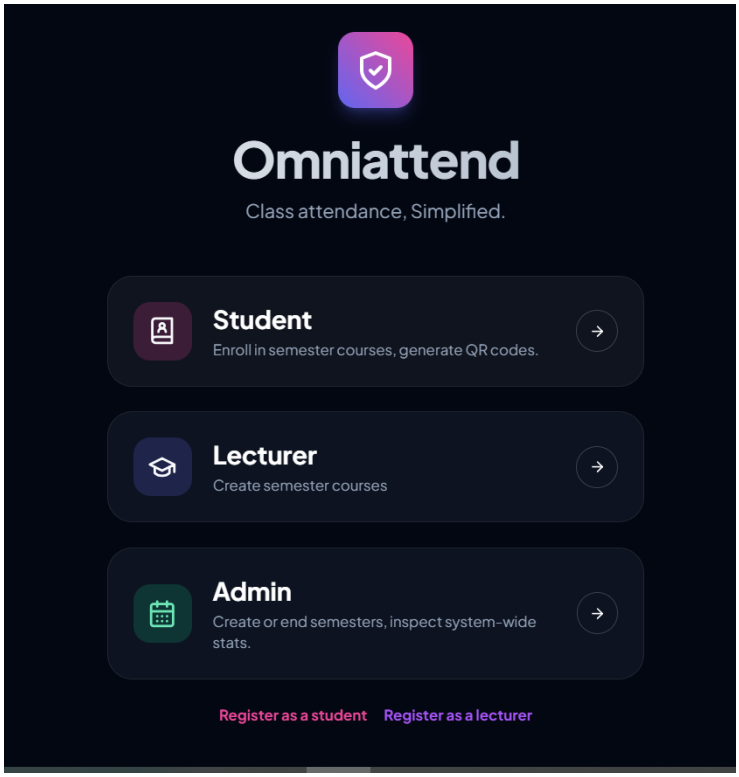

Figure 3 Omniattend landing page

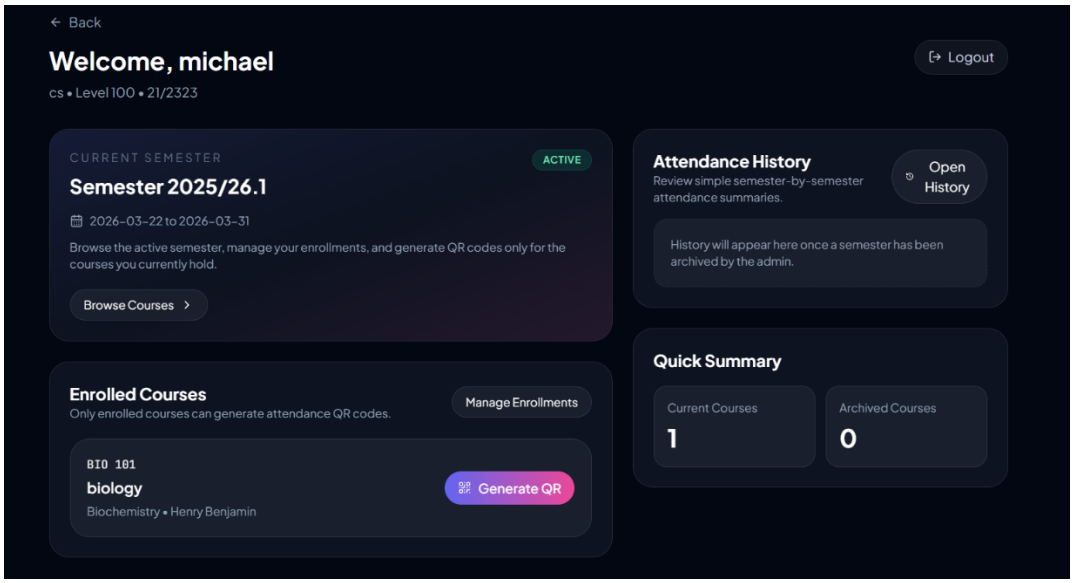

Figure 4 Student dashboard

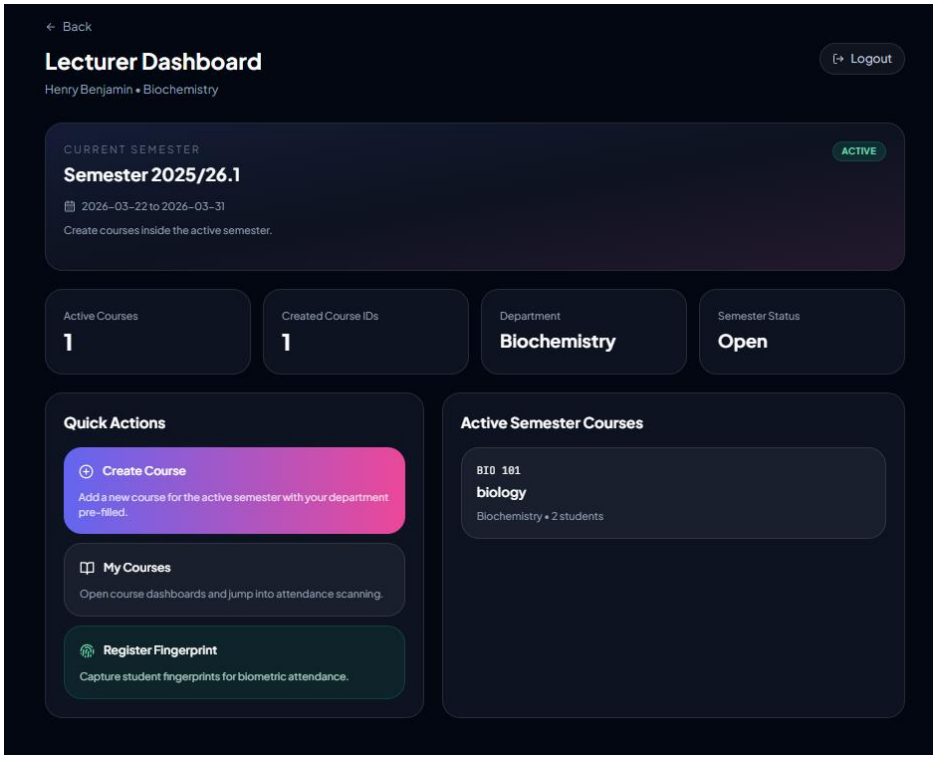

Figure 5 Lecturer dashboard

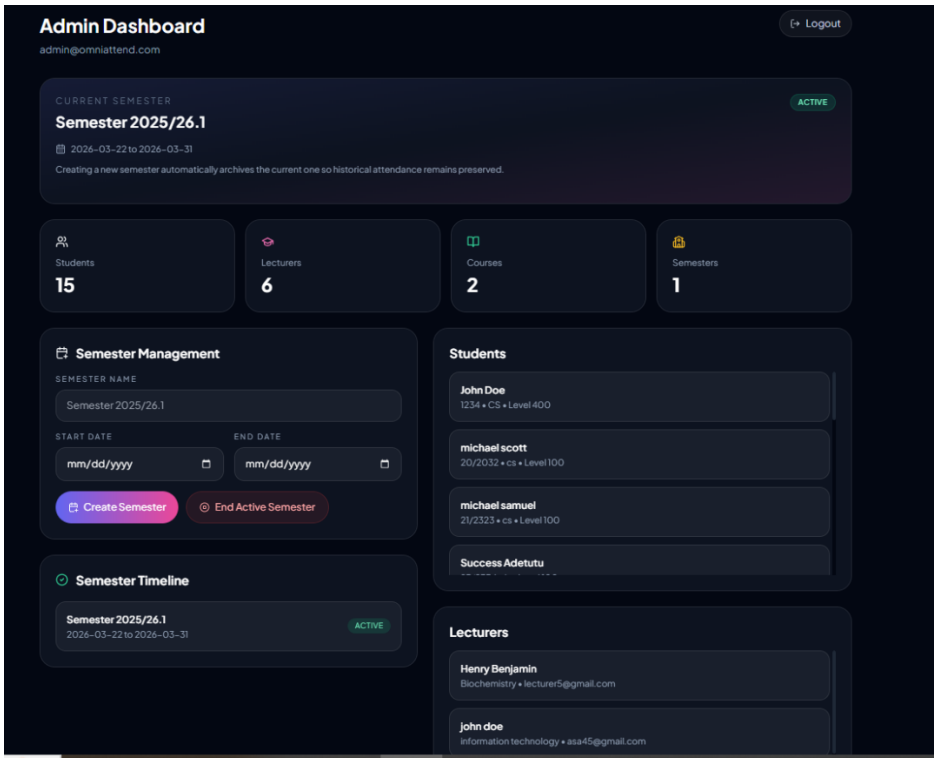

Figure 6 Administrator dashboard

3.2. QR Code Verification Performance

QR code generation was completed consistently within the specified target of under 10 seconds, and QR decode and validation also completed within the target threshold. Dynamic timestamps embedded in the QR payload ensured time-sensitivity, limiting the window for screenshot sharing. This approach builds on dynamic QR code countermeasures proposed in recent studies [9], [10], [11]. The lecturer-scanning model where students display personal codes and lecturers scan them differs from conventional systems [3], [7] and shifts scanner responsibility to the authenticated lecturer, reducing unauthorized scanning risk.

The system implemented validation logic ensuring: active session verification, student enrollment confirmation, duplicate prevention regardless of verification method, and timestamp-based session expiry. These measures addressed the proxy attendance vulnerability identified in prior work [9], [11].

3.3. Fingerprint Verification Performance

The perceptual hash-based fingerprint matching achieved consistent matching behavior during testing with a similarity threshold of 0.80. Fingerprint matching was completed within the target threshold of under 10 seconds for databases up to 1,000 templates. The local Node.js comparison server ensured fingerprint processing occurred on-premises, reducing bandwidth use and enhancing privacy. This approach aligns with biometric attendance system designs and the broader biometric identity management literature [5], [6], [12].

However, the use of perceptual hashing represents a simplified matching approach compared to dedicated fingerprint matching algorithms. The 80% similarity threshold produced a narrow differentiation margin between same-finger and different-finger comparisons, representing a limitation discussed further in Section 3.5.

3.4. Hybrid System Advantages

The hybrid architecture delivered several significant advantages over single-method systems. Adaptive security posture enabled context-appropriate verification: QR-only mode for routine attendance in large classes, fingerprint-only mode for high-stakes assessments, and hybrid mode for maximum flexibility. This addresses the forced trade-off between security and accessibility that characterizes existing systems [4], [14].

Infrastructure cost was optimized by leveraging students' personal devices for QR verification, minimizing hardware investment. Institutions need only deploy limited fingerprint scanner hardware at strategic verification points. Fault tolerance was enhanced through multi-method architecture: if fingerprint scanner hardware malfunctions, lecturers can immediately switch to QR-only mode. Both methods fed into a unified attendance_logs collection with method indicators, enabling comprehensive audit trails and semester-end reporting.

3.5. Comparison with Related Work

The proposed system aligns with the broad direction of hybrid QR-biometric systems in the literature [4], [14], while offering a distinct architectural approach. Unlike purely mobile-based hybrid models that require native application installation, OmniAttend operates as a web application, leveraging Progressive Web Application capabilities [8] to provide cross-platform compatibility without app installation requirements. The system's lecturer-scanning QR model (where the lecturer scans student-displayed codes) differs from the conventional student-scanning-projector model [3], [7], reducing the probability of screenshot sharing among students.

The Firebase Firestore real-time synchronization architecture provides live attendance monitoring capabilities consistent with cloud-based attendance recommendations. Local fingerprint processing via the Node.js bridge application contrasts with cloud-based biometric template transmission, prioritizing privacy and reducing latency. The perceptual hashing approach, while computationally efficient, represents a trade-off against dedicated biometric matching algorithms described in the biometric systems literature [5], [6], [12], [13].

3.6. Limitations

Several limitations were identified during implementation. The DigitalPersona scanner requires a Windows operating system exclusively, preventing operation on macOS or Linux platforms. The perceptual hash fingerprint matching algorithm has a narrow differentiation margin, creating potential for false positives and false negatives. The system requires a continuous internet connection for all operations, including QR code generation and attendance marking, with no offline data capture capability. Fingerprint templates stored in Firestore (approximately 500 KB per student)

create scalability and cost concerns at large institutional scale. Administrator accounts require manual creation via the Firebase Console without a self-service workflow.

4. Conclusion

This paper presented OmniAttend, a web-based hybrid attendance management system that integrates QR code scanning and biometric fingerprint verification for higher education institutions. The system was built on a three-tier architecture combining a React.js frontend, Firebase Firestore backend, and a local Node.js fingerprint bridge, developed incrementally to deliver a modular, testable solution.

The hybrid verification model successfully eliminates the forced security-versus-accessibility trade-off that limits single-method systems. By enabling lecturers to configure verification mode per session and providing students with multiple attendance pathways, OmniAttend offers operational flexibility without sacrificing data integrity. The unified attendance log with method indicators supports comprehensive auditing and semester-end reporting.

Key limitations — Windows-only fingerprint scanner support, perceptual hash matching accuracy, and network dependency identify clear directions for future work. Recommended enhancements include adopting dedicated fingerprint matching algorithms to improve biometric accuracy, implementing offline attendance capture with synchronization, developing dedicated mobile applications, integrating facial recognition as an additional verification modality, and connecting with institutional Student Information Systems. These improvements would further strengthen the system's suitability for large-scale deployment in diverse higher-education contexts.

Compliance with ethical standards

Acknowledgments

The authors express sincere gratitude to God Almighty for guidance throughout this research. Profound appreciation is extended to Oluwaseyi Segun Adediran (Project Supervisor) for invaluable guidance, constructive criticism, and continuous support. The authors also acknowledge the lecturers and staff of the Department of Information Technology, Babcock University, and their parents and families for their financial support and encouragement during this study.

Disclosure of conflict of interest

The authors declare that there are no financial or non-financial conflicts of interest related to this research.

References

[1] A. Aristovnik, D. Keržič, D. Ravšelj, N. Tomaževič, and L. Umek, "Impacts of the COVID-19 pandemic on life of higher education students: A global perspective," *Sustainability*, vol. 12, no. 20, p. 8438, 2020, doi: 10.3390/su12208438.

[2] D. Turnbull, R. Chugh, and J. Luck, "Transitioning to e-learning during the COVID-19 pandemic: How have higher education institutions responded to the challenge?," *Educ. Inf. Technol.*, vol. 26, no. 5, pp. 6401–6419, 2021, doi: 10.1007/s10639-021-10633-w.

[3] S. Srivastava, "Advance QR Code-Based Attendance Management System," in *Proc. IEEE Int. Conf. on Computing, Communication and Security*, 2023.

[4] E. S. K. Siew, Z. Y. Chong, S. N. Sze, and R. Hardi, "Streamlining Attendance Management in Education: A Web-Based System Combining Facial Recognition and QR Code Technology," *J. Adv. Res. Appl. Sci. Eng. Technol.*, vol. 33, no. 2, pp. 198–208, 2023, doi: 10.37934/araset.33.2.198208.

[5] W. Birouk, A. Lahoulou, A. Bouridane, and M. H. Kheli, "Robust perceptual fingerprint image hashing: a comparative study," *Int. J. Biometrics*, 2023.

[6] A. K. Jain, A. A. Ross, and K. Nandakumar, *Introduction to Biometrics*. New York, NY, USA: Springer, 2011.

[7] Winanti, A. H. A. Latif, and D. Kurniawan, "Integrating QR Code to e-Class System for Managing Attendance Lecture Services in Higher Education," *Int. J. Inf. Eng. Electron. Bus.*, vol. 16, no. 4, pp. 93–104, 2024, doi: 10.5815/ijieeb.2024.04.06.

[8] A. Biørn-Hansen, T. A. Majchrzak, and T.-M. Grønli, "Progressive Web Apps for the unified development of mobile applications," in *Web Information Systems and Technologies*, Springer, 2018, pp. 64–86.

[9] Kanapathy P, Mustafa N, Yusop N, Ibrahim Y. *"Enhancing attendance authenticity in higher education using a QR code and GPS-based smart attendance system".* Int J Res Innov Soc Sci. 2026;10(1):7855–7867. doi:10.47772/IJRISS.2026.10100611

[10] J. Hendrawan, "QR Code-Based Attendance Systems in Education: A Systematic Literature Review," in *Proc. Int. Conf. on Computer, Education, and Social Sciences*, 2025.

[11] M. Aldabagh et al., "A Review of Students Attendance Management Systems," *EAJSE*, 2024.

[12] M. Hamadouche et al., "Securing Biometric Systems by using Perceptual Hashing Techniques," in *Proc. IEEE Int. Conf. on Biometrics*, 2023.

[13] T. Tongse et al., "Hybrid Approach for Attendance ERP Using Biometric and QR Code Verification," *Int. J. Sci. Technol. Eng.*, 2025.

[14] A. Nuhi et al., "Smart Attendance System using QR Code," in *Proc. IEEE Conf. on Information Technology*, 2020.