

A Secure and Privacy-Preserving NFC-Based Mobile Attendance Monitoring System for Higher Education

Dorin Iordache

Ovidius University Constanta
Bd. Mamaia, nr. 124, Romania
dorin.iordache@365.univ-
ovidius.ro

Mihai Teliman

Ovidius University Constanta
Bd. Mamaia, nr. 124, Romania
mihai.teliman@365.univ-
ovidius.ro

Aurelian Nicola

Ovidius University Constanta
Bd. Mamaia, nr. 124, Romania
aurelian.nicola@365.univ-
ovidius.ro

ABSTRACT

Attendance recording in educational environments is still often done on paper, which offers limited auditability and is vulnerable to fraud. Although QR-code and biometric solutions improve automation, they also introduce trade-offs: QR codes remain vulnerable to remote sharing, while biometric systems raise privacy and deployment concerns because they process sensitive personal data. This paper proposes an NFC-based attendance solution with local processing that preserves privacy and is suitable for university environments. Each student is associated with an NFC tag identifier without storing personal data on the tag, and all attendance records are encrypted on the instructor's Android device. The system was implemented using MVVM, Room, SQLCipher, and Android Keystore-based protection. It was evaluated in a real university setting across multiple attendance sessions. Results show that NFC-based registration is fast, reliable, and suitable for real-time use, while supporting GDPR-aligned data minimization, security, and usability.

AUTHOR KEYWORDS

NFC; Security; Privacy; GDPR; Encryption; Authentication; Attendance.

DOI: 10.37789/icusi.2026.6

INTRODUCTION

Student attendance management is a core administrative function in higher education. In many institutions, attendance is still based on paper or manually updated electronic records. These approaches are susceptible to fraud, require administrative costs, and complicate both centralized reporting and compliant management of personal data. The use of smart-mobile technology has automated attendance recording, performing it in near real-time, with the effect of increasing operational efficiency and data integrity. Mobile solutions reduce registration latency, reduce administrative burden, and support near real-time consolidation of attendance records [5]. Such systems must simultaneously meet data protection obligations under the General Data Protection Regulation (GDPR), requiring appropriate technical and organizational measures, such as encryption of personal data, and adherence to the principles of privacy by design. This paper presents an NFC-based mobile technology attendance system designed for campus environments characterized by intermittent connectivity and increased privacy requirements. In the proposed architecture, NFC-enabled Android smartphones serve exclusively as tag readers; all student records are stored on the device in a local AES-256 encrypted database governed by controlled access mechanisms. Although NFC tags do not contain any

personally identifiable information, securing the associated records ensures that students remain anonymous in the event of tag compromise. Given the evolving threat landscape, encrypting locally stored personal information is a key control against unauthorized disclosure [2]. The system incorporates platform-level defenses, including secure storage of cryptographic keys via Android Keystore, local databases encrypted via SQLCipher, and robust user authentication. The threat model explicitly addresses device theft, database extraction, and application-level compromise. An Android application was developed using the platform's NFC APIs. The system was evaluated in a real university setting across multiple attendance sessions and student groups. Experimental results demonstrate an NFC registration latency below 20 ms in steady-state conditions, resulting in a significant safety margin compared to the 100 ms human perception threshold [13], while maintaining strong privacy guarantees and data minimization in compliance with GDPR.

RELATED WORK AND COMPARATIVE POSITIONING

The proposed NFC attendance model aligns with three main families of current solutions: web/QR code systems, biometric and NFC hardware platforms, and privacy-oriented architectures aimed at GDPR compliance. QR-code and web-based attendance systems remain popular for their low cost and straightforward deployment [9, 19]. Mobile devices scan static or dynamic QR codes, enabling students to register without extra hardware. Although dynamic QR codes reduce replay attacks, QR images can still be forwarded remotely, allowing absent students to participate by sharing images. Some implementations add geofencing or location checks to reduce fraud [14], but indoor positioning is degraded by signal loss and multipath effects caused by building structures, which severely limit GPS accuracy indoors [16]. Moreover, continuous GPS tracking raises privacy concerns because pervasive location logging can exceed the purpose of attendance verification. Recent research has attempted to mitigate these shortcomings by exploiting machine learning and cloud services to automate attendance capture, yielding substantial time savings over manual approaches [5]. These observations indicate that although QR code-based schemes are economical and simple [18], they are inadequate for high-integrity presence scenarios because they cannot reliably guarantee physical presence and remain vulnerable to proxy presence through QR code sharing [11].

Biometric platforms represent a second major direction: fingerprint, facial recognition, NFC cards, and other modalities into centralized terminals [1, 10, 15], improving identity assurance and spoofing resistance [15], but substantially increasing

system complexity. Fingerprint subsystems face practical issues stemming from sensor variability, image quality, and vendor interoperability [1], complicating integration and maintenance in institutional deployments. Biometrics deliver high authentication accuracy and anti-spoofing strength, making them attractive when identity certainty is paramount. However, campus-wide rollout demands significant hardware investment, maintenance, and specialized infrastructure, which can be prohibitive for budget-limited institutions [10]. The heterogeneity of sensors and acquisition standards [1] further raises the cost of achieving consistent behavior across devices. Critically, biometric data is treated as sensitive personal data under Article 9 of the GDPR, requiring a lawful basis and stringent technical and organizational safeguards [8, 17]. Unlike passwords, biometric traits are permanent; a breach of biometric data has long-term, irreversible privacy implications [8]. Systematic reviews of facial-recognition attendance illustrate high accuracy but also document bias, privacy, and legal compliance concerns [20], underscoring the ethical and regulatory sensitivity of biometric solutions in education.

Standalone NFC-based attendance systems have also been explored independently of QR-code or biometric approaches. Early work demonstrated the technical feasibility of NFC-enabled smartphones for automated classroom attendance capture using tag UUIDs read by the lecturer's device [4], while comparative studies contrasted NFC against RFID-based deployments in terms of cost and infrastructure requirements [6]. More recent systems have combined NFC with complementary verification mechanisms, such as GPS-based location checks to further constrain proxy attendance [7], or evaluated dedicated NFC tag deployments for automated attendance data collection pipelines [11]. However, none of these systems provide a structured security analysis of the stored attendance data or address GDPR-driven privacy requirements, which is a central contribution of the present work.

Alternatives like ultra-wideband (UWB) indoor positioning deliver precise location detection useful for attendance [3], but UWB requires specialized transceivers and dense infrastructure, increasing complexity and cost for broad university deployment. Hybrid approaches that combine dynamic QR codes with geofencing and IMEI checks (e.g., Nwabuwe et al. [14]) strengthen fraud resistance by embedding location coordinates in the QR and validating device location and IMEI. While these methods can counter proxy attendance and token sharing, they depend on a mobile app, continuous location verification, and device identifiers, thereby introducing privacy and GDPR concerns related to device-level monitoring.

Attendance systems in educational contexts must comply with GDPR principles such as lawfulness, fairness, purpose limitation, data minimization, and accountability. Recent studies promote privacy-aware architectures that limit personally identifiable data collection while preserving authentication reliability. These analyses reveal a trade-off: biometric systems gather highly sensitive data, whereas QR schemes collect minimal data but remain prone to fraud, creating tension between security, privacy, and implementation cost.

In this landscape, the NFC-based attendance approach offers a middle ground between security, privacy, expense, and usability. Versus QR-code methods [19], NFC requires physical proximity to the instructor's device, reducing remote-sharing fraud and token distribution attacks. Unlike GPS-dependent

schemes, NFC does not rely on continuous location tracking, avoiding indoor positioning limitations caused by signal degradation and multipath [16] and the surveillance concerns of constant geolocation monitoring [14]. In contrast to biometric deployments [10, 20], the solution avoids storing biometric templates or images, relying instead on inexpensive NFC tags, on-device processing, and encrypted local storage; using only the tag UUID as presence marker aligns with GDPR data minimization [8]. Compared with dynamic QR + geofencing + IMEI methods, it also reduces reliance on persistent connectivity and device-level tracking, shifting trust toward physical proximity and the instructor's device. The result is a privacy-conscious, moderately low-cost, operationally simple scheme capable of verifying physical presence without retaining sensitive biometric data. Thus, the proposed NFC design emerges as a pragmatic alternative that balances security, privacy, cost, and usability, filling the gap between weak QR systems, intrusive geofencing/app strategies, and privacy-sensitive biometric options, making it well-suited for educational settings with limited infrastructure and strict GDPR constraints.

PROPOSED SYSTEM ARCHITECTURE

Use case and stakeholders

The system is designed around a single primary actor: the instructor, who operates all application features via their mobile device. Students interact indirectly through pre-assigned NFC tags and do not need to use the application themselves. The interaction model and functional relationships are summarized in the UML Use Case diagram shown in Figure 1. System functionality is grouped into four principal categories, corresponding to the subsystems depicted in the Use Case diagram. The first group, Group Management, covers administration of academic entities, including importing timetables and student rosters from CSV files, and inspecting group details via the Timetable Import, Student Import, and Group View use cases. The second group, Tag Management, handles NFC tag administration: the instructor can bind or unbind NFC tag UUIDs to student records using the Assign Student Tag and Remove Student Tag use cases.

The third group, Attendance Processing, is the core attendance module. It supports creating attendance sessions (Create Attendance), which incorporates two mandatory include relationships—Student Search and Student Tag Scan—reflecting the two identification methods supported: manual lookup and NFC scanning. This subsystem also allows viewing session details (Attendance View) and exporting the data for

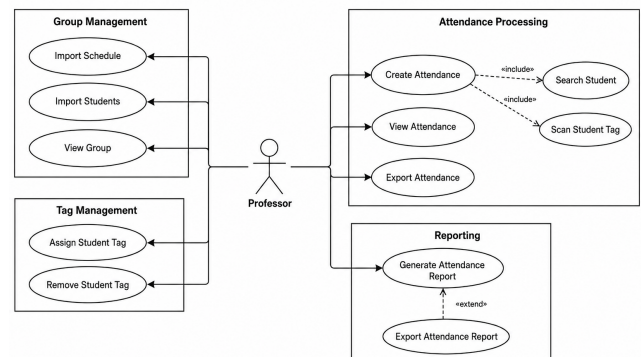


Figure 1: Use Case Diagram.

an individual session (Attendance Export). The fourth group, Reporting, addresses semester-level reporting through the Attendance Report Generation use case. This case is optionally augmented by Attendance Report Export (an extend relationship), denoting that report export is an additional behavior distinct from the per-session Attendance Export provided in the Attendance Processing subsystem. Operationally, the workflow starts with semester initialization, proceeds to NFC tag assignment, and attendance is captured during each lecture or lab session.

Functional Requirements

The main objectives of the proposed system can be formalized through the following functional requirements:

- **FR1 – Academic Data Import:** The system shall support the import of academic schedules and student lists from CSV files, without requiring a fixed column order or predefined header names, by dynamically matching column content to the expected data fields.
- **FR2 – NFC Tag Management:** The system shall enable the association of the hardware UID of a physical NFC tag with a corresponding student record stored in the database, as well as the reassignment or removal of this association when necessary.
- **FR3 – Attendance Session Management:** The system shall allow the creation of attendance sessions for lectures or laboratory activities, supporting student registration both through NFC tag scanning and manual search mechanisms.
- **FR4 – Attendance Validation:** The system shall verify that a scanned NFC tag is linked to a valid student record and ensure that the same student is not registered multiple times within the same attendance session. In both cases, the system shall display a distinct warning notification indicating the nature of the validation failure.
- **FR5 – Data Persistence:** The system shall locally store attendance sessions and their associated attendance records, including the timestamp corresponding to each attendance entry.
- **FR6 – Data Export:** The system shall provide export functionality for both individual attendance sessions and semester-level attendance reports, both in CSV format, with support for Romanian diacritical characters.
- **FR7 – Attendance Reporting:** The system shall generate attendance reports for each academic subject, automatically calculating attendance percentages and highlighting, by means of a distinct color indicator, students whose attendance rate falls below a minimum threshold defined by the instructor within the application settings.
- **FR8 – Offline Operation:** The system shall operate entirely without internet connectivity, with all data being processed and stored exclusively on the instructor's mobile device.

Together, these requirements—covering academic imports, NFC tag handling, session management, validation, local storage, export functionality, and offline operation—align with the architectural constraint of

a self-contained mobile application with no server-side dependencies, where all processing, storage, and interaction occur on the instructor's device.

Database structure

The database schema represented in Figure 2 is divided into two functional modules: an academic hierarchy and an attendance management module. The academic hierarchy models the institutional structure as Faculty → Specialization → Year of study → Group → Student, with each entity referencing its parent via foreign keys. The Student table stores student details and the associated NFC tag identifier (TagID), which can be null until a tag is assigned. The Professor table contains instructor information and is linked into the teaching model through foreign keys in the Course and Laboratory tables. Course and Laboratory entries define teaching activities: Course references the year of study and the assigned professor, while Laboratory links to the course, professor, and the relevant academic group. The Timetable table specifies scheduled entries, including start/end times, day of the week, recurrence frequency, and foreign keys to the linked Lab and Course records. Timetable also directly references the Group to unambiguously identify which group the scheduled activity pertains to. The Attendance module handles recording. The Sessions table represents a concrete instance of a teaching activity and may reference a Timetable entry via an optional foreign key, allowing ad-hoc sessions outside the predefined schedule. The Attendance table logs each student's participation for a session, including the timestamp of the NFC scan.

APPLICATION IMPLEMENTATION

Architectural overview

The system is implemented as a native Android application designed exclusively for academic staff and operates entirely offline, without requiring communication with external servers. All application logic and data are stored locally on the instructor's mobile device. From an architectural perspective, the application follows the Model-View-ViewModel (MVVM)

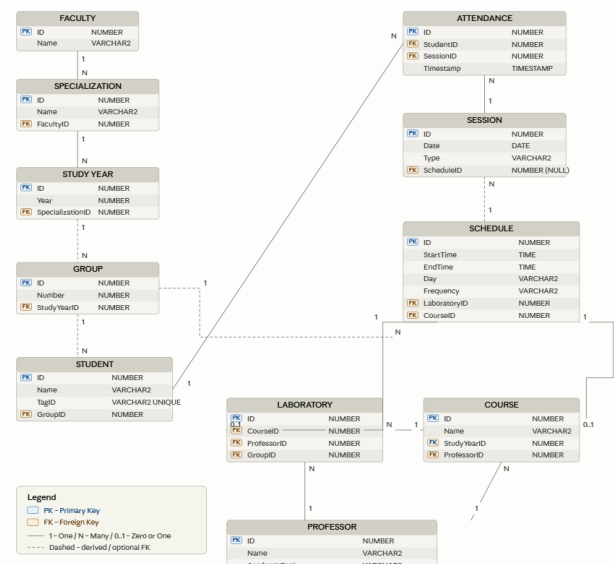


Figure 2: Database structure.

pattern recommended by Google for modern Android development. The architecture separates the system into three distinct layers. The Model layer represents the single source of data and includes the database entities, the data access interface (UniversitateDAO), and the UniversitateRepository class, which abstracts data persistence from the upper layers. Local persistence is implemented using the Room library, configured to operate on top of SQLCipher rather than the standard SQLite engine, providing AES-256 file-level encryption for all stored data. The ViewModel layer contains the business logic of the application, including NFC workflow management, CSV import/export processing, and data orchestration. Communication with the user interface is achieved through LiveData objects, enabling automatic UI updates when data changes. The View layer is responsible only for data presentation and user interaction. It consists of fragments, adapters, and dialogs coordinated through a Single Activity – Multiple Fragments design centered around MainActivity. Additionally, the system integrates a transversal NFC communication module that enables the instructor's device to read students' physical NFC tags. NFC events are captured by the Android operating system through the Intent mechanism and routed by MainActivity to the currently active fragment.

Platform and technologies

The application was developed in Java using Android Studio as the integrated development environment. Java was selected for its mature ecosystem, extensive Android SDK support, and broad availability of documentation and community resources for Android development. The system targets Android 7.0 (API level 24) as the minimum supported version, reflecting the adoption threshold above which the required Jetpack libraries and NFC APIs are fully supported, while covering the majority of active Android devices. Deployment requires only an NFC-capable mobile device, without any additional dedicated hardware. The implementation relies on several Android technologies and libraries. The Android SDK manages the application lifecycle, Intent-based communication, and NFC functionality through the NfcAdapter interface. Data persistence is implemented using the Android Jetpack Room library, configured to operate on top of SQLCipher instead of the standard SQLite engine, providing AES-256 encryption for locally stored data. The architecture integrates the Jetpack ViewModel and LiveData components for business logic separation and reactive UI updates. CSV import and export operations are handled through the Android Storage Access Framework, ensuring secure access to external storage. Encryption key management is handled through the Android Keystore system, which provides hardware-backed secure storage for cryptographic keys. Project compilation and dependency management are performed using Gradle.

Application architecture

The project structure follows the MVVM architecture, improving modularity, scalability, and maintainability. The Model layer contains the database entities, DAO interfaces, Room database configuration with SQLCipher encryption, and type converters. The ViewModel layer includes dedicated ViewModels responsible for attendance management (AttendanceViewModel), NFC tag enrollment (EnrollmentViewModel), and application settings (SettingsViewModel). The UI layer

contains activities, fragments, adapters, and dialogs responsible for user interaction and presentation logic. Communication between layers follows a unidirectional flow: the View observes LiveData exposed by the ViewModel, the ViewModel accesses the Repository layer, and the Repository communicates with the database through DAO interfaces. NFC events represent a transversal communication flow, intercepted by MainActivity and forwarded to the active fragment and corresponding ViewModel, as illustrated in Figure 3.

Security Mechanisms

Although the offline architecture significantly reduces the attack surface compared to server-based solutions, the instructor's mobile device remains vulnerable to scenarios such as theft or unauthorized access. To address these risks, the system integrates multiple complementary security mechanisms. The implemented mechanisms assume a non-rooted device operating within a trusted Android environment, as elaborated in the security analysis. The application database is protected using SQLCipher, which replaces the standard SQLite engine and provides AES-256 file-level encryption. As a result, the database remains unreadable without the corresponding decryption key, even in cases of direct file system access.

Encryption key management is implemented through a two-layer mechanism based on the Android Keystore system. During the first application initialization, a hardware-backed AES-256 key is generated and securely stored within the Keystore, preventing software-based extraction. This key is then used to encrypt a randomly generated 32-byte (256-bit) database password using AES/GCM encryption — an authenticated encryption mode that provides both confidentiality and integrity guarantees. The encrypted password and initialization vector are stored locally, while the plaintext password exists only temporarily in memory during database initialization. On subsequent launches, the encrypted password is retrieved and decrypted using the stored Keystore key before being passed to SQLCipher to unlock the database.

To prevent unauthorized data extraction through Android backup mechanisms, all backup features are explicitly disabled. The application configuration blocks cloud backups, device transfer backups, and ADB-based extraction, ensuring that sensitive database files cannot be copied through standard Android recovery mechanisms. Additionally, the compiled application code is obfuscated using ProGuard/R8

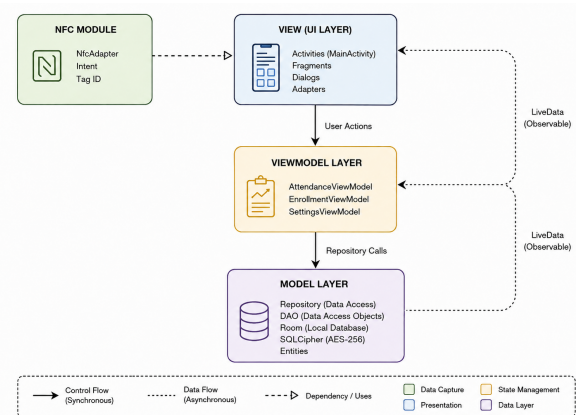


Figure 3: Application architecture.

tools integrated into Gradle. Although not a primary security measure, code obfuscation increases resistance against reverse engineering attempts and complements the layered security strategy implemented by the system.

EXPERIMENTAL SETUP AND PERFORMANCE EVALUATION

Performance evaluation

The performance evaluation was conducted on a Samsung Galaxy A71 (SM-A715F), running Android 13 (One UI 5). The device features a Qualcomm Snapdragon 730 octa-core processor (8 nm), 6 GB RAM, 128 GB UFS 2.1 storage, and hardware NFC support. Samsung Knox provides hardware-backed TEE support for Android Keystore operations. The experimental evaluation was conducted using real academic scenarios involving multiple student groups, attendance sessions, and teaching activities executed on Android devices equipped with NFC functionality. The system was tested in both single-class and multi-class configurations under fully offline operation, reflecting the intended deployment environment. NTAG213 NFC tags were used during all evaluation sessions, and registration latencies were measured through application-level timestamps recorded at the moment of NFC tag detection and database write completion.

Performance evaluation must be interpreted in the context of the physical attendance collection process. Unlike conventional software systems where latency is critical, NFC attendance registration is naturally constrained by human interaction time, namely the time required for students to approach the instructor's device, scan the NFC tag, and allow the next student to perform the same action. This physical interaction lasts several seconds per student and largely exceeds any computational overhead introduced by the implemented security mechanisms.

In our measurements, the protocol-level tag negotiation (ISO/IEC 14443A) contributed only a few milliseconds to the total latency. Additional processing time introduced by the Android NFC stack, Keystore decryption, and SQLCipher database access brings the total end-to-end registration latency to under 20 ms under normal conditions, as confirmed by the experimental measurements presented below. According to the official SQLCipher documentation [21], AES-256 database encryption introduces an overhead of approximately 5–15% compared to standard SQLite operations. In the proposed system, attendance registration requires only a simple student lookup and an insert operation, resulting in execution times well below the threshold of human perception regardless of encryption overhead.

To characterize the complete latency distribution of the attendance registration operation, Figure 4 presents the Cumulative Distribution Function (CDF) of the steady-state population ($n = 382$, excluding first-scan initialization events). The CDF confirms that 50% of scans complete within 3.70 ms, 95% within 10.45 ms, and 100% of steady-state registrations complete within 19.40 ms — well below the 100 ms human perception threshold [12, 13]. This distribution demonstrates that the cryptographic overhead introduced by SQLCipher and Android Keystore does not produce any perceptible delay in the attendance registration workflow.

As shown in Table 1, the steady-state median of 3.70 ms and P95 of 10.45 ms remain almost an order of magnitude below the 100 ms human perception threshold, leaving a

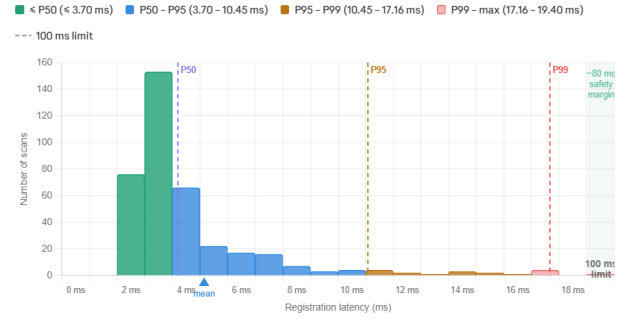


Figure 4: Cumulative Distribution Function (CDF).
CDF of NFC-to-database registration latency — steady-state population ($n = 382$). Dashed lines indicate P50, P95, and the 100 ms human perception limit.

Population	N	Mean	Median	P95	P99	Max
All measurements	389	5.60	3.72	12.56	24.46	196.25
Steady-state (excl. first scan, >20 ms)	382	4.65	3.70	10.45	17.16	19.40

Table 1: NFC attendance registration latency statistics.

First-scan outliers ($n = 7$, >20 ms) reflect Keystore key derivation and are excluded from steady-state analysis.

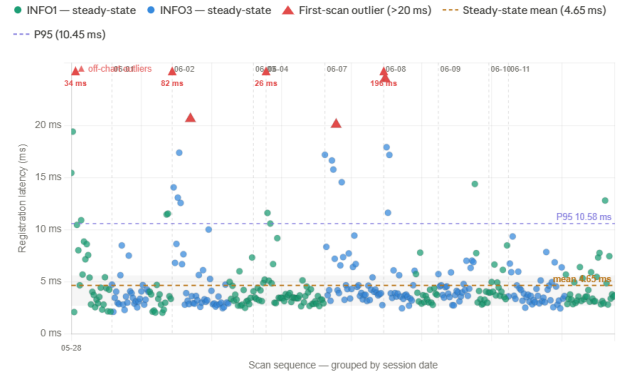


Figure 5: Chronological timeline of NFC attendance registration latency.

Chronological timeline of NFC attendance registration latency across all sessions ($n = 389$). Triangular markers indicate first-scan initialization outliers (>20 ms). Vertical lines delimit session boundaries by date.

wide margin even under less favorable hardware or usage conditions. Overall, the implemented security mechanisms — SQLCipher encryption, Android Keystore-based key management, and backup protection — can therefore be deployed without any usability trade-off, at the cost of a modest and well-characterized runtime overhead.

Figure 5 presents the chronological sequence of all 389 individual scan latencies across ten measurement sessions spanning May 28 to June 11, 2026, grouped by date and colored by course cohort (INFO1 — green; INFO3 — blue). First-scan initialization outliers ($n=7$, >20 ms) are marked as triangles. Vertical dashed lines delimit session boundaries. The steady-state mean line (4.65 ms) remains consistent across all sessions, and no upward drift is observable over the full measurement period as the database accumulated records. Two exceptional cold-start events are visible above the plot area: 82.12 ms (INFO3GR1, June 2) and 196.25 ms (INFO3GR2, June 8), both attributable to Android Keystore re-initialization rather than system degradation. This confirms that system performance is stable over time and independent of database growth.

Attack Vector	Protection Mechanism	Residual Risk
Physical theft of the device	AES-256 database encryption using SQLCipher	Low
Database extraction through ADB	Backup disabling combined with database encryption	Low
Unauthorized cloud backup	Android data_extraction_rules restrictions	Eliminated
Data transfer between devices	Android data_extraction_rules restrictions	Eliminated
Reverse engineering of the APK	ProGuard/R8 code obfuscation	Medium
Compromised or rooted device	Encryption persists, but root access may extract the Keystore key	High
NFC tag UID cloning/emulation	Requires prior physical access to the victim's tag; only exploitable when the enrolled student is absent	High

Table 2: Attack vectors and corresponding protection mechanisms.

Security and robustness analysis

The implementation of the security mechanisms described in Section IV addresses the main attack scenarios identified for the proposed architecture. The attack vectors identified and the corresponding mitigations are analyzed in the following subsection.

Threat Model. The security analysis of the proposed system is grounded in a structured threat model that defines the adversarial context, the assets under protection, and the assumptions governing the deployment environment, and is summarized in Table 2.

Adversarial actors. Four distinct attacker profiles are considered. A physical attacker who can gain temporary or permanent access to the instructor's device—assuming it is locked or without an active app session—through theft or unattended access. An Android Debug Bridge (ADB)-based attacker connects the device to a workstation via USB and attempts to extract app data via ADB commands or backup mechanisms. A privileged attacker operates on a rooted device and may attempt to bypass OS-level isolation to directly access the app's memory or encrypted database. An opportunistic attacker acquires a copy of the distributed APK file or gains access to backup artifacts—via cloud storage, device transfer mechanisms, or third-party backup services—and attempts to extract app logic or data outside of the protected execution environment. A *peer attacker* attempts to circumvent attendance verification by cloning or emulating the UID of another student's NFC tag, exploiting the absence of cryptographic authentication on the deployed tag type.

Protected assets. The primary assets targeted by the identified threat scenarios are: (i) the SQLCipher-encrypted database containing student records, NFC tag associations, and attendance history; (ii) the hardware-backed AES-256 key stored within the Android Keystore, used to encrypt the SQLCipher database password.

Attack vectors and mitigations. Table 2 maps attack vectors to the corresponding protection mechanisms. SQLCipher

encryption addresses physical and ADB-based extraction; Keystore-based key management prevents software-level key recovery; backup restrictions eliminate cloud and transfer-based exfiltration; and ProGuard/R8 obfuscation raises the cost of APK reverse engineering. Residual risk from privileged attackers on rooted devices, as well as from UID cloning by a peer attacker, remains outside the implemented protection scope, as mitigating the latter would require cryptographically authenticated tags.

Security assumptions. The threat model under consideration is based on the following assumptions, which define the limits of the security guarantees provided: (i) the deployment environment consists of institutional mobile devices that are not rooted and are subject to organizational security policies; (ii) the Android platform kernel and the Keystore hardware enclave are considered trusted; application-level components and user space are outside the trusted compute base; and (iii) the attacker does not have persistent physical access to an unlocked device with an active application session.

SQLCipher database encryption ensures that stored data remains unreadable even in cases of direct file system access, while the use of a hardware-based Android Keystore prevents software-based recovery of database decryption credentials. In addition, disabling cloud backups and device migration eliminates a frequently overlooked attack vector in Android applications. The main limitation of the proposed solution concerns rooted devices. In such scenarios, the process isolation provided by the Android sandbox can be bypassed, as an attacker with root privileges could access the application's memory while the database password exists in clear text during an active session. This residual risk is considered acceptable in the context of intended academic deployment, where rooted devices are typically excluded from institutional security policies.

Although the NFC tags used (NTAG213) lack cryptographic authentication and their UID can in principle be read and duplicated onto a writable tag or emulated via a smartphone, exploiting this vulnerability requires prior physical access to the victim's tag and is only effective when the enrolled student is absent from the session, since the system associates each scanned UID with the corresponding student record and would register a legitimate scan by the tag owner as normal attendance in any subsequent session. This physical-access requirement, combined with the low per-session incentive relative to the effort involved, is likely to limit the practical scale of such fraud, although it does not eliminate the underlying risk. Unlike encoding schemes with cryptographic tag authentication (e.g., NTAG 21x DNA or MIFARE DESFire), which could mitigate this vector, the residual risk associated with UID cloning is assessed as High under our threat model, consistent with the rating applied to rooted-device compromise, and represents a direction for future hardware-level hardening.

Privacy and GDPR Compliance. The application architecture is designed to comply with GDPR principles through privacy and security measures. Data minimization is applied at the attendance registration level, as follows: for each attendance event, only a reference to the student record, the NFC tag UID and the timestamp of the registration are captured. The broader database schema additionally stores academic organizational data - faculty hierarchy, course and student group

structure - which are necessary for the system to function and are equally subject to the same protection mechanisms. The NFC UID itself does not allow for the identification of the individual without access to the application's protected database; however, the association between the UID and the student identity stored in the encrypted database constitutes personal data under Recital 26 of the GDPR, justifying the applied encryption and access controls. No biometric, location or other sensitive data is collected. Purpose limitation is supported by the application design, which restricts the operations available to attendance tracking and academic reporting, with no functionality that allows the use of secondary data. The offline design eliminates automatic transmission of data to third parties and prevents network-based behavioral profiling, reducing the risk of unintended secondary uses. Data integrity and confidentiality are ensured through AES-256 encryption via SQLCipher and secure key management using Android Keystore; backup mechanisms are disabled to prevent unintended exposure. This design prioritizes privacy over availability, and the absence of a secure backup constitutes a limitation with respect to Article 32 GDPR. Broader compliance aspects — including data retention, deletion schedules, and access control policies — fall outside the current implementation scope and are identified as future work.

CONCLUSIONS AND FUTURE WORK

This study presents the design, implementation, and experimental evaluation of an Android mobile attendance monitoring application using NFC-tokens that can be used in educational institutions. The proposed system is a practical and efficient alternative to manual attendance monitoring processes, which are error-prone, difficult to audit, and susceptible to fraud.

Attendance automation improves productivity and supports more reliable recordkeeping. As discussed in the experiments section, NFC strikes a favorable balance between security, utility, and implementation cost compared to QR and biometric methods. NFC's short-range interaction requires close physical presence and using only the tag's UID-instead of storing personal identifiers on the tag - aligns the design with GDPR data minimization objectives, enhancing privacy even if the tag is compromised. The system implements an autonomous workflow for attendance monitoring, including timetable import, student roster management, NFC tag enrollment, real-time registration, and periodic attendance reporting. Experimental results confirm steady-state registration latencies below 20 ms across all sessions, with a median of 3.70 ms, demonstrating that the cryptographic overhead introduced by SQLCipher and Android Keystore does not affect real-time usability. In conclusion, the application provides an institutionally relevant and privacy-aware attendance management tool suitable for deployment at the university department level. The main limitations concern data lifecycle management: the system currently provides neither automated retention-based deletion nor a secure backup mechanism. Future work could explore optional centralized synchronization for multi-instructor deployments, subject to appropriate GDPR safeguards, alongside role-based access control and integration with existing university IT systems to improve interoperability and long-term usability.

ACKNOWLEDGMENTS

This research received no external funding. All research and publication costs were covered by the authors' own funds.

References

- [1] A. J. M. Abdul Cader, Jasmine Banks, and Vinod Chandran. 2023. Fingerprint Systems: Sensors, Image Acquisition, Interoperability and Challenges. *Sensors* 23, 14 (2023), 6591. doi:10.3390/s23146591
- [2] Wasyihun Sema Admass, Yirga Yayeh Munaye, and Abebe Abeshu Diro. 2024. Cyber Security: State of the Art, Challenges and Future Directions. *Cyber Security and Applications* 2 (2024), 100031. doi:10.1016/j.csa.2023.100031
- [3] Abdulrahman Alarifi, AbdulMalik Al-Salman, Mansour Alsaleh, Ahmad Alnafes-sah, Suheer Al-Hadhrani, Mai A. Al-Ammar, and Hend S. Al-Khalifa. 2016. Ultra-Wideband Indoor Positioning Technologies: Analysis and Recent Advances. *Sensors* 16, 5 (2016), 707. doi:10.3390/s16050707
- [4] Bálint Benyó, Bálint Sódor, Tamás Doktor, and Géza Fördös. 2012. Student Attendance Monitoring at the University Using NFC. In *Proceedings of the Wireless Telecommunications Symposium (WTS)*. 1–8. doi:10.1109/WTS.2012.6266137
- [5] Nilton E. Cayturo-Silva et al. 2024. Optimizing Attendance Management in Educational Institutions Through Mobile Technologies: A Machine Learning and Cloud Computing Approach. *International Journal of Interactive Mobile Technologies (IJIM)* 18, 12 (2024), 112–128. doi:10.3991/ijim.v18i12.46917
- [6] C. B. Chew, Manmeet Mahinderjit-Singh, K. C. Wei, T. W. Sheng, M. H. Husin, and N. H. Ahmed Hassain Malim. 2015. Sensors-Enabled Smart Attendance Systems Using NFC and RFID Technologies. *International Journal of New Computer Architectures and their Applications (IJNCAA)* 5, 1 (2015), 19–28.
- [7] Tzu-Wei Chiang, Chao-Yang Yang, Guo-Jyun Chiou, Frank Yeong-Sung Lin, Yao-Nan Lin, Vincent R. L. Shen, Tong-Ying Tony Juang, and Chun-Yao Lin. 2022. Development and Evaluation of an Attendance Tracking System Using Smartphones with GPS and NFC. *Applied Artificial Intelligence* 36, 1 (2022), 2083796. doi:10.1080/08839514.2022.2083796
- [8] European Parliament and Council of the European Union. 2016. *Regulation (EU) 2016/679 on the Protection of Natural Persons with Regard to the Processing of Personal Data and on the Free Movement of Such Data (General Data Protection Regulation)*. Technical Report. Official Journal of the European Union. 1–88 pages. <https://gdpr-info.eu/>
- [9] Joko Hendrawan, Ika Dyah Perwitasari, and Fajar Maulana. 2025. QR Code-Based Attendance Systems in Education: A Systematic Literature Review on Data Accuracy and Sustainable School Management. In *Proceedings of the International Conference on Computer Science, Engineering, Social Science, and Multi-Disciplinary Studies*, Vol. 1. 80–87. doi:10.64803/cessmuds.v1.14
- [10] Marcela Hernández-de Menéndez, Ruben Morales-Menendez, Carlos Escobar, and Jorge Arinez. 2021. Biometric Applications in Education. *International Journal on Interactive Design and Manufacturing (IJIDeM)* 15 (2021), 365–380. doi:10.1007/s12008-021-00760-6
- [11] Gunawan Anak Hutagalung, Yolanda Amelia Dalimunte, Iffonny Khairina, Muhammad Zarlis Lubis, Doni Firmansyah, Dedi Nofrianto Sinaga, Iron Sahat Simanjuntak, and Zulfa Indandi. 2024. Attendance Data Collection Using NFC Tags. *International Journal of Research in Vocational Studies (IJRVOCAS)* 3, 4 (2024), 67–72. doi:10.53893/ijrvocas.v3i4.28
- [12] Wei Liu, Fanjiang Lin, Lin Guo, Tse-Hsun Chen, and Ahmed E. Hassan. 2025. GUIWatcher: Automatically Detecting GUI Lags by Analyzing Mobile Application Screenshot. arXiv preprint arXiv:2502.04202. doi:10.48550/arXiv.2502.04202
- [13] Jakob Nielsen. 1993. Response Time Limits. Nielsen Norman Group, <https://www.nngroup.com/articles/response-times-3-important-limits/>. Updated 2014. Accessed March 26, 2026.
- [14] Anthony Nwabue, Balvinder Sanghera, Temitope Alade, and Faith Olajide. 2023. Fraud Mitigation in Attendance Monitoring Systems Using QR Code, Geofencing and IMEI Technologies. *International Journal of Advanced Computer Science and Applications (IJACSA)* 14, 4 (2023), 938–945. doi:10.14569/IJACSA.2023.01404104
- [15] Shalini Pahuja and Namita Goel. 2024. Multimodal Biometric Authentication: A Review. *AI Communications* 37 (2024), 1–23. doi:10.3233/AIC-220247
- [16] Pavel Puricer and Pavel Kovář. 2007. Technical Limitations of GNSS Receivers in Indoor Positioning. In *Proceedings of the 17th International Conference Radioelektronika*. IEEE, 205–209. doi:10.1109/RADIOELEK.2007.371487
- [17] Muhammad Qandeel. 2024. Facial Recognition Technology: Regulations, Rights and the Rule of Law. *Frontiers in Big Data* 7 (2024), 1354659. doi:10.3389/fdata.2024.1354659
- [18] Pratiksha Shirole, Tejaswini Pingat, Snehal Jagdale, Rutuja Jori, and S. K. Thorat. 2025. QR Code-Based Student Attendance System. *International Journal of Advanced Research in Computer and Communication Engineering (IJARCCCE)* 14, 4 (2025), 81–83. doi:10.17148/IJARCCCE.2025.14411
- [19] Xin Wei, Anjali Manori, Nikhil Devnath, Nilesh Pasi, and Vinod Kumar. 2017. QR Code Based Smart Attendance System. *International Journal of Smart Business and Technology* 5, 1 (2017), 1–10. doi:10.21742/ijbsbt.2017.5.1.01
- [20] Bernad Wijaya Aqraldo and Meyliana. 2023. Face Recognition Model for Attendance Systems: A Systematic Literature Review. *Journal of Theoretical and Applied Information Technology* 101, 16 (2023), 6653–6672. <http://www.jatit.org/volumes/Vol101No16/32Vol101No16.pdf>
- [21] Zetetic LLC. [n. d.]. SQLCipher. <https://github.com/sqlcipher/sqlcipher>. Accessed March 26, 2026.