

A Session-Bound Attendance Protocol with Rotating QR Codes and Uncertainty-Aware Geofencing: Implementation and Controlled Evaluation

Smaila Amoanu^{1*}, Bright Wiredu Nuakoh², Stephen Asiedu³

Department of Mathematics and Statistics, University of Nevada, Reno, Nevada, USA¹

African Institute for Mathematical Sciences (AIMS), Kigali, Rwanda²

King's College London, London, United Kingdom³

Abstract—A scanned attendance credential does not, by itself, establish that its holder is eligible or physically present. This study specifies and evaluates a session-bound attendance protocol in Arkos AttendPro, a web-based platform for workforce and event attendance. The protocol combines authenticated rotating QR tokens, session and schedule eligibility checks, valid check-in and check-out transitions, and a geofence rule that tightens acceptance as reported location uncertainty increases. Controlled application tests and synthetic geofence experiments under three location-error models, repeated across 30 seeds, examine credential validity, boundary classification, and concurrent submissions. The principal finding is a security–usability trade-off: at 15 m reported accuracy, the primary radial-error simulation yielded false acceptance of 4.15% and false rejection of 3.70% under the fixed-radius rule; an uncertainty weight of 0.50 reduced false acceptance to 0.00% while increasing false rejection to 20.30%. Alternative error models preserved this trade-off but changed its magnitude. Token checks rejected the tested invalid credentials, and an atomic check-out revision prevented the duplicate success transitions observed in the archived implementation. However, a current forwarded token paired with fabricated inside-geofence coordinates was accepted. The findings show how session binding and conservative geofencing govern attendance decisions, while identifying the limits of browser-reported location and the cost of stricter acceptance policies.

Keywords—Attendance management; rotating QR code; geofencing; location uncertainty; proxy attendance; authentication; controlled evaluation

I. INTRODUCTION

Attendance records are operational evidence. In employment settings, they can affect payroll, lateness and overtime calculations, workforce deployment, compliance reporting, and internal accountability. The value of an attendance system therefore depends not only on how efficiently it records an event, but also on whether the recorded event is trustworthy. Manual registers and supervisor-entered records are slow to reconcile and are vulnerable to transcription error. Basic electronic systems improve speed and storage, but they do not by themselves prevent false claims of presence. A user may ask another person to register on their behalf, reuse a photographed code, submit a request from outside the authorized area, or exploit a workflow that verifies a credential without verifying the session, schedule, and attendance state to which that credential should apply.

Attendance-system research has consequently moved from single-factor recording toward combinations of identity, possession, location, and device evidence. A systematic review by Ali et al. [1] identifies QR codes, barcodes, RFID, magnetic cards, and biometric methods among the dominant approaches. QR codes remain attractive because they are inexpensive, contactless, and compatible with ordinary smartphones. Their simplicity, however, is also a weakness: a static code can be copied or forwarded, and a successful scan may reveal little about the person presenting it or the context in which it is used.

Dynamic and randomized QR schemes reduce this exposure by limiting the useful life of a captured code. Imanullah and Reswan [2] developed a randomized QR-code scheme for attendance, illustrating the use of changing credentials in an inexpensive implementation. The underlying time-based principle is consistent with the broader security rationale for short-lived authentication values, which are accepted only within a prescribed interval [3]. Rotation alone, however, does not prevent a current code from being transmitted and used before it expires. A defensible attendance decision must therefore evaluate additional context.

Geofencing supplies spatial context by comparing a reported device position with an authorized region. Recent systems have used circular or polygonal boundaries to restrict attendance to lecture halls, campuses, offices, and event venues [4], [5], [6], [7]. The closely related study by Nwabuwe et al. [4] combined dynamic QR codes, polygonal geofencing, and device IMEI verification to mitigate proxy attendance and code sharing. More recent implementations have paired dynamic QR authentication with fixed-radius GPS validation and reported latency and indoor–outdoor location drift [8]. These studies establish that combining temporal and spatial evidence is practical; they also make clear that the mere coexistence of a rotating QR code and a geofence is no longer, by itself, a sufficient claim of novelty (Appendix).

A central methodological difficulty is that device location is measured with uncertainty. The W3C Geolocation specification states that browser geolocation provides location information associated with the hosting device but does not guarantee the device's actual physical position; it also exposes an accuracy value, expressed in meters at a 95% confidence level, alongside latitude and longitude [9]. Geofencing research similarly emphasizes the effects of measurement conditions, boundary

*Corresponding author.

design, privacy requirements, and implementation choices [10]. Despite this uncertainty, attendance systems commonly apply the deterministic rule $d \leq R$, where d is the calculated distance from the authorized center and R is the selected radius. Under that rule, an estimate one meter inside the boundary is accepted even when its reported uncertainty is tens of meters. The decision can therefore appear precise even when the underlying measurement is not.

The treatment of location uncertainty is connected to a broader weakness in existing evaluations. Many attendance studies emphasize software construction, black-box functionality, or small user-acceptance exercises. Fewer report false acceptance and false rejection under controlled legitimate and unauthorized scenarios, examine behavior near a geofence boundary, compare alternative location-decision rules, or test simultaneous duplicate submissions. Several systems also validate a code and a location without specifying a complete server-side policy that binds the evidence to an authenticated employee, a particular attendance session, an active time interval, an organizational scope, an assigned schedule, and a valid check-in or check-out transition.

This study addresses these limitations through a session-bound attendance protocol implemented in Arkos AttendPro. The employee device collects and presents evidence, but the server makes the attendance decision. A request is accepted only when the QR token is authentic and temporally valid, the employee is authenticated and eligible, the referenced session is active, the employee falls within the permitted organizational and schedule scope, the attendance transition is valid, and the applicable geofence condition is satisfied. The protocol also compares the conventional distance rule with accuracy-adjusted alternatives so that the security-usability trade-off is measured rather than assumed.

The study is guided by the following research questions:

- How effectively does the proposed session-bound rotating-QR and geofence protocol reject unauthorized attendance attempts relative to single-context baselines?
- Under controlled location error, how do reported accuracy and distance from the geofence boundary affect acceptance and rejection?
- How does the uncertainty weight in the geofence rule change the trade-off between false acceptance and false rejection?
- How does QR-token age determine the forwarding exposure of a copied credential before and after expiration?
- How effectively do database integrity controls prevent duplicate attendance transitions under concurrent submissions, and what limitations arise from a local SQLite configuration under stress?

The principal contributions are fourfold. First, the study specifies a session-bound decision model that integrates token validity, authenticated identity, session state, schedule eligibility, organizational scope, geofence evidence, and attendance-state integrity. Second, it specifies and evaluates a conservative accuracy-adjusted geofence rule, comparing the standard rule

$d \leq R$ with alternatives that tighten acceptance as reported uncertainty increases. This differs from prior uncertainty-based approaches that expand the effective geofence radius [11]. Third, it evaluates static badge QR, rotating-QR-only, geofence-only, and combined verification under a controlled automated design using false acceptance, false rejection, token age, simulated boundary error, and duplicate-record outcomes. Fourth, it retains synthetic decision evidence, executable validation functions, and source hashes, allowing controlled outcomes to be audited. The evaluation distinguishes token and eligibility components, application-route transactions, and stochastic location models.

The remainder of the article reviews relevant attendance and geofencing research, presents the protocol and mathematical decision rules, describes the Arkos AttendPro implementation, defines the controlled evaluation, and reports the resulting security and integrity findings.

II. RELATED WORK

A. Automated Attendance Technologies

Automated attendance systems have been developed around RFID, barcodes, QR codes, magnetic cards, biometric recognition, smartphones, and cloud-based platforms. The review by Ali et al. [1] shows that no single technology satisfies every operational requirement. RFID and card systems can process users quickly, but they depend on issued hardware and may permit credential sharing. Biometric systems bind records more directly to a person, but they introduce cost, privacy, environmental, and failure-rate concerns. Smartphone-based QR and location systems reduce dedicated hardware requirements and are easier to deploy across distributed sites, but they shift the security problem toward token sharing, device trust, location quality, and server-side policy enforcement.

B. Dynamic QR Codes and Credential Integrity

Early work by Masalha and Hirzallah [12] used a displayed QR code and student identity to record lecture attendance. The broader QR security survey by Krombholz et al. [13] shows why the convenience of optical scanning must be considered alongside security and user behavior. Static QR codes primarily identify an object or user; they do not by themselves establish freshness. Dynamic schemes add temporal validity by changing the displayed value or limiting its acceptance interval. Imanullah and Reswan [2] developed a randomized QR-code attendance scheme. Annurrahma et al. [14] later used UUID-based dynamic QR codes with GPS validation in a meeting-attendance application. These systems illustrate two related design requirements: the token should be difficult to predict, and its useful lifetime should be limited. The replay taxonomy of Syverson [15] also motivates specifying the context in which a captured message is reused. Here, forwarding a currently valid token and replaying an expired token are tested separately because expiration addresses only the latter. A remaining question is whether the token is also bound to the exact session and server-side policy under which attendance is recorded. A short-lived token that is not tied to a specific session, version, and lifecycle state may still be reused in an unintended context.

C. Geofencing and Location Uncertainty

Geofencing defines a virtual spatial boundary and triggers a decision when a device is classified as inside or outside that boundary. Tamboli et al. [6] and Eweoya et al. [7] use geofencing as a primary attendance control, while Tajul et al. [5] combine QR scanning with geolocation. Shevchenko and Reips [10] provide a broader methodological account of geofencing and emphasize implementation, privacy, and measurement challenges. These issues are directly relevant to attendance verification because a reported coordinate is an estimate rather than a perfect observation of physical presence. The W3C specification [9] reinforces this limitation and exposes an accuracy measure that can be incorporated into the decision. Several of the reviewed implementations use fixed-radius validation, whereas others adapt the boundary to location uncertainty. These approaches should be distinguished from controlled comparisons of decision policies near the boundary.

Empirical studies of assisted GPS and smartphone positioning document measurement error under different devices and environments [16], [17]. These studies motivate explicit treatment of location error; they do not calibrate the synthetic distributions used in this article.

Location uncertainty and location authenticity are separate problems. An accuracy value describes uncertainty in an estimated position, but it does not establish that the device or measurement process is trustworthy. Tschofenig et al. [18] distinguish several forms of location spoofing, including reporting an arbitrary location and replaying another person's location. Experimental work has also shown that falsified satellite signals can cause ordinary receivers to report incorrect position information, motivating independent checks when location carries a security decision [19]. Location-proof research instead considers mechanisms for providing evidence of where a user was located [20]. The present study models measurement uncertainty in the geofence rule, but it does not treat browser-provided coordinates as cryptographic proof of presence.

D. Combined QR and Geofence Verification

Nwabuwe et al. [4] provide a closely related antecedent to the present study. Their solution combines dynamic QR codes, polygonal geofencing, and IMEI verification and evaluates several attendance-fraud scenarios. Siregar et al. [8] similarly combine dynamic QR codes with a 30-meter geofence and report response time, indoor-outdoor location drift, and rejection of fake-GPS attempts in their black-box tests. These reported tests do not establish resistance to every form of location spoofing. Korkortsi et al. [11] combine cross-platform geofencing, device binding, time restrictions, and an uncertainty-dependent effective radius. Their formulation expands the nominal boundary using a Gaussian quantile and fused location-error estimate. The present rule instead tightens the acceptance region as reported uncertainty increases; the contribution is therefore the conservative policy and its controlled evaluation, rather than the general use of uncertainty in geofencing. Gaftandzhieva and Lyavova [21] present AttendEase, combining time-limited UUID-based QR tokens, duplicate-registration checks, device binding, and a 50-meter Haversine geofence. Their evaluation

involves 48 students and one teacher across five courses and emphasizes operational feasibility. This study is a particularly relevant comparator because it already binds attendance to a session and combines several verification layers. The present contribution is distinguished by the explicit accuracy-adjusted decision rule, controlled FAR-FRR comparisons under alternative error models, and separate tests of token timing and concurrent attendance transitions. This body of work establishes the practical value of combining verification layers. It also narrows the remaining research problem: a new study must contribute more than another implementation of QR plus GPS.

E. Authentication and Transactional Integrity

Attendance credentials also depend on established security and persistence mechanisms. HMAC authenticates messages under a shared secret [22]; URL-safe encoding makes signed values suitable for browser transmission without providing confidentiality [23], [24]. Freshness and one-time use remain separate application properties. Similarly, relational uniqueness protects the key it constrains, whereas lawful updates require transactional state checks. SQLite's isolation and writer behavior [25] are therefore relevant when attendance experiments claim both duplicate-row prevention and duplicate-transition prevention. ACIDRain demonstrates how concurrent web requests can exploit application-level integrity failures despite the use of database transactions [26]. This motivates examining accepted state transitions as well as final row counts. These distinctions inform the present cryptographic specification and the separate insert and check-out race tests.

F. Synthesis and Research Gap

The studies compared in Table I support three conclusions. First, rotating QR codes, geofencing, and device binding are established attendance controls; their simple combination is not, by itself, a new contribution. Second, location quality is acknowledged as an implementation challenge, but it is rarely represented explicitly in the attendance-decision rule. Third, evaluations commonly report functionality or rejection of selected attacks without estimating the competing risks of accepting an unauthorized attempt and rejecting a legitimate one under a common controlled design.

The present study is therefore positioned around protocol binding and measurement. It treats attendance as a server-authoritative conjunction of evidence rather than as an isolated scan. The QR token is bound to a particular session and validity interval; the employee must be authenticated and eligible for that session; the server verifies schedule and attendance state; and the geofence decision is evaluated under alternative treatments of location uncertainty. The study then compares single-context and combined methods using scenario-level acceptance, false acceptance, false rejection, token age, controlled boundary classification, and duplicate-integrity outcomes. This positioning distinguishes the work from a product description and from studies whose principal contribution is the implementation of QR and GPS features.

TABLE I. COMPARISON OF RECENT ATTENDANCE AND GEOFENCING STUDIES

Study	Verification approach	Evaluation emphasis	Gap relative to the present study
Ali et al. (2022) [1]	Systematic review of RFID, QR, barcode, card, and biometric attendance systems.	Technology taxonomy and literature synthesis.	Does not formulate or evaluate a session-bound QR–geofence protocol.
Imanullah and Reswan (2022) [2]	Randomized QR-code attendance scheme.	Low-cost implementation and security motivation.	Motivates changing credentials; the present evaluation separately tests current-token forwarding and expired-token replay.
Nwabuwe et al. (2023) [4]	Dynamic QR, polygonal geofence, login, and IMEI verification.	Proxy-attendance and QR-sharing scenarios.	No comparison of accuracy-adjusted geofence rules or FAR–FRR trade-offs.
Shevchenko and Reips (2024) [10]	Methodological geofencing framework for location-based research.	Implementation, privacy, and empirical geofence studies.	Not an attendance protocol; provides the location-methodology foundation.
Taju et al. (2024) [5]	QR attendance with geolocation validation.	Implementation and functional testing.	Limited treatment of replay, token lifetime, uncertainty, and comparative error rates.
Annurrahma et al. (2025) [14]	Dynamic UUID-based QR with GPS validation.	Black-box testing and user acceptance.	Small evaluation; no common-baseline FAR–FRR or boundary analysis.
Eweoya et al. (2025) [7]	Web-based geofence attendance with user authentication.	Mixed-method requirements and usability evaluation.	No short-lived QR factor or formal uncertainty model.
Siregar et al. (2026) [8]	Dynamic QR plus fixed 30-meter GPS geofence.	Latency, indoor–outdoor drift, and reported fake-GPS rejection tests.	Fixed-radius validation; reported spoofing tests have a limited experimental scope.
Korkortsi et al. (2026) [11]	Cross-platform attendance with device binding, time controls, and an uncertainty-expanded geofence.	Verification time, success rate, and location-error mitigation.	Expands the acceptance boundary; the present study evaluates conservative contraction and controlled FAR–FRR trade-offs.
Gaftandzhieva and Lyavova (2026) [21]	Time-limited session QR tokens, device binding, duplicate checks, and a fixed 50-meter Haversine geofence.	Feasibility testing with one teacher and 48 students across five courses.	Does not compare accuracy-adjusted geofence policies under alternative error distributions or report synchronized attendance-transition races.

III. PROPOSED PROTOCOL AND MATHEMATICAL FRAMEWORK

A. Protocol Overview

The proposed protocol treats attendance as a server-authoritative decision rather than as a result asserted by the employee device. An administrator first creates an attendance session and specifies its active period, eligible employee population, permitted attendance methods, and, when required, the geographic center and radius. While the session remains open, the system publishes signed QR tokens that are bound to the session and its current token version. An authenticated employee scans the displayed token through the employee portal and submits it together with a fresh location measurement and the associated reported accuracy. The server verifies the token, resolves the session, checks employee eligibility and schedule conditions, evaluates the applicable geofence rule, determines

the lawful attendance-state transition, and records either an accepted or rejected decision.

This separation of evidence collection from decision authority is deliberate. The browser obtains the QR token and location measurement, but it is not trusted to decide whether the employee is present, eligible, or within the authorized time window. All policy checks are performed against server-side session data. The implementation retains operational audit evidence, including decision reasons, token fingerprints, reported accuracy, and location outcomes. The evaluation harness separately records controlled token ages and synthetic measurement details. Together these records support operational review and the controlled experiments described later in the article.

B. Threat Model and Trust Assumptions

The threat model considers an employee or external party who may obtain a screenshot of a valid QR code, forward a recently displayed code, replay an expired code, alter a token payload, submit a token issued for a different session, attempt attendance outside the authorized region, or repeat a completed attendance action. It also includes authenticated employees who are outside the session scope or are not expected to work during the relevant period. The adversary may inspect and modify client-side requests because browser code and form fields cannot be treated as trusted enforcement points. In the controlled forwarding scenario, the adversary submits a current valid token from a location outside the configured geofence. The evaluation also includes a compound attack in which the adversary forwards a valid token and fabricates coordinates inside the geofence.

The protocol assumes that the application server, signing secret, database, and administrative account controls are not compromised. It also assumes that the server clock is sufficiently accurate for the selected QR lifetime. Browser-reported location is treated as uncertain contextual evidence rather than as unquestionable proof of physical presence. A user who controls a compromised device may fabricate coordinates. Accordingly, the present web implementation is intended to reduce the success of the proxy-attendance, token-reuse, and invalid-state attacks represented in the declared scenario matrix; it does not claim an absolute guarantee of physical presence or resistance to location spoofing. The compound-attack test directly examines the consequence of that trust limitation; additional independent presence evidence is outside this article's scope.

C. Session-bound Rotating QR Token

Each rotating QR token contains an attendance-session identifier, the current token version, its issue time, an attendance-specific purpose, and a random nonce. The server authenticates these values using a shared secret. Changing the session version revokes previously issued tokens, while the purpose restriction prevents a credential from being accepted for another operation. Authentication protects the token contents against modification; it does not conceal them.

The pinned evaluation uses the ItsDangerous timed-token mechanism with HMAC-SHA-1 authentication [24], [22]. Its browser-compatible encoding provides no confidentiality [23]. The distinction between hash collision resistance and keyed message authentication is discussed in RFC 6194 [27]; the present specification documents the evaluated construction rather than recommending an algorithm for new deployments.

Acceptance requires both authentication and temporal validity. The outer signing timestamp must be no more than 300 seconds old and cannot lie ahead of the receiving server clock. The attendance-specific lifetime is restricted to 10–300 seconds, with a default of 45 seconds. The issue time must fall within that lifetime; an issue time up to 30 seconds ahead is tolerated only when the outer signing timestamp itself remains valid. These are separate checks, so the issue-time tolerance does not provide a general allowance for clock disagreement. Exact expiry is accepted at integer-second precision. The server also verifies that the referenced session exists, its version

remains current, its active period has not ended, and the requested attendance method is enabled. Let

$$Q_i = 1 \{\text{all required token and session checks pass}\}. \quad (1)$$

Server receipt time governs these checks. Refreshing the displayed QR image does not invalidate an older token that remains within its permitted lifetime.

A cryptographically generated 96-bit nonce distinguishes issued tokens. Because the system does not maintain a registry of consumed nonces, this value does not independently prevent forwarding or reuse of a currently valid token. Expiry, version changes, session status, and attendance-state checks provide the corresponding controls. The signing secret is supplied through deployment configuration, and the application refuses the development default in production mode. The historical procedure used to generate the deployment secret was not recovered; the controlled evaluation instead used a deliberately public test secret.

D. Spatial Distance Model

Let (φ_0, ℓ_0) denote the latitude and longitude of the authorized session center, and let (φ_i, ℓ_i) denote the coordinates reported for attendance attempt i . The symbol ℓ is used for longitude to avoid conflict with the uncertainty-weight parameter introduced later. Angles are converted to radians before calculation. Define $\Delta\varphi_i = \varphi_i - \varphi_0$ and $\Delta\ell_i = \ell_i - \ell_0$. The Haversine terms are

$$h_i = \sin^2\left(\frac{\Delta\varphi_i}{2}\right) + \cos(\varphi_0) \cos(\varphi_i) \sin^2\left(\frac{\Delta\ell_i}{2}\right), \quad (2)$$

and

$$d_i = 2R_E \arcsin\left(\sqrt{h_i}\right), \quad (3)$$

where R_E is the mean radius of the Earth and d_i is expressed in meters when $R_E = 6,371,000$ m. For the short distances typically used in workplace geofences, this calculation provides a stable and interpretable measure of separation between the reported position and the authorized center.

E. Uncertainty-aware Geofence Decision

A conventional circular geofence accepts an attempt whenever $d_i \leq R$, where R is the authorized radius. This rule ignores uncertainty associated with the location measurement. The Geolocation specification supplies an accuracy value that represents an estimated horizontal uncertainty radius in meters at a 95% confidence level [9]. This reported value is not independently verified as a calibrated confidence bound for each device. Let a_i denote the reported accuracy. The proposed framework defines an effective distance

$$d_i^*(\lambda) = d_i + \lambda a_i, \quad 0 \leq \lambda \leq 1, \quad (4)$$

where λ controls the weight assigned to reported uncertainty. The binary geofence decision is

$$G_i(\lambda) = \begin{cases} 1, & \text{if } 0 < a_i \leq a_{\max} \text{ and } d_i + \lambda a_i \leq R, \\ 0, & \text{otherwise.} \end{cases} \quad (5)$$

The parameter $a_{\max}$ is the largest reported-accuracy value accepted by the session policy. Setting $\lambda = 0$ recovers the standard rule $d_i \leq R$. Setting $\lambda = 1$ gives the conservative

rule $d_i + a_i \leq R$. Intermediate values produce weighted rules that may offer a more practical balance between unauthorized acceptance and legitimate rejection. The framework does not assume that one value of λ is universally preferable; the controlled study estimates the consequences of that policy choice. The archived location evaluator implements the endpoints $\lambda = 0$ and $\lambda = 1$; intermediate weights are research extensions. By the triangle inequality, if the true position lies within an uncertainty disk of radius a_i around the reported position, $d_i + a_i \leq R$ is sufficient for that disk to lie inside the geofence. This gives the endpoint rule a geometric rationale, conditional on the accuracy radius being meaningful. A fractional weight is a tunable policy margin, not an automatically calibrated presence probability.

For interpretation near the boundary, define the effective margin

$$M_i(\lambda) = R - d_i^*(\lambda). \quad (6)$$

A positive margin indicates that the attempt satisfies the selected geofence rule, a zero margin places the effective location on the boundary, and a negative margin indicates rejection. Recording $M_i(\lambda)$, d_i , and a_i separately makes it possible to distinguish physical distance from uncertainty-induced rejection during subsequent analysis.

F. Operational Eligibility and Attendance State

Location and possession of a valid QR token are not sufficient conditions for attendance. Let A_i indicate that the employee has an authenticated and active portal account. Let S_i indicate that the resolved session is open and that the request falls within its authorized time window. Let W_i indicate that the employee belongs to the session scope and satisfies the applicable work-schedule or exception rules. Finally, let D_i indicate that the requested action is a valid attendance-state transition. For example, a check-in is invalid after a check-in has already been recorded, and a check-out is invalid when no preceding check-in exists or when the employee has already checked out.

When a session does not require location verification, $G_i(\lambda)$ is set to one by policy. Let $Y_i = 1$ denote acceptance and $Y_i = 0$ denote rejection. The complete server decision is summarized by

$$Y_i = 1 \iff Q_i = A_i = S_i = W_i = D_i = G_i(\lambda) = 1. \quad (7)$$

Eq. (7) expresses the central design principle of the protocol in direct logical form: attendance is accepted if and only if every required check passes. A valid token cannot compensate for an inactive session, an authorized location cannot compensate for an ineligible employee, and successful authentication cannot override an invalid attendance state. This conjunctive structure also supports precise rejection codes because the first failed component can be recorded without retaining the raw token or unnecessary personal data.

G. Decision Procedure and Audit Evidence

On receipt of a submission, the server first validates the token signature and extracts the bound session identifier. It then retrieves the current session record and checks the token

version, token age, session lifecycle state, and permitted attendance method. After establishing the employee's identity, session scope, and schedule eligibility, the server evaluates the required location policy using the coordinates and accuracy supplied with that request. The current attendance record is read within the same decision process so that the server can determine whether the lawful transition is check-in, check-out, or rejection. An accepted transition and its associated operational audit events are then committed to persistent storage. The separate checkout update guard and its evaluation are described in section IV.

The evaluation harness records synthetic observations separately from operational attendance. Its exports retain scenario, method, expected outcome, observed outcome, token age, location decision, and declared random seeds where applicable. These are evaluation files, not evidence of a research-observation table in the archived application. Source hashes associate the executed functions with the retained build.

H. Security Properties and Limitations

Subject to the stated trust assumptions, the signed token protects integrity because an altered payload fails server verification unless the signing key is compromised. The age condition in Eq. 1 limits the useful period of a copied code, while the session identifier and token version prevent a valid token from being transferred to another session or retained after administrative revocation. Scope, schedule, and attendance-state checks address authenticated but unauthorized requests and repeated transitions. The geofence term reduces the usefulness of forwarding a current token to a device that reports a position outside the authorized region.

These properties do not make the protocol fraud-proof. A copied QR token may still be used during its valid lifetime by another authenticated employee who can satisfy the remaining conditions. Browser-provided coordinates may also be falsified on a compromised device. In particular, a valid forwarded token submitted with fabricated inside-geofence coordinates passed the controlled component and route tests when the other conditions were satisfied. The contribution is therefore bounded by the evidence supplied to the server: short-lived session binding restricts selected token misuse, while browser coordinates provide conditional spatial context. The evaluation quantifies this reduction and the corresponding cost in legitimate rejection rather than relying on absolute security claims.

Principal notation is summarized in Table II.

TABLE II. PRINCIPAL NOTATION USED IN THE PROPOSED PROTOCOL

Symbol	Meaning
T	Signed rotating QR token.
τ	Maximum permitted token age.
d_i	Haversine distance from the reported position to the authorized center.
a_i	Reported horizontal location accuracy for attempt i .
R	Authorized geofence radius.
λ	Weight assigned to reported location uncertainty.
$M_i(\lambda)$	Effective geofence boundary margin.
$Q_i, A_i, S_i, W_i, D_i, G_i$	Binary variables for token validity, authentication, session validity, work eligibility, attendance-state validity, and geofence compliance.
Y_i	Final server attendance decision.

IV. SYSTEM ARCHITECTURE AND IMPLEMENTATION

Arkos AttendPro is a web-based attendance management platform for workforce operations and organized events. Administrators configure attendance sessions, participant eligibility, permitted verification methods, and location requirements, while attendance records and audit logs support subsequent review. The wider platform accommodates employee attendance and event check-in workflows; the present study evaluates the rotating QR and geofencing protocol in an archived workforce build.

The proposed protocol was implemented in AttendPro as a modular web application. The research baseline uses Python and Flask for the application layer, SQLAlchemy for relational persistence, server-rendered HTML templates for the administrative and employee interfaces, and client-side JavaScript for QR acquisition and location requests. The implementation separates user interaction from the attendance decision: the browser collects evidence, while the server validates that evidence against the current session, employee, schedule, geofence, and attendance state. This separation is central to the protocol because no client-side success state is sufficient to create an attendance record.

A. System Architecture

Fig. 1 summarizes the principal components. The administrator interface defines attendance sessions, selects participating employees or organizational units, associates work schedules, enables attendance methods, and configures location requirements. The session manager maintains the lifecycle and policy of each session. For sessions that permit rotating QR attendance, the QR generator creates short-lived signed tokens bound to the current session and token version. Employees access the attendance workflow through an authenticated portal, where a QR code can be captured with the device camera or supplied as an image. When location evidence is required, the browser geolocation service returns latitude, longitude, and a horizontal accuracy estimate.

The verification engine is the authoritative policy component. It resolves the session referenced by the token, validates the complete eligibility chain, evaluates the selected geofence rule, and determines the lawful attendance transition. The operational database supplies employee, schedule, session, and current attendance information and receives an accepted state transition. Operational session events and audit logs retain evidence associated with accepted and rejected requests.

B. Attendance-session Configuration and Lifecycle

An attendance operation is represented by an explicit session rather than by a globally available check-in endpoint. A session records its date, start and end times, time zone, attendance type, permitted methods, participant scope, linked schedule, location policy, and token configuration. The scope may cover the entire company, selected departments, selected employees, or employees assigned to selected schedules. A session can be configured for check-in, check-out, or an automatic mode that derives the next lawful action from the employee's current attendance record.

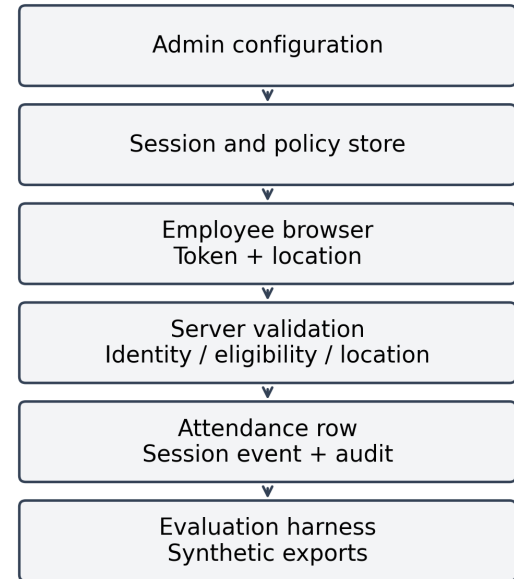


Fig. 1. Arkos AttendPro system architecture. The employee device supplies token and location evidence, but the server-side verification engine makes the attendance decision.

The implementation uses the lifecycle state set

$$\mathcal{L} = \{\text{Draft, Scheduled, Open, Paused, Closed, Expired, Cancelled}\}. \quad (8)$$

Only an open session whose current server time lies within the configured interval can authorize attendance. Closing or expiring a session increments its token version, thereby invalidating QR tokens created under the preceding version. Scheduled sessions may open automatically at their start time, and active sessions may close automatically after their end time. Time comparisons use UTC-normalized values, while user-facing dates and times are rendered in the configured company or session time zone.

Representative administrative interfaces are shown in Fig. 2. The session-creation workflow exposes the fields that bind the time window, participant scope, enabled attendance methods, and location policy. The attendance-station view displays the session-bound rotating QR credential together with session status and summary counters. These interfaces support configuration and evidence presentation, but they do not bypass server-side validation. The screenshots illustrate the implemented workflow and are not themselves experimental observations.

C. Rotating QR Generation and Employee Evidence Acquisition

The rotating QR service generates a token only when the referenced session is open, within its active time window, and configured to permit rotating QR attendance. Token construction and validation follow the specification in section III. The server-held signing secret is not encoded in the displayed QR image, and a valid authenticated token alone does not authorize attendance.

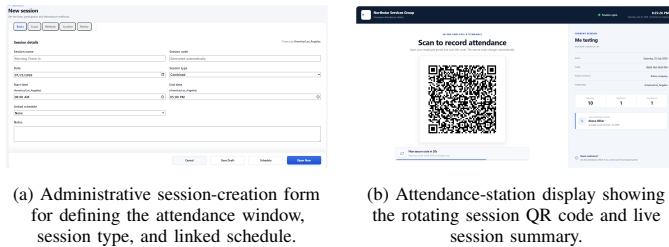


Fig. 2. Representative administrative interfaces from the implemented system.

The employee portal requires an authenticated account linked to an active employee record. QR acquisition is supported through a live camera stream and an image-upload fallback. The scanner first attempts decoding in the browser and may submit an image frame to the application decoder when a browser-native decoder is unavailable. The recovered token is not treated as proof of attendance; it is one item of evidence submitted to the verification endpoint.

Location is requested only when required by the session or company attendance policy. The portal requests a fresh high-accuracy reading and transmits the reported latitude, longitude, and accuracy with the token. It also includes a CSRF token in the request. The attendance endpoint does not accept a client-provided assertion that the device is inside the geofence. Instead, it recomputes the distance and policy outcome on the server from the submitted measurements and the stored session configuration.

Fig. 3 presents representative employee-facing mobile screens. The home dashboard exposes the current attendance state and whether an eligible session is open. The scan interface provides the camera-based QR workflow and communicates both successful and terminal attendance states. In particular, the completed-attendance screen illustrates the state-validation control that prevents another transition after check-in and check-out have already been recorded. The client makes the evidence-acquisition and response sequence visible to the user, but the server remains the decision authority.

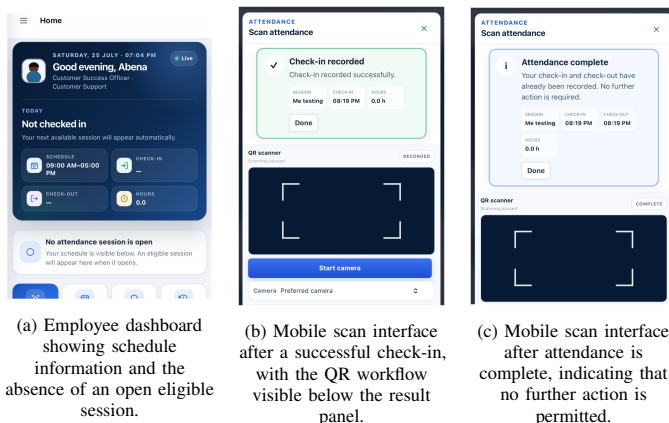


Fig. 3. Representative employee-facing mobile interfaces.

D. Server-authoritative Verification Workflow

Fig. 4 presents the decision sequence. Token validation precedes any attendance mutation. The server verifies the token signature, purpose, payload structure, session identifier, issue time, version, lifetime, and enabled attendance method. A token that is malformed, altered, expired, issued implausibly in the future, revoked, or associated with a closed or unavailable session is rejected with a normalized reason code.

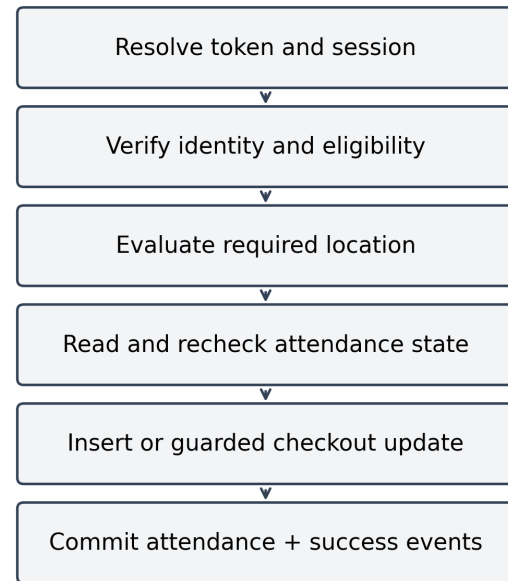


Fig. 4. Server-authoritative attendance-verification workflow. Every failed decision branch produces a rejected outcome and structured rejection evidence.

After the token resolves to an eligible session, the server confirms that the employee account is active, the employee belongs to the session scope, an applicable work schedule exists, the employee is expected to work on the session date, and any explicitly linked schedule matches. The server then applies the session-specific location rule described in section III. A missing, non-finite, non-positive, or insufficiently precise accuracy reading causes rejection when accuracy is required. For a valid reading, the server calculates d_i , forms $d_i^*(\lambda)$, and compares the result with the stored radius.

The final stage evaluates the current attendance state. A check-in is rejected when a check-in already exists, and a check-out is rejected when no preceding check-in exists or the record is already complete. The application requests a row-level lock when reading the attendance record and repeats the state checks before modification. The operational employee-date uniqueness constraint provides a final barrier against competing inserts. SQLite does not implement the requested row-level lock, however, and row uniqueness cannot prevent two updates to an existing record. The controlled integration tests therefore examined both check-in and check-out races rather than inferring complete transition integrity from uniqueness alone.

The archived check-out path permitted two simultaneous

successful updates and two success audit events. A separate local revision adds an atomic conditional update: a row can acquire a check-out time only if its check-in exists, its check-out is still null, and it is not finalized. Exactly one affected row permits success processing; a competing zero-row update causes rollback and a concurrency response. The revised and archived sources are retained separately. This revision was evaluated locally and was not deployed to the live service.

The accepted route flushes the attendance mutation, creates its session and audit events, and commits them together. An integrity exception causes rollback and a concurrency response; other exceptions in the mutation block cause rollback and an error response. Rejection auditing follows a separate path. The archived connection hook requests foreign keys, WAL mode, a 10-second busy timeout, and normal synchronization. The local integration connection reported these WAL and timeout settings. WAL permits readers and a writer to proceed concurrently, while write operations remain serialized [28]. The deliberately fail-fast 50 ms insert pilot is a different configuration. SQLite serializes writers, but transaction labels alone do not supply an effective row lock [25].

E. Operational Attendance Records

The operational attendance record stores the employee and date, check-in and check-out timestamps, resolved work schedule, authorizing attendance session, and attendance source. When geofencing is applied, it also stores the submitted coordinates, reported accuracy, calculated distance, and geofence outcome for operational review. Schedule-based metrics are recalculated after an accepted transition so that lateness and attendance status remain consistent with the employee's assigned schedule.

The implementation distinguishes a rotating session QR token from an employee badge QR. A rotating QR identifies a short-lived attendance session and is presented by an authenticated employee through the portal. An employee badge is a longer-lived opaque credential used by an authorized staff-operated scanner to identify the employee within a selected session. Both paths ultimately apply session, schedule, scope, and attendance-state controls, but they remain separate experimental methods because their possession and replay characteristics differ. Fig. 5 shows the administrative view through which employee identity and badge status are managed.

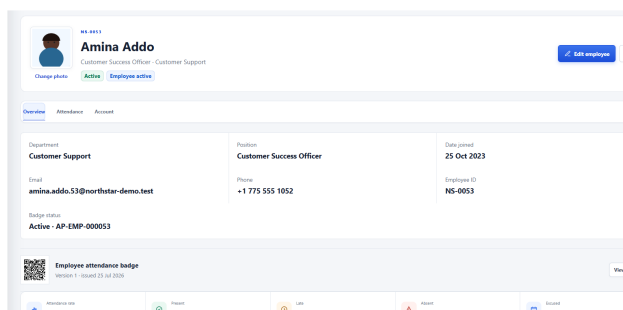


Fig. 5. Administrative employee-profile view showing the issued attendance badge, badge status, and summary attendance indicators.

F. Operational Audit and Evaluation Exports

Operational session events record the method, outcome, reason code, action, token fingerprint, and relevant location results. The raw QR token is not written to the operational audit tables; a truncated cryptographic fingerprint is used for correlation. Accepted and rejected outcomes are both retained so that experimental analysis is not restricted to successful transactions.

The fresh evaluation used pseudonymous synthetic identifiers and source hashes, with expected decisions kept separate from observed outcomes. The archived application does not contain the research-observation schema or experiment controls described in earlier manuscript versions. Accordingly, claims about frozen experiments and atomic research-table writes are not attributed to this build. Operational records and synthetic evaluation exports serve different purposes.

G. Application Security Controls

The implementation applies authenticated sessions, inactivity timeouts, role-based permissions [29], and CSRF validation [30] to administrative and employee operations. Attendance pages are returned with no-store caching directives. Response headers restrict framing and declare camera and geolocation permissions for the application origin. The application also refuses to start in production mode with the default development secret.

These controls protect the web workflow and the integrity of the stored decision process; they do not establish that browser-reported coordinates are physically authentic. The current system therefore treats location as contextual evidence. Device attestation, mock-location detection, and device binding are reserved for a future native mobile implementation and are not included in the claims evaluated in this study.

V. CONTROLLED EXPERIMENTAL METHODOLOGY

A. Study Design and Evidence Provenance

The evaluation combined source-level component execution, local application-route integration, synthetic location-error simulation, and an insert-contention pilot. The archived SkillFest source dated 7 September 2026 was recovered from an application backup. The original manuscript's research harness, random seed, raw observations, and exact research-build identity were not recovered. Earlier reconstruction scripts repeated manifest outcomes and simplified token rules; they were not retained as implementation evidence. The present results instead arise from fresh executions; the authors retain the associated source hashes, input definitions, code, and generated observations. This provenance prevents the new stochastic realizations from being presented as the unavailable original data.

The token, eligibility, badge, Haversine, and location functions were extracted through Python's abstract syntax tree without rewriting their bodies. Component tests used synthetic database responses and schedule records, a public test-only secret, and a controlled server clock. These fixtures isolate decision functions but do not test authentication, database persistence, or schedule resolution. The integration cases therefore

loaded the archived application and exercised its actual attendance endpoints and database models and a temporary SQLite database. Authentication was supplied through a synthetic logged-in session; CSRF was disabled by the archived testing configuration. No participants, production records, browser captures, live requests, or deployed changes were involved. A separate patched source was used only to evaluate the atomic check-out revision.

B. Comparison Methods and Scenario Execution

The four evidence configurations are summarized in Table III. Eligibility components remain common, but the badge-scanner and employee self-service paths are different workflows. The comparison concerns controlled evidence requirements rather than identical end-user interactions.

TABLE III. EVIDENCE CONFIGURATIONS IN THE COMPONENT COMPARISON

Method	Credential	Location requirement
M_1	Static employee badge	None
M_2	Rotating session token	None
M_3	No rotating token	Required
M_4	Rotating session token	Required

The legacy scenario definitions supplied 44 applicable method–scenario cells, each repeated 200 times. Four additional compound-attack cells combined current credentials with fabricated inside-geofence coordinates. The resulting 9,600 observations were produced by constructing inputs, executing the extracted functions, and comparing actual outcomes with separately stored expectations. Fresh tokens and badge identifiers were generated where applicable. A requested-session mismatch condition in the component harness was identified as a harness guard, because the production rotating validator itself resolves the session bound to its token. Schedule and identity fixtures remained explicit limitations of this comparison.

Repetition checked consistency across constructed instances; it did not create independent users or a sampled population of adversarial attacks. Consequently, deterministic component outcomes were reported as scenario coverage and acceptance counts, without Wilson intervals or a cross-method aggregate security ranking. The shared closed-session, duplicate-check-in, and invalid-check-out cases were directly comparable across all four configurations. Credential-specific scenarios were retained as additional coverage. An accepted compound attack was counted as an observed weakness, even when acceptance agreed with the expected behavior of the trust model.

C. Signed-token and Timing Experiments

Real signed credentials were generated and verified at lifetimes $\{15, 30, 60, 120\}$ seconds. For each lifetime, the integer ages were $\{0, \tau - 1, \tau, \tau + 1, \lfloor 1.5\tau \rfloor\}$. Two methods and three contexts—legitimate inside use, forwarding with an outside report, and forwarding with fabricated inside coordinates—were evaluated with 100 freshly issued credentials per cell. This yielded 12,000 observations. Exact expiry and the next integer second were both included.

A further 252 deterministic cases crossed the four lifetimes with receipt delays $\{0, 1, 2, 5, 10, 15, 30, 60, 120\}$ seconds and receiving-server clock offsets $\{-31, -30, -1, 0, 1, 30, 31\}$ seconds. Both the serializer timestamp and payload time checks were executed under the controlled clock. These were sensitivity settings, not observed network-delay distributions. Additional tests covered wrong keys, purpose, malformed fields, nonexistent sessions, revocation, disabled methods, closed sessions, and future payload issue times. The public test vector was reproduced independently from the recorded bytes.

D. Controlled Geofence Simulation

The geofence analysis was a Monte Carlo sensitivity study, not a real-device GPS experiment. A circular geofence with radius $R = 100$ m was evaluated at true signed boundary offsets

$$\mathcal{B} = \{-20, -10, 0, 10, 20\} \text{ m}, \quad (9)$$

where positive values denote locations inside the boundary. Reported-accuracy levels were

$$\mathcal{A} = \{5, 15, 30\} \text{ m}, \quad (10)$$

and the uncertainty weights were

$$\Lambda = \{0, 0.25, 0.50, 0.75, 1.00\}. \quad (11)$$

For a selected offset $b_i \in \mathcal{B}$, the known true radial distance was

$$d_{\text{true},i} = R - b_i. \quad (12)$$

Thus, $b_i > 0$ represents a point inside the geofence, $b_i < 0$ represents a point outside it, and $b_i = 0$ represents the boundary. In the primary model, the signed radial measurement error was generated as

$$e_i \sim \mathcal{N}\left(0, \left(\frac{a_i}{2}\right)^2\right) \quad \text{conditional on} \quad -a_i \leq e_i \leq a_i, \quad (13)$$

and the measured radial distance was

$$d_i = d_{\text{true},i} + e_i. \quad (14)$$

The error in Eq. 13 is signed: negative values move the reported position toward the centre, whereas positive values move it away from the centre. It is therefore a one-dimensional radial measurement model, not a two-dimensional Gaussian model for latitude and longitude and not a half-normal error magnitude. The truncation prevents the simulated radial error from exceeding the selected accuracy value in absolute magnitude. This distribution was chosen as a controlled sensitivity setting; it was not estimated from a particular browser or device population.

For each accuracy–offset cell, 1,000 errors were generated with seed 20260924. Each measurement was reused across the five policies; the 75,000 decision rows therefore represent 15,000 generated positions, not 75,000 independent positions. One thousand draws per cell limit the approximate worst-case 95% Monte Carlo margin for an estimated cell proportion to about 3.1 percentage points. Each resulting measured position was evaluated under all five values of λ , yielding

$$3 \times 5 \times 1000 \times 5 = 75,000 \quad (15)$$

observations. FAR and FRR were calculated from physically outside and physically inside offsets, respectively. The exact-boundary point was excluded from the primary FAR–FRR table because equality at $d = R$ is policy-sensitive; it was retained in the complete raw dataset.

For interpretation, balanced error was defined as

$$\text{BER}(\lambda) = \frac{\text{FAR}(\lambda) + \text{FRR}(\lambda)}{2}. \quad (16)$$

The simulation was intended to expose the direction and magnitude of the security–usability trade-off. The truncated radial distribution is a declared sensitivity model rather than a model calibrated from particular phones, browsers, or buildings. It was not used to claim that any selected value of λ is optimal across physical environments.

E. Alternative Location-error Models

Two additional models were used to determine whether the principal FAR–FRR pattern depended on the one-dimensional radial construction. For these models, a bearing θ_i was sampled uniformly from $[0, 2\pi)$ and the true position was represented in local east–north coordinates as

$$x_i = d_{\text{true},i} \sin \theta_i, \quad y_i = d_{\text{true},i} \cos \theta_i. \quad (17)$$

After generating an error pair $(E_{x,i}, E_{y,i})$, the simulated browser position became

$$\tilde{x}_i = x_i + E_{x,i}, \quad \tilde{y}_i = y_i + E_{y,i}. \quad (18)$$

The displaced point was converted to latitude and longitude by a spherical forward calculation from a fixed centre. The AST-extracted archived Haversine function then recomputed d_i from the centre and the simulated browser coordinates. Consequently, the alternative models supplied the same form of latitude, longitude, and accuracy input that the browser supplies to the application, while preserving the known simulated truth needed to calculate FAR and FRR.

In the two-dimensional Gaussian model,

$$E_{x,i}, E_{y,i} \stackrel{\text{iid}}{\sim} \mathcal{N}(0, \sigma_{a_i}^2), \quad \sigma_{a_i} = \frac{a_i}{\sqrt{-2 \log(0.05)}} \approx \frac{a_i}{2.448}. \quad (19)$$

The resulting radial error is Rayleigh distributed, and this scale makes a_i its 95th percentile. The heavy-tailed model used

$$E_{x,i} = s_{a_i} T_{x,i}, \quad E_{y,i} = s_{a_i} T_{y,i}, \quad T_{x,i}, T_{y,i} \stackrel{\text{iid}}{\sim} t_3, \quad (20)$$

where

$$s_{a_i} = \frac{a_i}{q_{0.95}}, \quad q_{0.95} = 4.36618. \quad (21)$$

Here, $q_{0.95}$ is the numerically estimated 95th percentile of $\sqrt{T_x^2 + T_y^2}$ for independent standard t_3 variables. This calibration also places 95% of radial errors within a_i , but the distribution remains unbounded and permits occasional large displacements.

The robustness analysis retained the same radius, offsets, accuracy levels, uncertainty weights, and 1,000 draws per accuracy–offset cell. It was repeated under 30 seeds for all three error models. Each seed therefore generated 45,000 measured positions and 225,000 geofence decisions, giving

1.35 million measured positions and 6.75 million decisions overall. FAR, FRR, and BER were first calculated within each seed; the reported robustness results are means across the 30 seeds, accompanied by standard deviations in the manuscript. These standard deviations describe variability across simulated realizations, not uncertainty for real devices. The primary realization and the robustness summaries both belong to the fresh evaluation reported here. Only the first seed’s full robustness observations are retained; all 30 per-seed summaries and their seeds are available.

The inherited Student radial quantile was independently checked with three seeds and two million error pairs per seed. Coverage at the stated constant was compared with 0.95 using the Bernoulli Monte Carlo standard error. Numerical checks additionally evaluated points 1 mm inside, on, and 1 mm outside each effective threshold. Exact-boundary floating-point outcomes were retained without introducing a new tolerance into the decision rule.

F. Application Integration and Transition Races

The integration cases exercised the actual attendance routes with a synthetic active employee, assigned flexible schedule, open workforce session, and location requirement. Outcomes included whether the request created a new attendance transition, committed attendance rows, success session events, and audit entries. A successful response was not treated automatically as a newly recorded transition: a request concerning an already completed action could receive a successful acknowledgment without changing the attendance record.

Sequential cases tested valid check-in, outside location, repeated check-in, checkout without check-in, valid checkout, repeated checkout, and a current token with fabricated centre coordinates. Injected failures at success-event creation checked rollback after the attendance mutation had been flushed or conditionally updated. Two concurrent workers were then synchronized immediately after the final attendance-state query returned. This deliberately constructed interleaving examined competing check-in and check-out operations. Archived and locally patched sources were evaluated separately; the test was a targeted race check rather than a throughput benchmark. The connection reported WAL and a 10-second busy timeout.

G. Separate SQLite Insert Pilot

The insert pilot used concurrency levels $\{10, 25, 50, 100, 250\}$, ten bursts per level, and both same-employee and distinct-employee requests, giving 100 bursts. It created a minimal unique employee–date table with WAL and a deliberately short 50 ms timeout. Completed inserts, duplicate violations, lock errors, other operational errors, and synchronization failures were counted separately. Committed row counts were queried after each burst. This experiment tested insert uniqueness and contention, not the complete application transaction or check-out behavior.

H. Reproducibility and Ethics

The authors retain archived and patched source hashes, pinned dependencies, input definitions, synthetic observations, seed summaries, code, public cryptographic test vectors, and

integration outcomes. Geofence FAR and FRR were recomputed from generated observations, and robustness means were checked against per-seed summaries. The retained research records exclude deployment secrets and production user databases. All identities and observations used in these experiments were synthetic.

VI. RESULTS

A. Component Decisions and Common Scenarios

All 9,600 component outcomes agreed with their separately declared expectations. The 44 legacy cells accounted for 8,800 observations; the four compound-attack cells added 800. Agreement here refers to the executed functions and synthetic fixtures, not to a complete attack-resistance claim. In particular, expected acceptance of an attack remained acceptance of an attack.

Fig. 6 displays observed acceptance by applicable scenario. The common closed-session, duplicate-check-in, and invalid-check-out cases were rejected in all 200 repetitions for each method. Every valid baseline was accepted. Modified or revoked credentials were rejected where applicable. Rotating-token expiration was enforced, while the requested-session mismatch component case was rejected by the explicit harness guard. Static badges remained usable in another valid session when the selected session otherwise permitted attendance.

The selected contrasts in Table IV avoid aggregation over different attack sets. A current forwarded credential with an outside location was accepted by M_1 and M_2 and rejected by M_4 . When the location was fabricated at the centre instead, all four configurations accepted the constructed request. The M_3 compound case tests fabricated location with authenticated eligibility; it has no rotating credential requirement.

B. Signed-token Lifetime and Clock Sensitivity

The 12,000 signed-token observations admitted ages 0, $\tau - 1$, and τ , and rejected $\tau + 1$ and $\lfloor 1.5\tau \rfloor$ for each tested lifetime. Fig. 7 shows that before expiry M_2 accepted all three location contexts, whereas M_4 rejected the outside report but admitted the fabricated inside report. These observations executed the timed serializer and session validator, rather than a normalized-age formula alone.

The 252 clock-delay cases changed decisions at the same receipt-time boundaries. For a 15-second token, a 15-second delay with zero clock offset remained valid; a 30-second delay did not. A receiving clock one second behind at zero delay failed the serializer's future-timestamp check. Separately signed payloads with an issue time 30 seconds ahead passed when their outer signing timestamp was current; 31 seconds ahead failed. Wrong-key, wrong-purpose, malformed-field, missing-session, revoked-version, disabled-method, and closed-session cases were rejected. Both library-based and independent standard-library reconstructions reproduced the retained token vector exactly.

C. Primary Geofence Realization

Table V, Fig. 8 report the fresh radial-error realization. At 15 m accuracy, $\lambda = 0$ gave FAR 4.15% and FRR 3.70%; $\lambda = 0.25$ gave 0.90% and 9.55%; and $\lambda = 0.50$ gave 0.00%

and 20.30%. At $\lambda = 1$, FRR was 50.15%. At 30 m accuracy, the corresponding endpoint FAR–FRR pairs were 15.45%–15.30% and 0.00%–84.80%. Each FAR or FRR cell used 2,000 eligible outside or inside measurements. Exact-boundary positions were excluded from these summaries (Fig. 8).

At 5 m accuracy all non-boundary measurements were classified correctly under each weight. Fig. 9 shows that the fixed-distance rule had the lowest balanced error for 15 m and 30 m in this realization. The complete raw file retains the boundary positions and all shared-measurement policy decisions.

D. Alternative Error Models and Seed Variability

Table VI reports mean and between-seed standard deviation at 15 m accuracy. At $\lambda = 0.50$, mean FAR was 0.00%, 0.10%, and 0.34% for the radial, Gaussian, and independent-coordinate Student models; the respective mean FRRs were 19.38%, 18.91%, and 14.97%.

Independent calibration and numerical boundary checks supported the stated error scale and classification behavior.

E. Application-route and Transaction Outcomes

The actual route rejected an outside-location request without an attendance row, recorded a valid check-in, and prevented a sequential duplicate from creating a new transition. Checkout without check-in made no attendance row. A valid checkout completed the existing row, and a repeated request produced no additional success transition. Injected failures at success-event creation rolled back check-in insertion and check-out mutation. In the latter case the original check-in remained and check-out was null.

The compound request carrying a valid current token and fabricated centre coordinates was accepted and committed an attendance row, one success session event, and one audit entry. The local route therefore reproduced the limitation observed in the component test.

The synchronized check-in race produced one recorded response and one concurrency rejection, with a single employee-date row and success event. The archived checkout race instead produced two recorded responses and two success events for one row. Under the separate atomic-update revision, the check-out race produced one recorded response, one concurrency rejection, one success session event, and one audit entry. These are targeted two-worker results under the retained interleaving; they are not production-scale reliability estimates.

F. Separate Insert-contention Pilot

Each same-employee burst committed exactly one row at every concurrency level. At level 250, the 10 bursts produced 10 inserts, 2,054 duplicate violations, and 436 lock errors. No other operational or barrier errors were recorded. Distinct-employee lock-error rates were 0%, 0%, 0%, 0.50%, and 32.12% at concurrency 10, 25, 50, 100, and 250, respectively, as shown in Fig. 10. Queried row counts agreed with completed inserts. These values describe the fresh local 50 ms pilot.

Valid baseline	100%	100%	100%	100%
Forwarded: outside report	100%	100%		0%
Modified credential	0%	0%		0%
Expired token		0%		0%
Revoked credential	0%	0%		0%
Requested-session mismatch*	100%	0%		0%
Missing QR		0%		0%
Outside geofence			0%	0%
Outside scope	0%	0%	0%	0%
No schedule	0%	0%	0%	0%
Closed session	0%	0%	0%	0%
Poor accuracy			0%	0%
Duplicate check-in	0%	0%	0%	0%
Invalid check-out	0%	0%	0%	0%
Fabricated inside report	100%	100%	100%	100%
	M1	M2	M3	M4

* Rotating requested-session mismatch includes a harness guard.

Fig. 6. Observed component acceptance. Blank cells are not applicable. Compound forwarding uses fabricated inside-geofence coordinates.

TABLE IV. ACCEPTED COMPONENT REQUESTS PER 200 REPETITIONS FOR SELECTED SCENARIOS

Scenario	M_1	M_2	M_3	M_4
Valid baseline	200	200	200	200
Closed session	0	0	0	0
Duplicate check-in	0	0	0	0
Invalid check-out	0	0	0	0
Forwarded current credential, outside report	200	200	N/A	0
Current credentials, fabricated inside report	200	200	200	200

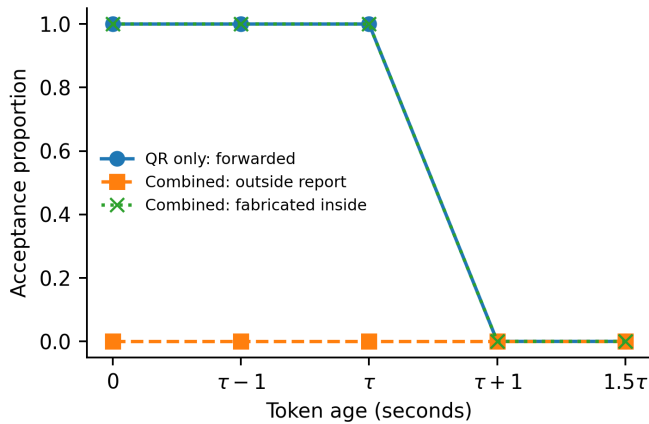


Fig. 7. Signed-token acceptance by age and supplied location context. The points aggregate the retained repetitions across lifetimes.

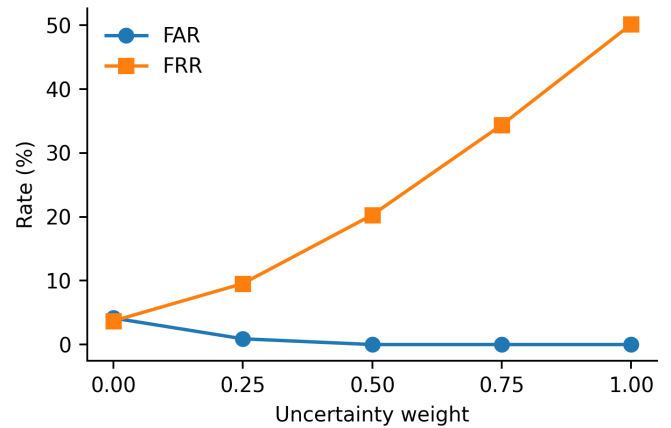


Fig. 8. Fresh primary FAR and FRR at 15 m reported accuracy.

TABLE V. FRESH PRIMARY GEOFENCE RATES (PERCENT), EXCLUDING EXACT-BOUNDARY POSITIONS

λ	Accuracy 15 m			Accuracy 30 m		
	FAR	FRR	BER	FAR	FRR	BER
0.00	4.15	3.70	3.92	15.45	15.30	15.38
0.25	0.90	9.55	5.22	5.30	30.50	17.90
0.50	0.00	20.30	10.15	1.20	49.50	25.35
0.75	0.00	34.40	17.20	0.00	69.00	34.50
1.00	0.00	50.15	25.07	0.00	84.80	42.40

TABLE VI. FAR AND FRR AT 15 M ACCURACY: MEAN (STANDARD DEVIATION), IN PERCENT, OVER 30 SEEDS

λ	Truncated radial		2D Gaussian		Independent t_3 coordinates	
	FAR	FRR	FAR	FRR	FAR	FRR
0.00	3.68 (0.50)	3.66 (0.32)	2.52 (0.34)	2.78 (0.43)	1.69 (0.36)	2.05 (0.28)
0.25	0.57 (0.17)	9.37 (0.56)	0.56 (0.18)	8.38 (0.68)	0.72 (0.20)	5.21 (0.44)
0.50	0.00 (0.00)	19.38 (0.84)	0.10 (0.07)	18.91 (0.68)	0.34 (0.13)	14.97 (0.73)
0.75	0.00 (0.00)	33.71 (0.90)	0.02 (0.03)	34.01 (0.87)	0.20 (0.10)	33.94 (0.63)
1.00	0.00 (0.00)	50.12 (1.04)	0.00 (0.00)	51.23 (0.76)	0.11 (0.08)	50.54 (0.65)

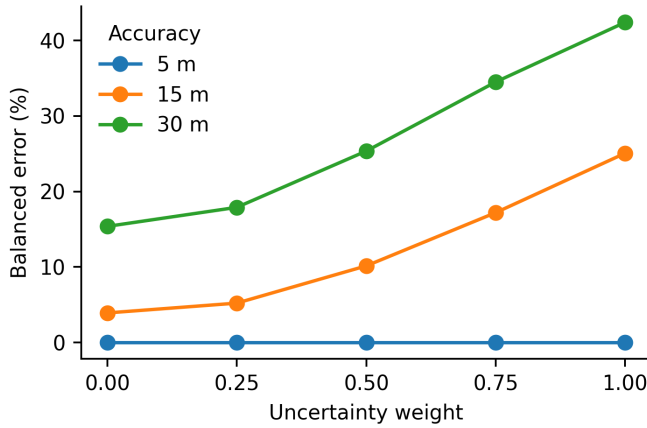


Fig. 9. Balanced error in the fresh primary simulation.

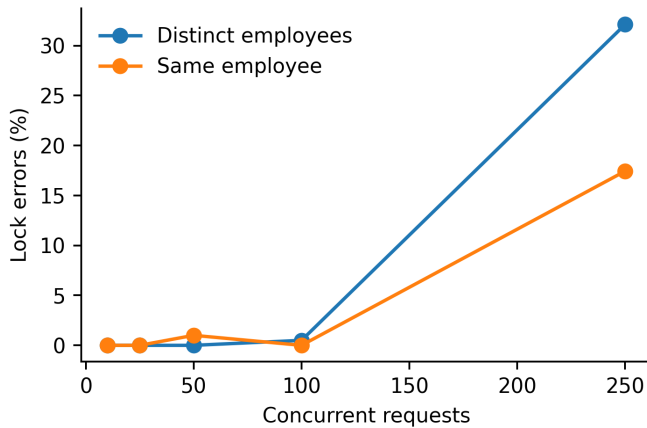


Fig. 10. Lock-error rates in the fresh 50 ms SQLite insert pilot. Duplicate constraint violations are reported separately.

VII. DISCUSSION

The findings distinguish credential authenticity, contextual eligibility, and evidence of physical presence. Timed authentication and session versioning restricted the tested token misuse, but possession of a current authenticated token remained transferable. Location verification rejected the remote report only when the submitted coordinates actually placed the request outside. A fabricated inside report passed the tested route. Shorter expiry reduces the time available for forwarding; it cannot establish that an accepted coordinate is genuine. The observed compound attack therefore limits the security claim without removing the operational value of session binding.

The uncertainty rule has a similarly conditional interpretation. For nonnegative accuracy, acceptance regions shrink as λ increases. FAR cannot increase and FRR cannot decrease on the same measurements simply because of this nesting. Agreement on that direction across error models is consequently expected from the policy algebra. The informative simulation findings concern magnitudes, tail behavior, and the cost of conservatism. Bounded radial errors yielded zero FAR in several cells, whereas the unbounded alternatives retained occasional false acceptance. Neither a zero observed rate nor a larger margin supplies a general guarantee.

The fixed rule's lower balanced error in the primary realization also prevents a universal preference for the conservative rule. Equal weighting of FAR and FRR is one criterion; an organization that assigns higher cost to false acceptance may select a different operating point. The intermediate weights provide a sensitivity framework rather than calibrated probabilities. Real-device measurements would be needed to connect reported accuracy to a deployment-specific policy.

The transaction results separate row integrity from transition integrity. Employee-date uniqueness prevented competing inserts but did not prevent duplicate checkout updates and audits. The conditional update addressed the tested race without relying on SQLite row locks. Its benefit is specific to that revision and test; the archived baseline and patched behavior

must not be conflated. Similarly, the fail-fast insert pilot and application integration used different busy timeouts and cannot be combined into a production-capacity claim.

Relative to the studies in Table I, the contribution lies in making these bindings and limits explicit. Dynamic QR and geofencing are established controls, but evaluation must separate authenticated bytes, server policy, uncertain measurements, fabricated evidence, and persistent state transitions. The controlled evaluation supports this distinction through executable components, local route outcomes, synthetic error models, and retained provenance.

VIII. LIMITATIONS, FUTURE WORK, AND CONCLUSION

A. Limitations

The original research harness and raw observations were unavailable. This article therefore reports a fresh evaluation of an archived source and a separately identified local checkout revision, rather than claiming recovery of the original stochastic realizations. Historical deployment key generation and dependency versions were not established; the cryptographic description refers to the pinned test environment and recovered source settings.

Component tests used synthetic database responses and schedule records. Integration tests exercised actual application endpoints and database operations, but used a simulated request interface, a supplied authenticated session, disabled cross-site request forgery protection, synthetic records, and local SQLite. They did not measure browser scanning, location acquisition, TLS, real network latency, login security, production throughput, or another database backend. The synchronized races selected one relevant two-worker interleaving and do not prove absence of all concurrency failures. No recruited-user usability or accessibility evaluation was performed.

The geofence distributions were theoretical sensitivity settings, not estimates from smartphone measurements. The radial model bounded errors by the chosen accuracy, while alternative models assigned approximately 95% radial coverage and unbounded tails. Browser accuracy may be miscalibrated and coordinates may be fabricated. The accepted compound attack demonstrates this trust limitation directly. Seed variability quantifies Monte Carlo variation, not environmental or device uncertainty. Only a representative robustness seed is retained in full, alongside all per-seed summaries and reproducible generators.

B. Future Work

Real-device studies should measure inside, boundary, and outside positions across phones, browsers, and environments, compare reported accuracy with observed error, and record end-to-end delay. Additional independent presence evidence may address stronger forwarding and spoofing attacks, but it is outside the present contribution. Further transaction testing should cover administrative races, retries, intended production backends, and sustained concurrent requests. A deployment study can then connect the policy trade-off with usability and operational requirements.

C. Conclusion

The study specifies and evaluates a server-authoritative attendance protocol combining authenticated session tokens, active time windows, employee eligibility, attendance state, and geofence context. Fresh component executions and local route tests demonstrated selected token and policy checks while exposing the limits of fabricated location evidence. A compound forwarding request passed when supplied with an inside-geofence report, so the protocol cannot be interpreted as proof of physical presence.

Synthetic location experiments quantified the cost of uncertainty margins: conservative policies reduced false acceptance but increased legitimate rejection, with numerical rates depending on the error model. The fixed rule had lower balanced error in the primary realization, whereas higher weights reflected a different risk preference. A separate transaction analysis showed that row uniqueness and lawful update integrity require distinct controls; the local conditional checkout revision closed the tested race.

The resulting contribution is an auditable account of session binding, measurement uncertainty, and persistent state, supported by retained code, observations, specifications, and provenance. Its claims remain bounded by the controlled experiments and provide a basis for subsequent device and deployment validation.

FUNDING

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

DECLARATION OF COMPETING INTEREST

Smaila Amoanu is the designer and developer of Arkos AttendPro and may have a commercial interest in the software. The controlled evaluation used synthetic data and was not influenced by an external sponsor. Bright Wiredu Nuakoh and Stephen Asiedu declare no competing interests.

DATA AVAILABILITY

The data and code supporting this study are available from the corresponding author upon reasonable request.

ETHICS STATEMENT

The study used synthetic employees and sessions and involved no recruited human participants or production employment records.

ACKNOWLEDGMENTS

The first author gives thanks to God for the strength, wisdom, and perseverance to complete this research, drawing encouragement from Philippians 4:13: "I can do all things through Christ which strengtheneth me" (KJV). The authors thank the reviewers for their constructive comments, which helped improve the clarity and rigor of the manuscript.

DECLARATION ON GENERATIVE AI

ChatGPT assisted with drafting, language editing, simulation code, and result checks; the authors take responsibility for the final work.

REFERENCES

- [1] Nabeel Salih Ali, Ahmed Hazim Alhilali, Hasanein D. Rjeib, Haider Alsharqi, and Basheer Al-Sadawi. Automated attendance management systems: Systematic literature review. *International Journal of Technology Enhanced Learning*, 14(1):37–65, 2022.
- [2] Muhammad Imanullah and Yuza Reswan. Randomized QR-code scanning for a low-cost secured attendance system. *International Journal of Electrical and Computer Engineering*, 12(4):3762–3769, 2022.
- [3] David M'Raihi, Salah Machani, Mingliang Pei, and Johan Rydell. TOTP: Time-based one-time password algorithm. Technical Report 6238, Internet Engineering Task Force, May 2011.
- [4] Augustine Nwabuwe, Baljinder Sanghera, Temitope Alade, and Funmiyi Olajide. Fraud mitigation in attendance monitoring systems using dynamic QR code, geofencing and IMEI technologies. *International Journal of Advanced Computer Science and Applications*, 14(4):938–945, 2023.
- [5] S. W. Taju, Y. P. Mamahit, and J. A. Pongantung. Implementing QR code and geolocation technologies for the student attendance system. *Cogito Smart Journal*, 10(1):221–232, 2024.
- [6] M. Tamboli, G. Katore, S. Patale, and N. J. Philips. Attendance management system using geofencing technology. In *2024 First International Conference on Pioneering Developments in Computer Science and Digital Technologies (IC2SDT)*, pages 137–141. IEEE, 2024.
- [7] Ibukun Eweoya, O. J. Adeniyi, A. O. Awoniyi, E. Mgbuehuruike, J. O. Adewuyi, T. O. Adigun, and Y. A. Mensah. Design and implementation of a university attendance management system using geo-fencing. *Asian Journal of Computer Science and Technology*, 14(1):28–46, 2025.
- [8] Rosma Siregar, Bagoes Maulana, Muhammad Isnaini, Elsa Sabrina, and Harvei Desmon Hutahaean. Performance evaluation of a mobile attendance system using dual-factor dynamic QR code and GPS geofencing. *Jurnal Multimedia dan Teknologi Informasi (Jatilima)*, 7(06):1126–1134, 2026.
- [9] World Wide Web Consortium. Geolocation. W3c candidate recommendation snapshot, World Wide Web Consortium, March 2026. 26 March 2026; accessed 10 October 2026.
- [10] Yury Shevchenko and Ulf-Dietrich Reips. Geofencing in location-based behavioral research: Methodology, challenges, and implementation. *Behavior Research Methods*, 56(7):6411–6439, 2024.
- [11] Wisdom Elikplim Korkotsi, Edmund Dasori Azundow, Ramatu Al-Hassan, Yaw Amankrah Sam-Okyere, Samuel Kafui Kwawukume, and Mohammed Izzu-deen Zakariah. A cross-platform mobile attendance system utilizing geofencing and multi-factor authentication. *Applied Engineering, Innovation, and Technology*, 3(1):15–31, 2026.
- [12] Fadi Masalha and Nael Hirzallah. A students attendance system using QR code. *International Journal of Advanced Computer Science and Applications*, 5(3):75–79, 2014.
- [13] Katharina Krombholz, Peter Frühwirt, Peter Kieseberg, Ioannis Kapsalis, Markus Huber, and Edgar Weippl. QR code security: A survey of attacks and challenges for usable security. In *Human Aspects of Information Security, Privacy, and Trust (HAS 2014)*, volume 8533 of *Lecture Notes in Computer Science*, pages 79–90. Springer, 2014.
- [14] Kholifah Dwi Annurrahma, Sayekti Harist Suryawan, and Abdul Rahim. Design of a meeting attendance system based on dynamic QR code with universally unique identifier (UUID v4). *Jurnal Riset Informatika*, 7(4):336–346, 2025.
- [15] Paul Syverson. A taxonomy of replay attacks. In *Proceedings of the Computer Security Foundations Workshop VII (CSFW7)*, pages 187–191. IEEE, 1994.
- [16] Paul A. Zandbergen and Sean J. Barbeau. Positional accuracy of assisted GPS data from high-sensitivity GPS-enabled mobile phones. *Journal of Navigation*, 64(3):381–399, 2011.
- [17] Krista Merry and Pete Bettinger. Smartphone GPS accuracy study in an urban environment. *PLoS ONE*, 14(7):e0219890, 2019.
- [18] Hannes Tschofenig, Henning Schulzrinne, and Bernard Aboba. Trustworthy location. Technical Report 7378, Internet Engineering Task Force, December 2014.
- [19] Shinan Liu, Xiang Cheng, Hanchao Yang, Yuanchao Shu, Xiaoran Weng, Ping Guo, Kexiong Zeng, Gang Wang, and Yaling Yang. Stars can tell: A robust method to defend against GPS spoofing attacks using off-the-shelf chipset. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 3935–3952. USENIX Association, 2021.
- [20] Stefan Saroiu and Alec Wolman. Enabling new mobile applications with location proofs. In *Proceedings of the 10th Workshop on Mobile Computing Systems and Applications (HotMobile '09)*, pages 1–6. ACM, 2009.
- [21] Silvia Gaftandzhieva and Gabriela Lyavova. AttendEase: Fraud-resistant attendance tracking in higher education using dynamic tokens and geofencing. *International Journal of Advanced Computer Science and Applications*, 17(9):230–242, 2026.
- [22] Hugo Krawczyk, Mihir Bellare, and Ran Canetti. HMAC: Keyed-hashing for message authentication. Technical Report 2104, Internet Engineering Task Force, 1997.
- [23] Simon Josefsson. The base16, base32, and base64 data encodings. Technical Report 4648, Internet Engineering Task Force, 2006.
- [24] Pallets Projects. Itsdangerous 2.2 documentation. Accessed 10 October 2026; version 2.2.0 released 16 April 2024; see Signing Interface, Signing With Timestamps, and URL Safe Serialization.
- [25] SQLite. Isolation in sqlite. Accessed 10 October 2026.
- [26] Todd Warszawski and Peter Bailis. ACIDRain: Concurrency-related attacks on database-backed web applications. In *Proceedings of the 2017 ACM International Conference on Management of Data (SIGMOD '17)*, pages 5–20, Chicago, IL, USA, 2017. ACM.
- [27] Tim Polk, Lily Chen, Sean Turner, and Paul Hoffman. Security considerations for the SHA-0 and SHA-1 message-digest algorithms. RFC 6194, IETF, March 2011.
- [28] SQLite. Write-ahead logging. Accessed 10 October 2026.
- [29] Ravi S. Sandhu, Edward J. Coyne, Hal L. Feinstein, and Charles E. Youman. Role-based access control models. *Computer*, 29(2):38–47, 1996.
- [30] Adam Barth, Collin Jackson, and John C. Mitchell. Robust defenses for cross-site request forgery. In *Proceedings of the 15th ACM Conference on Computer and Communications Security (CCS '08)*, pages 75–88, Alexandria, VA, USA, 2008. ACM.

APPENDIX

GEOFENCE DATA-GENERATING PROCESS AND ALGORITHM

This appendix records the complete algebraic path from a controlled true position to the final geofence classification. It is included to distinguish variables used only to generate synthetic observations from variables available to the deployed application.

A. Truth, Measurement, and Decision

Let (ϕ_0, ℓ_0) denote the geofence centre and let R denote its radius. The simulation first selects a signed boundary offset b_i and defines

$$d_{\text{true},i} = R - b_i. \quad (22)$$

The value b_i is known to the simulation and determines the ground truth. It is not supplied to AttendPro. A positive value indicates a true inside position, while a negative value indicates a true outside position.

For the two-dimensional models, a uniformly distributed bearing θ_i places this true distance in a local tangent coordinate system:

$$x_i = d_{\text{true},i} \sin \theta_i, \quad y_i = d_{\text{true},i} \cos \theta_i. \quad (23)$$

The selected error model produces $(E_{x,i}, E_{y,i})$, after which

$$\tilde{x}_i = x_i + E_{x,i}, \quad \tilde{y}_i = y_i + E_{y,i}. \quad (24)$$

Define the measured radial displacement and bearing by

$$\tilde{r}_i = \sqrt{\tilde{x}_i^2 + \tilde{y}_i^2}, \quad \tilde{\theta}_i = \text{atan2}(\tilde{x}_i, \tilde{y}_i). \quad (25)$$

With Earth radius R_E , a spherical forward calculation converts $(\tilde{r}_i, \tilde{\theta}_i)$ to the simulated browser coordinate $(\tilde{\phi}_i, \tilde{\ell}_i)$:

$$\tilde{\phi}_i = \arcsin \left[\sin \phi_0 \cos \delta_i + \cos \phi_0 \sin \delta_i \cos \tilde{\theta}_i \right], \quad (26)$$

$$\tilde{\ell}_i = \ell_0 + \text{atan2} \left(\sin \tilde{\theta}_i \sin \delta_i \cos \phi_0, \cos \delta_i - \sin \phi_0 \sin \tilde{\phi}_i \right), \quad (27)$$

where $\delta_i = \tilde{r}_i / R_E$ and all angles in the calculation are in radians. AttendPro then applies the Haversine formula to the centre and reported coordinate:

$$h_i = \sin^2 \left(\frac{\tilde{\phi}_i - \phi_0}{2} \right) + \cos \phi_0 \cos \tilde{\phi}_i \sin^2 \left(\frac{\tilde{\ell}_i - \ell_0}{2} \right), \quad (28)$$

$$d_i = 2R_E \arcsin \sqrt{h_i}. \quad (29)$$

The uncertainty-adjusted distance and decision are

$$d_i^*(\lambda) = d_i + \lambda a_i, \quad G_i = \mathbf{1}\{0 < a_i \leq a_{\max}, d_i^*(\lambda) \leq R\}. \quad (30)$$

All simulated accuracies were positive and below the configured $a_{\max} = 50$ m. This appendix describes the spatial decision alone, rather than the complete attendance acceptance

conjunction. Only $(\tilde{\phi}_i, \tilde{\ell}_i, a_i)$ and the stored policy parameters are needed by the application. The values b_i , $d_{\text{true},i}$, and the generated errors exist only in the simulation. Comparing G_i with the known true status identifies a false acceptance or false rejection.

B. Simulation Algorithm

For each seed, error model, accuracy level, and signed offset, the robustness procedure was:

- 1) Compute $d_{\text{true}} = R - b$ and record whether the true position is inside or outside.
- 2) Draw a bearing $\theta \sim \text{Uniform}(0, 2\pi)$.
- 3) Construct the true local coordinates (x, y) .
- 4) Generate the location error according to the selected radial, Gaussian, or Student- t model.
- 5) Form the measured local coordinates $(\tilde{x}, \tilde{y})$.
- 6) Convert the measured displacement to a simulated browser latitude and longitude.
- 7) Apply the Haversine function to obtain the measured distance d .
- 8) For each $\lambda \in \Lambda$, calculate $d^*(\lambda) = d + \lambda a$ and accept when $d^*(\lambda) \leq R$.
- 9) Compare the application decision with the known truth and update the FAR and FRR counts.

This separation is important. The data-generating process creates controlled browser-like inputs and retains their truth; the application logic receives only the simulated measurement and makes the same form of decision used in deployment.