COMPUTOLOGY: Journal of Applied Computer Science and Intelligent Technologies
Volume 6, Issue 1, Jan-Jun 2026, pp. 52-72
DOI: 10.17492/computology.v6i1.2603
https://www.journalpressindia.com/cjacsit
© 2026 Journal Press India

Design and Implementation of a Geolocation-based Student Attendance Management System

Michael F. Edafeajiroke, Amanda E. Ekata** and Promise Eyindah****

ABSTRACT

This study presents the design, implementation, and evaluation of a geolocation-based student attendance management system designed to overcome key limitations of existing methods, including proxy attendance, hardware dependency, and single-point validation. The system employs a dynamic geofencing mechanism in which a lecturer initiates a session and defines a virtual boundary. To be marked present, students must remain within this boundary for at least 80% of the class duration, a rule enforced through periodic location checks every 20 minutes. Developed with a Flutter frontend for cross-platform mobile access and a Node.js/PostgreSQL backend, the solution features role-specific dashboards for students, lecturers, and administrators, real-time notifications via Firebase Cloud Messaging, and automated report generation. Performance testing under simulated academic scenarios demonstrated high reliability, with 94.3% accuracy in classroom environments and excellent user satisfaction, yielding a System Usability Scale (SUS) score of 78.5. The dwell-time verification logic effectively ensures sustained physical presence, thereby promoting accountability and engagement. The study concludes that this software-only, intelligent system offers a significant advancement in attendance management, though future work should address scalability under extreme loads and inherent GPS technology limitations to achieve universal robustness

Keywords: *Geolocation, attendance system; GPS tracking; Mobile application; Node.js; Flutter.*

1.0 Introduction

Attendance tracking plays a critical role in the academic performance and administrative monitoring of students in educational institutions. Bailke *et al*. (2024).

**Corresponding author; Assistant Lecturer, Computer Science Department, University of Port Harcourt, River State, Port Harcourt, Nigeria (E-mail: edafemichaelfavour@gmail.com)*

***Final Year Student, Computer Science Department, University of Port Harcourt, River State, Port Harcourt, Nigeria (E-mail: amandaekata@gmail.com)*

****Senior Lecturer, Computer Science Department, University of Port Harcourt, River State, Port Harcourt, Nigeria (E-mail: promise.enyindah@uniport.edu.ng)*

As learning environments become more data-driven and technologically integrated, there is a growing demand for innovative systems that automate and improve the reliability of attendance tracking. Most institutions have relied on manual methods such as roll-calls, paper sheets, and physical registers. However, these methods are highly susceptible to manipulation, human error, and inefficiency (Sandhya *et al.,* 2025). To address these inefficiencies, digital and automated solutions have been proposed and implemented in various forms. Among them, geolocation-based systems have emerged as a promising approach. These systems use location data from GPS-enabled mobile devices to verify a student's physical presence within a defined area (geofence) before marking their attendance.

Despite the fact that recent innovations have shown to be promising, they have not yet delivered fully dynamic and adaptable solutions. Most rely on static geofence boundaries, single-point validation, or classroom entry timestamps without tracking continued presence during lectures (Chauhan *et al.*, 2024; Kundariya *et al.*, 2025). As a result, these systems may inaccurately reflect student participation. Additionally, many lack robust analytics or reporting tools to help educators evaluate attendance trends or intervene with disengaged students early (Sandhya *et al*., 2025). There is, therefore, a pressing need for a cost-effective, scalable, cross-platform, and intelligent attendance management system. One that not only verifies the student's location in real-time but also ensures they remain engaged throughout the session. Such a solution must minimize the risks of proxy attendance, reduce dependency on costly hardware, and empower stakeholders with meaningful analytics (Jeevarathinam & Janani, 2025).

The goal is not to force attendance but to promote accountability and meaningful participation, ensuring that students are present, attentive, and acquiring the knowledge needed for personal growth and national development.

1.1 Aim and objectives

- This study aims to design and implement a Geolocation-based student attendance management system.

The Specific Objectives are to:

- Design a user-friendly mobile interface using Figma and define a scalable database schema with PostgreSQL to support user management, attendance tracking, and geolocation data storage.
- Implement the mobile application front-end using Flutter and Dart and develop a secure backend system using Node.js to handle user authentication, geolocation validation, attendance session control, and communication with the database.

- Test the system through simulations of real-world academic scenarios to evaluate geolocation accuracy, session tracking reliability, and the effectiveness of proxy attendance prevention.
- Evaluate the system's usability, performance, and effectiveness through user's feedback, functional testing, and analysis of attendance data accuracy and reporting capabilities.

2.0 Review of Literature

The integration of mobile technologies into educational environments has revolutionized traditional classroom operations, specifically through the development of automated attendance management systems. While historical methods like roll calls are prone to errors and manipulation, institutions have increasingly adopted automated models involving QR codes, RFID, biometrics, and GPS technology. Among these, GPS-based models are preferred for their scalability and reduced hardware dependence.

Several studies have attempted to integrate geolocation with other technologies to curb attendance fraud. Sandhya *et al.* (2025) developed a hybrid system combining QR codes with GPS verification, yet this model remained vulnerable to proxy attendance as students could share QR code screenshots remotely. Similarly, Jeevarathinam & Janani (2025) utilized the Haversine formula to measure proximity to a classroom, but their reliance on static geofences failed to adapt to venue changes, and the system only recorded entry rather than sustained presence. In contrast, the system proposed in this study eliminates reusable QR codes and introduces flexible geofencing, ensuring attendance is only validated if a student remains within the lecturer's radius for 80% of the session.

Limitations regarding system flexibility and platform compatibility are prevalent in existing literature. Eweoya *et al.* (2025) utilized geofencing linked to registration data but were constrained by fixed-radius designs that could not adapt to dynamic schedules. Kundariya *et al.* (2025) proposed a real-time geofencing model that efficiently captured presence but was limited to Android devices and fixed location spans. Addressing these shortcomings, the implemented system utilizes the Flutter framework to ensure cross-platform compatibility for both Android and iOS, while allowing lecturers to dynamically adjust geofence radii.

Further research by Chauhan *et al.* (2024) introduced a low-cost mobile solution using Firebase, yet it logged attendance based solely on campus entry without tracking actual class participation, undermining student accountability. Additionally, web-based Location-Based Service (LBS) implementations by Syamaidzar & Sutopo (2024); Nazara & Nasien (2024) lacked mechanisms to detect GPS spoofing or ensure scalability. The system developed in this study overcomes these issues by enforcing app-level location permissions, performing periodic 20-minute checks to verify dwell time, and generating detailed Excel reports for analytics.

Biometric and hardware-based systems have also been explored but present distinct challenges. Bailke *et al.* (2024) and Ramon *et al.* (2024) combined geolocation with facial and fingerprint recognition, which introduced high implementation costs, privacy concerns, and potential hardware failures. Similarly, Valdez *et al.* (2024) used RFID technology, which allowed for card-sharing and failed to verify physical presence beyond the scanner's range. By focusing solely on intelligent geolocation validation without biometric hardware, the proposed system remains cost-effective and privacy-conscious while maintaining high reliability. Finally, facial recognition integrations by Shahab & Sarno (2020) and Babatunde *et al.* (2022) offered improvements over manual methods but often lacked user-friendly interfaces and real-time notification features. These systems typically check proximity only at sign-in rather than throughout the class. The system implemented here fills these gaps by providing a modern interface with real-time class reminders, dashboard analytics for students and lecturers, and continuous "dwell-time" tracking to ensure genuine academic engagement.

2.1 Study gaps and improvements

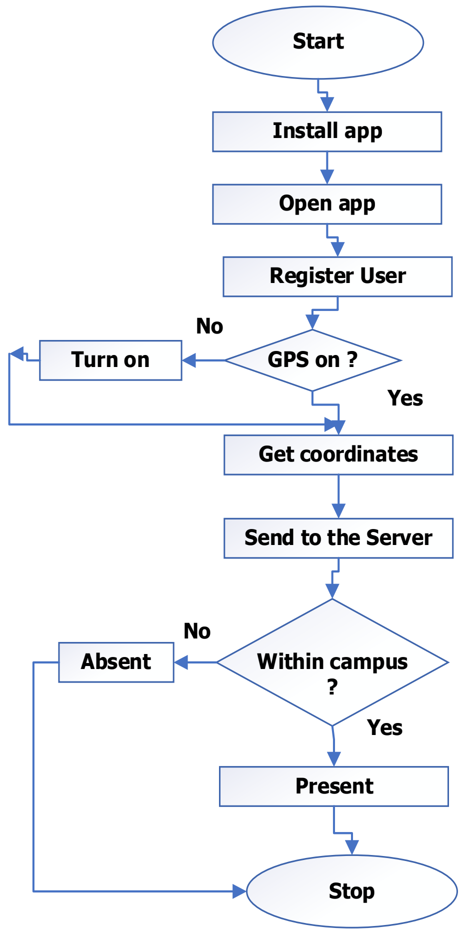
The review of related literature highlights persistent limitations in existing attendance systems, particularly regarding proxy attendance, static geofencing, and single-point validation. Systems relying on QR codes or RFID are susceptible to sharing, while biometric solutions incur high costs and privacy risks. Furthermore, many GPS models use rigid boundaries that fail to adapt to changing venues and only log attendance upon entry, ignoring premature departures. The developed system addresses these gaps by implementing a dynamic, cross-platform solution that validates attendance through continuous monitoring, ensuring students remain present for the majority of the lecture, thereby fostering greater accountability and transparency

3.0 Research Methodology

3.1 Study design

The system design employed a client-server architecture connected through a Firebase real-time database. Students interacted with the client-side application on mobile devices, while Firebase handled cloud data synchronization and server logic. The design followed a linear workflow. In the design, students' login details and coordinates were validated by the system server, which subsequently stored attendance status. The workflow of the mobile-based geolocation attendance system that utilizes geolocation and Firebase Cloud Storage, adapted from Chauhan *et al.* (2024), is presented in Figure 1.

Figure 1: Workflow of a Mobile-based Geolocation Attendance System



Source: Chauhan et al., 2024

3.2 Workflow diagram of the proposed study

The implemented system introduces a multi-layered, modular design that addresses the shortcomings of the earlier model. It is structured into three key layers: frontend, backend, and database, all connected through RESTful APIs for seamless data exchange. The frontend, built with Flutter, provides a unified interface for students, lecturers, and administrators. The backend, developed with Node.js using the Fastify framework, manages authentication, session control, and geolocation verification. The PostgreSQL database serves as a robust relational storage engine for attendance data, user profiles, and course information. Real-time notifications and reminders are managed through Firebase Cloud Messaging (FCM).

3.3 Algorithm and flow of the implemented system

The operational flow of the implemented geolocation-based attendance management system follows a structured series of interactions between the lecturer, students, and backend services. Each stage from class activation to final attendance recording ensures accuracy, transparency, and data security while minimizing manual intervention.

3.3.1 System activation and lecturer configuration

The process begins when the lecturer is notified via a push alert approximately one hour and ten minutes before the scheduled class time. This first notification serves as a confirmation of availability, allowing the lecturer either to proceed with or postpone the session. Once confirmed, students receive an automated notification indicating whether the class will be held. If the class proceeds, the lecturer can either manually start attendance or allow the system to initiate it automatically at the exact class start time, depending on the configuration. When a lecturer initiates a new attendance session through the mobile application. Upon activation, the system automatically retrieves the lecturer's current GPS coordinates (latitude and longitude) using the mobile device's location sensor. These coordinates represent the center point of the geofence. The system simultaneously retrieves the expected class size from the course settings, which determines the radius of the geofence.

3.3.2 Student location acquisition and verification

When attendance is activated, students must grant location access and maintain an active internet connection. Their mobile devices begin transmitting real-time coordinates to the backend at defined intervals. Each student's latitude and longitude are retrieved and compared to the lecturer's position to calculate the straight-line distance between both points. The system employs the Haversine formula, a well-known trigonometric function for determining the great-circle distance between two points on the Earth's surface. The computation is as expressed in equation 1.

$$d = 2r \times \arcsin\left(\sqrt{\sin^2\left(\frac{\Delta \text{lat}}{2}\right) + \cos(\text{lat}_1)\cos(\text{lat}_2)\sin^2\left(\frac{\Delta \text{long}}{2}\right)}\right) \quad ...1$$

where r is the Earth's radius (≈ 6371 km), and Δlat and Δlong represent coordinate differences.

3.3.3 Periodic location checks and dwell-time logic

To ensure that students are genuinely present and not merely passing through the geofence area, the system performs periodic location validations every 20 minutes throughout the session. Each check recalculates the student's distance from the lecturer's device coordinates. The dwell-time rule requires that a student remain within the lecturer-defined radius for at least 80 percent of the total class duration. If at any checkpoint the student's device moves beyond the allowed distance, the system marks a temporary absence flag. Conversely, consistent proximity within the geofence reinforces the student's attendance status. This repeated validation eliminates the possibility of marking attendance for students who leave midway through a session or attempt to manipulate GPS data.

3.3.4 Final validation and attendance recording

At the end of the class, the backend aggregates all location logs and determines the final attendance status. If the computed distance for a student remained within the geofence radius for ≥ 80 percent of the lecture duration, the system automatically updates the PostgreSQL database, marking the student as "Present." Otherwise, the record is set to "Absent." Once attendance is confirmed, all temporary geolocation data is immediately

discarded from memory to protect user privacy. The verified results are then synchronised with the database and reflected on the lecturer's dashboard. The lecturer can view real-time attendance summaries, download session data in .xlsx (Excel) format, or manually adjust entries in exceptional cases such as device malfunction or late arrival. This structured workflow ensures accuracy, transparency, and real-time monitoring. It also demonstrates the implemented system's robustness in automating attendance management while preserving privacy and maintaining academic accountability.

3.4 System Design

The system was divided into two major parts. The Frontend (Client-side) and the Backend (Server-side). The system will be conceptualized with three core actors: students, lecturers, and administrators, each with a clear interaction path and access level. The design focused on a responsive, user-centric mobile interface, backend logic for authentication and data flow, and administrative control dashboards. The UI/UX (user interface) was designed using Figma. This design contained the complete framework of the application, which made it easier to build the front-end of the application.

3.4.1 Frontend development

The frontend was developed using Flutter, allowing cross-platform deployment for both Android and iOS. Flutter was chosen due to its expressive UI capabilities and single codebase, which reduced development time and effort. Key frontend tasks included designing user dashboards for each role, creating location permission prompts and visual indicators, and displaying class schedules, attendance statistics, and reminders. The front-end will also integrate the backend logic, ensuring functionality with various API and endpoints. The app features role-based UI for students, lecturers, and admins with login/register screens. It will entail a map interface or GPS prompt for location verification, Real-time attendance records, and reminders in the form of notifications for upcoming lectures. The app also features downloadable reports for lecturers and admins.

3.4.2 Backend development

The backend was implemented using Node.js and the Fastify framework, which handles authentication, course management, session logging, and real-time communication with the frontend. It will integrate secure user authentication with JWT tokens, storage and retrieval of user and class metadata, attendance record generation and session validation and also, RESTful APIs for communication between frontend and backend. The backend is responsible for handling all core logic of the application

The following components were implemented:

- *User authentication using JWT and OAuth2:* instead of using Firebase Authentication, the system employed JSON Web Tokens (JWT) and OAuth2 protocols for secure, stateless user authentication. When a user logs in, the server generates a JWT that is stored securely on the client side and used for authenticating future requests. This ensures secure session handling without needing to maintain session state on the server. OAuth2 provides secure access delegation and supports integration with external identity providers, if needed.
- *Role-based access control:* The system recognizes multiple roles: students, lecturers, and administrators. Each role has specific permissions and API access restrictions enforced through middleware in the Node.js backend. For instance, only lecturers can initiate attendance sessions, while students are restricted to marking attendance.
    - *Geolocation validation logic:* For the Geolocation verification process, When a student attempts to mark attendance, the app retrieves the current GPS coordinates(latitude and longitude data) from both the student’s device and the lecturer’s device, sends the coordinates to the backend via a secured API, then the backend will compare the student’s location with the geofence coordinates i.e the defined radius proximity to the lecturer’s device location. If the student is within the acceptable radius (e.g., 50 metres), for a period of about 80% of the class, attendance is marked. The backend then updates the record in the database and

sends a confirmation to the app. The system checks the student's proximity to the lecturer every 20 minutes to implement this.

- *Attendance management:* Once geolocation and time validity checks are passed, the backend updates the attendance record in the cloud database. Lecturers can retrieve, view, and modify these records via protected endpoints.
- *Database operations:* A relational database (PostgreSQL) will be used to store structured data such as user profiles, attendance logs, course information, and class session metadata. The Node.js backend securely performs Create, Read, Update, and Delete (CRUD) operations
- *Notifications and session handling:* The backend will be capable of sending out class reminders and alerts using a scheduling library or third-party messaging service. Notifications can be triggered based on class start time or attendance events by Firebase Cloud Messaging (FCM)
- *API security:* All endpoints will be secured using token-based middleware, which verifies JWT signatures, checks token expiry, and ensures only authorised roles can access specific resources.

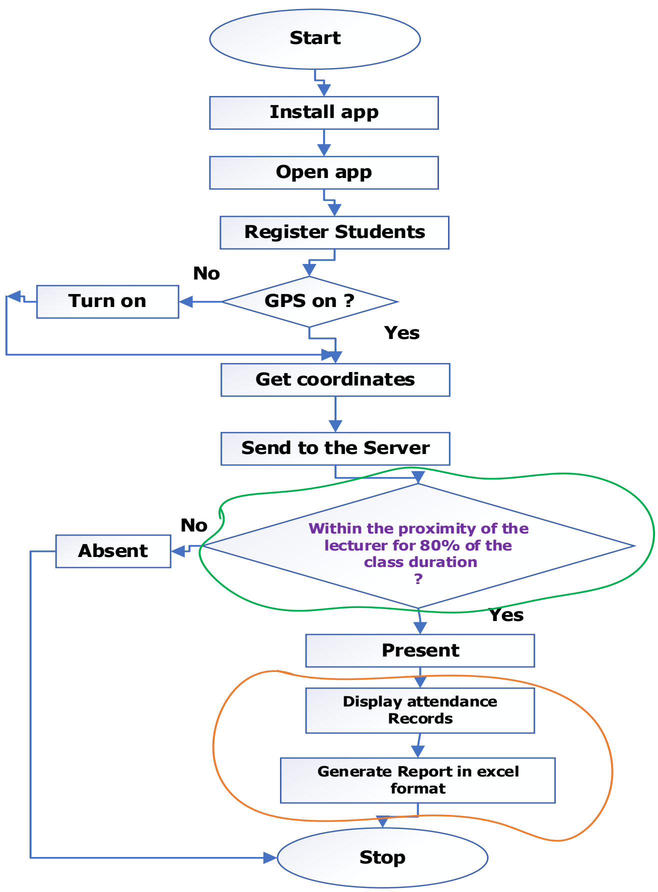
3.4.3 Location verification & attendance logic

Location data from mobile devices will be accessed using GPS sensors and verified against a predefined geofence radius set by lecturers. The student must remain within this radius for at least 80% of the class duration before attendance is marked. The app will use real-time timers and background location tracking, ensuring that only students physically present during class are validated. After the verification, the location is no longer tracked, and the data is discarded. Figure 2 below shows the workflow of the system.

3.5 Features extraction from the dataset

The data for this study were generated through simulated real-time testing within the application environment rather than collected from external sources. The data consisted of GPS coordinates, timestamps, student and lecturer identification details, attendance session logs, and course information. For each attendance instance, the dataset captured: Student ID (unique Identifier), Lecturer ID, course ID, and session time, Latitude and longitude values. Calculated proximity (in meters), Attendance status (present/absent) during testing, and the lecturer's coordinates (latitude and longitude) served as the reference point. For example, when the lecturer's position was recorded at latitude 6.9°N and longitude 4.98°E, the system calculated the positional differences of each student's device to determine whether it fell within a 25-metre radius of the lecturer's location, if the class size is set to small The proximity check was implemented using the Haversine formula, which measures the great-circle distance between two geographic coordinates. If the calculated distance was less than or equal to 25 metres, the student's device was considered within range. This validation process ran continuously at 20-minute intervals throughout the class duration, ensuring that only students who remained within the defined radius for at least 80% of the lecture were marked as present.

Figure 2: Workflow Diagram of the Developed Geolocation-based Attendance System



Source: Created by authors

3.6 Hardware and software requirements

The system was developed and tested on a laptop with macOS, a 16 GB RAM configuration, and a Quad-core processor. Testing was conducted using multiple Android and iPhone smartphones with GPS and internet connectivity to simulate both student and lecturer roles.

4.0 Result and Discussion

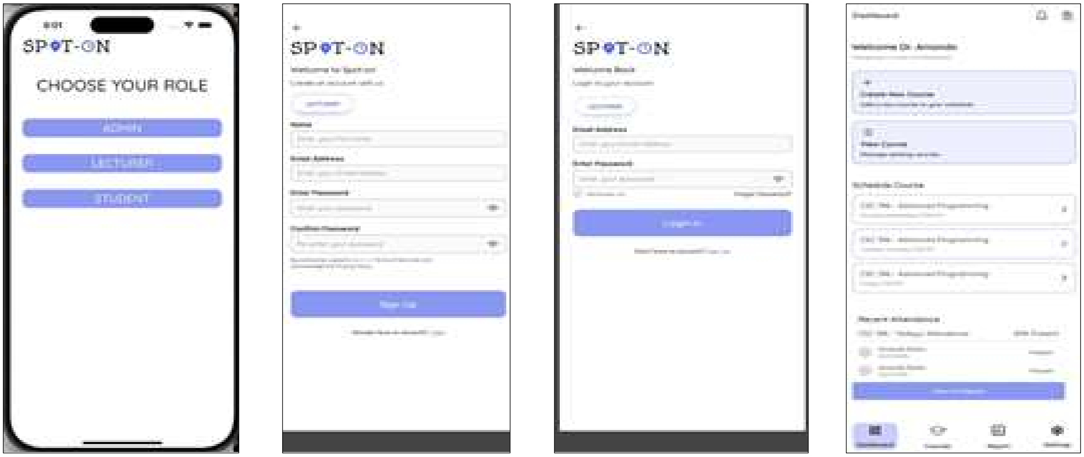
This session presents the outcomes of the system design and implementation process. The results reflect the functional behavior of the developed geolocation-based student attendance management system, focusing on geolocation accuracy, system responsiveness, and user experience during testing. The analysis discusses how the system components worked together to actualize the earlier state objectives. Figure 3 below shows the onboarding screens of the application. Once the app is launched, the user gets to view a series of user-friendly descriptions about the application

Figure 3: Onboarding Screens of the Application

Source: Results obtained from the experiment conducted with the developed system

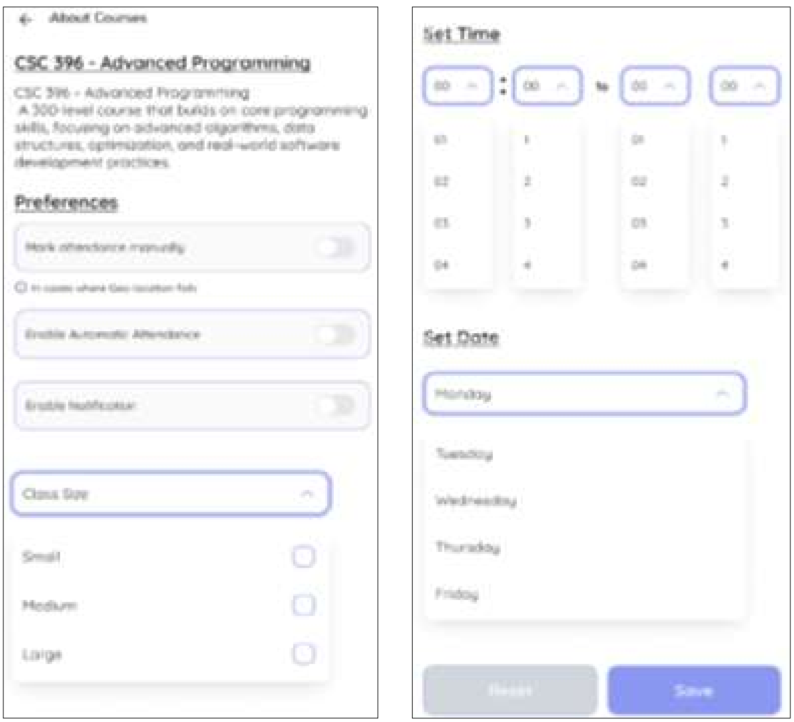
On choosing a role, the lecturer, student, and admin routes to their specific sign-up or sign-in screen. Figure 4 below shows the lecturer's in-app flow. When the lecturer signs in, he accesses his dashboard, which provides comprehensive access to various functionalities. From this central hub, lecturers can create new courses, review existing ones, manage profile settings, access notifications, view their course schedule, and see a summary of recent attendance records. The bottom navigation bar also shows that the lecturer can view their courses, the full attendance report, and navigate to their settings. Figure 5 below shows the About Courses screen, where the lecturer can select a registered course and edit some details about it. For example, setting the time and date, number of occurrences, toggles to enable manual and automatic attendance, and the ability to set class size, which carries a fixed geofence based on the class size picked. This easy-to-understand design makes it possible for the lecturer to always come back and adjust the class size if need be, for a dynamic feel. It also gives the lecturer full control.

Figure 4: Lecturer's Login Flow and Dashboard



Source: Results obtained from the experiment conducted with the developed system

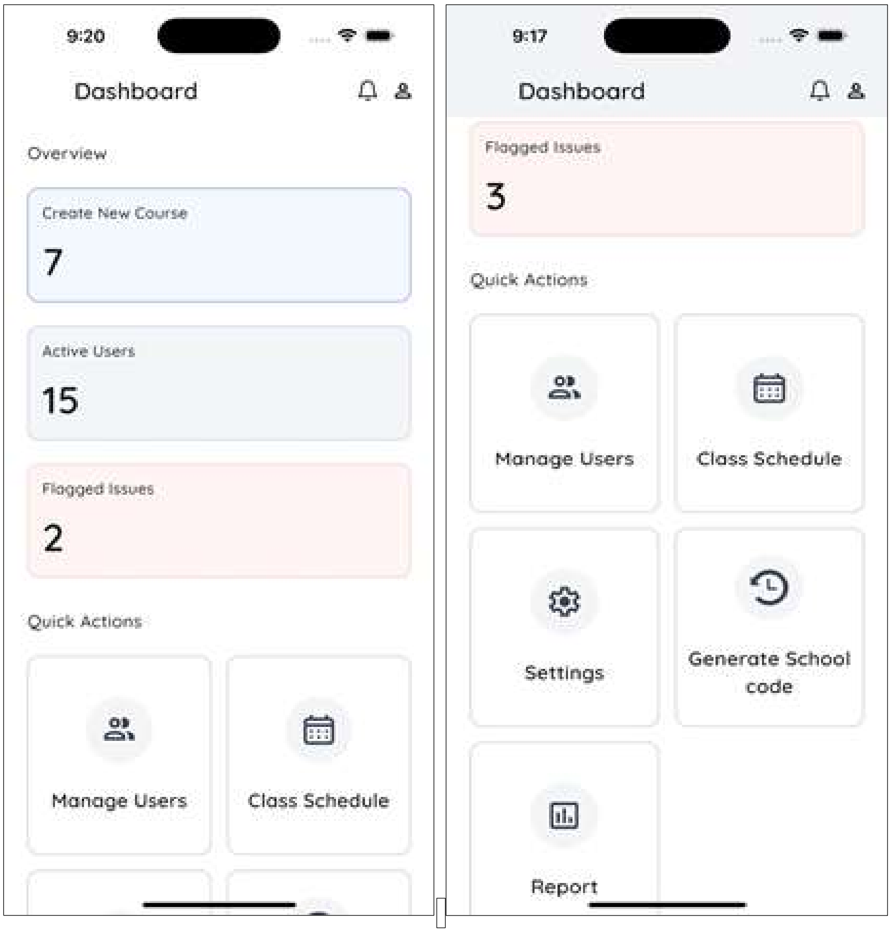
Figure 5: Class Setup Screen



Source: Results obtained from the experiment conducted with the developed system

The admin has a similar login flow to the lecturer, and on successful login, the dashboard, shown in Figure 6, provides administrators with comprehensive control and oversight. From this central hub, admins can manage courses, monitor active users, address flagged issues, access notifications and settings, generate school-specific registration codes for students, and review attendance reports. Each of these functions is accessible by clicking the corresponding buttons on the dashboard, which then directs the admin to the relevant screen.

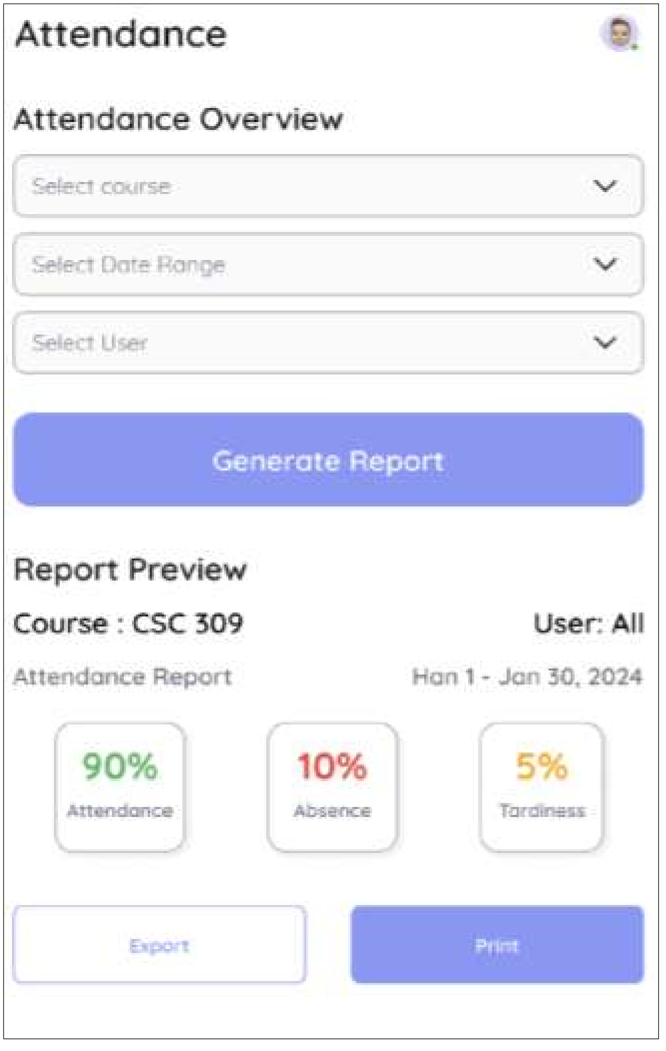
Figure 6: Admin Dashboard



Source: Results obtained from the experiment conducted with the developed system

Figure 7 presents the attendance report summary available to the admin. This comprehensive report offers robust decision-making capabilities, allowing the admin to filter by academic year, select specific date ranges, and even view individual student attendance. Furthermore, the admin can print or export the course attendance report as needed. Figure 8 shows the student's dashboard. Through their dashboard, the student can view their upcoming class, access their notifications and profile settings, monitor live or ongoing attendance, mark attendance, and view their attendance history. The bottom navigation bar shows that the students can route to attendance, schedule, and settings.

Figure 7: Admin Attendance Overview



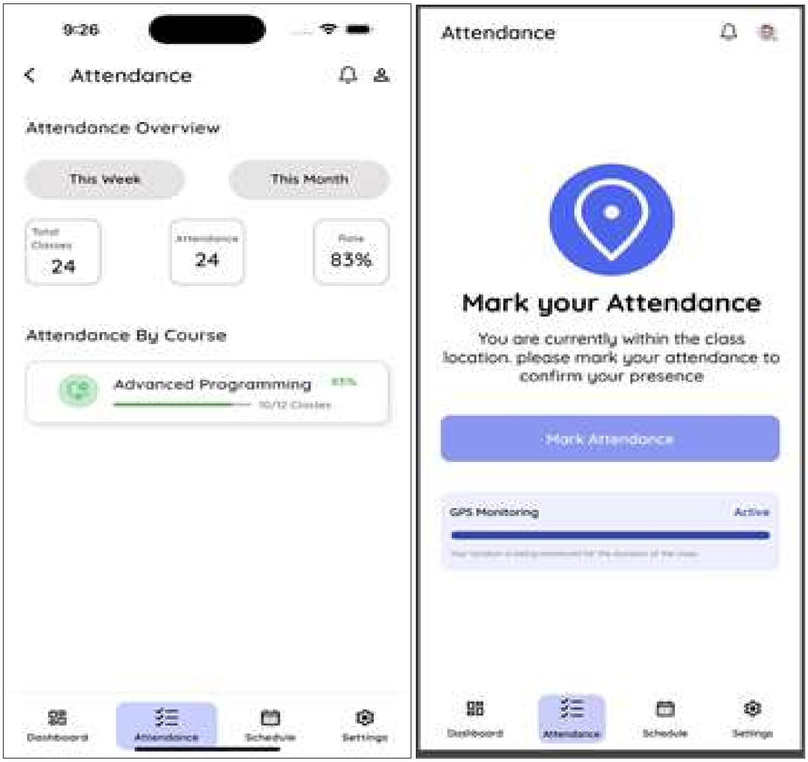
Source: Results obtained from the experiment conducted with the developed system

Figure 8: Student's Dashboard Screen

Source: Results obtained from the experiment conducted with the developed system

Figure 9 below displays the Attendance screen for the student. When a class or an attendance session is not ongoing, the screen on the left is displayed, and the student can view their attendance summary for the week or month, and get access to their attendance rate based on the week or month. They can also view courses and their specific attendance rate. The screen on the right is what comes up when the student is currently within the class location, and an attendance session is ongoing. The student can then mark their attendance and track the GPS monitoring.

Figure 9: Student Attendance Screens



Source: Results obtained from the experiment conducted with the developed system

4.3 Performance evaluation of the proposed study

To evaluate an attendance system's user experience, this study employed a mixed-methods approach with both surveys and structured interviews. The research involved 45 total users, specifically 30 students and 15 instructors, to capture the perspectives of both primary groups. System usability was quantitatively measured using the standardized System Usability Scale (SUS), while the qualitative interviews provided deeper context to the survey findings. Overall, the attendance system was received positively, with a high user satisfaction score of 4.2 out of 5 and an excellent System Usability Scale (SUS) score of 78.5. Its key strengths include an intuitive attendance process, valued real-time updates, a clear history dashboard, and efficient authentication.

However, opportunities for improvement remain in areas such as enhanced instructor controls, better handling of geolocation errors, expanded offline functionality, and more customizable notifications. User interface assessment showed that the system was highly effective, with a 96% success rate for key tasks. Users also learned it quickly, as their average task time dropped significantly after just one week of use. Table 1 shows that for normal academic loads of up to 500 users, the system performs reliably with fast response times and a high success rate. However, performance degrades significantly with 1,000 concurrent users, as database connections become a bottleneck and response times slow. The system otherwise proved very reliable, with excellent uptime and perfect data consistency across all tests.

Table 1: Testing Environment: Simulated academic scenarios with varying user loads

Concurrent Users	Response Time (avg)	Success Rate	CPU Usage
50 users	320ms	99.8%	24%
200 users	650ms	99.1%	58%
500 users	1.2s	97.3%	82%
1000 users	3.4s	88.7%	95%

Source: Results obtained from the experiment conducted with the developed system

Analysis of attendance data from the developed system showed that it demonstrates a consistently high level of accuracy across various testing scenarios, though its performance is not flawless. In standard operational environments, such as a simulated classroom, the system correctly identified over 94% of attendance records when compared to a manually verified ground truth. This strong core performance is supported by a high geolocation success rate and near-perfect time-based accuracy, indicating that the core mechanisms for capturing and logging attendance are fundamentally sound.

However, the analysis reveals a clear pattern where accuracy diminishes in more challenging conditions. The system's performance was slightly lower in a controlled lab setting and notably decreased to 90% when handling pre-defined edge cases. The presence of both false positives, where a student is incorrectly marked as present, and false negatives, where a present student is missed, suggests specific vulnerabilities. These inaccuracies are likely attributable to the limitations of the geolocation technology or complex, real-world scenarios that deviate from the system's standard operating parameters.

Table 2 presents This table presents the accuracy and error rates of a system across three progressively challenging test environments: a Controlled Lab, a Simulated Classroom, and a set of Edge Cases while the attendance system is highly reliable for routine use and excels in timestamp and permission accuracy, the data indicates a need for targeted improvements. Enhancing the system's robustness to handle edge cases and refining its geolocation logic to reduce both false positives and negatives would be necessary to push its accuracy closer to 100% in all potential academic situations.

Table 2: Performance Evaluation of Geolocation-Based Student Attendance System Across Different Test Environments

Test Scenario	Total Records	Correct	False Positive	False Negative	Accuracy
Controlled Lab	1,250	1,195	32	23	95.6%
Simulated Classroom	2,800	2,630	98	72	94.3%
Edge Cases	450	405	25	20	90.0%

Source: Results obtained from the experiment conducted with the developed system

4.2 Discussion of findings

The evaluation of the geolocation-based attendance system confirmed its design successfully addresses key flaws in existing methods, namely proxy fraud, rigid geofencing, and single-check validation. The core finding is that its dwell-time logic—requiring students to stay within a dynamic, lecturer-cantered geofence for 80% of the class, verified by 20-minute checks effectively ensures sustained presence, not just entry.

Technically, the system proved robust and user-friendly. Its cross-platform architecture (Flutter/Node.js/PostgreSQL) with role-based dashboards worked well. It showed strong performance under normal loads (over 97% success for up to 500 users) and high accuracy (94.3%) in simulated classrooms. User feedback was positive, with high usability scores and intuitive interfaces. However, the study also identified critical limitations. Performance degrades at 1,000 users, revealing a scalability bottleneck. Accuracy drops to 90% in edge cases, exposing inherent GPS technology flaws like signal loss or drift, which can cause false positives/negatives. Users requested improvements in error handling and offline functionality. The system represents a significant advance as a dynamic, cost-effective software solution that promotes accountability. Its main contribution is providing empirical evidence of both its efficacy and its practical limits, suggesting that for maximum reliability, future systems may need to combine geolocation with a secondary verification method.

5.0 Conclusion

This study successfully designed, implemented, and evaluated a novel Geolocation-Based Student Attendance Management System. The system addresses the critical limitations of traditional and existing digital methods, such as proxy attendance, hardware dependency, and static validation by introducing a dynamic, software-only solution. Its core innovation lies in the implementation of a dwell-time verification logic, which requires students to remain within a lecturer-defined geofence for at least 80% of the class session, validated through periodic 20-minute location checks. This ensures sustained physical presence, moving beyond simple entry logging to foster genuine academic engagement. The system's technical architecture, built with Flutter for cross-platform accessibility and a Node.js/PostgreSQL backend, proved effective and scalable for typical academic loads. Performance evaluation demonstrated high reliability and user satisfaction, with an excellent System Usability Scale (SUS) score of 78.5 and a 94.3% accuracy rate in simulated classroom environments.

The role-specific dashboards and real-time features were well-received, enhancing transparency and administrative efficiency. However, the evaluation also revealed inherent challenges, primarily tied to the limitations of GPS technology itself. Accuracy decreased in edge cases (90%), and performance bottlenecks emerged under extreme concurrent user loads. These findings underscore that while geolocation provides a powerful, cost-effective foundation, absolute reliability in all scenarios may require future augmentation with complementary lightweight verification methods. In conclusion, this work provides a validated, practical framework for the digital transformation of attendance management. It demonstrates that an intelligent, dwell-time-based geolocation system can significantly improve accuracy, accountability, and user experience in educational institutions. The system offers a substantial step forward, balancing innovation with practicality, and provides a clear roadmap for future enhancements focused on robustness and scalability.

5.1 Recommendation

This study recommends enhancing the system's handling of GPS errors and offline scenarios to improve reliability. Future development should focus on optimizing backend scalability to support over a thousand concurrent users. Implementing user-requested features, like granular instructor controls, is advised. For broader impact, a hybrid model combining geolocation with a secondary, lightweight verification method should be researched. A long-term pilot in a live academic setting is essential to validate real-world efficacy. Institutions must establish clear ethical policies on data privacy and location tracking before adoption. Ensuring robust campus WiFi and providing user training will be critical for successful deployment. These steps will solidify the system's role as a robust and accountable digital solution for modern education.

References

Babatunde, A. N., Oke, A. A., Babatunde, R. S., Ibitoye, O. & Jimoh, E. R. (2022). Mobile-based student attendance system using geo-fencing with timing and face recognition. *Journal of Advances in Mathematical & Computational Sciences, 9*(2), 75–90.

Bailke, P., Gujarathi, O., Gundecha, S., Gore, O., Jadhav, A. & Chaudhari, J. (2024). Smart Attendance System using Geolocation. *Grenze International Journal of Engineering & Technology (GIJET)*, *10*(1), 2230.

Chauhan, R. K., Prakash, R., Tabrej, S. & Singh, S. (2024). A review on geo-location based attendance management system. *International Journal of Innovative Science and Research Technology, 9*(2), 484-487.

Eweoya, I., Adeniyi, O. J. & Awoniyi, A. O. (2025). *Design and implementation of a university attendance management system using geo-fencing.* Asian Journal of Computer Science and Technology. Retrieved from https://doi.org/10.70112/ajcst-2025.14.1.4323

Jeevarathinam, A. & Janani, K. (2025). GPS-based attendance tracking system using Haversine distance algorithm. *International Research Journal of Modernization in Engineering Technology and Science, 7*(3), 4216-4222.

Kundariya, N., Patil, P., Ingle, A., Muthal, G. & Gaidhani, P. S. (2025). Geofencing and location-based attendance system. *International Journal of Advanced Research in Science, Communication and Technology (IJARSCT), 5*(4), 779–783. Retrieved from https://doi.org/10.48175/IJARSCT-24080

Nazara, E. M. & Nasien, D. (2024). Employee attendance system using rapid application development method based on location-based service. *Journal of Applied Business and Technology, 5*(2), 96–104. Retrieved from https://doi.org/10.35145/jabt.v5i2.148

Ramon, F. S., Chávez Meléndez, C. E., & Cerda García, R. O. (2024). Geolocation and biometric authentication system for the registration of attendance in educational centers. *In 2024 IEEE 4th International Conference on Advanced Learning Technologies on Education & Research (ICALTER)* (pp. 1–4). IEEE. Retrieved from https://doi.org/10.1109/ICALTER65499.2024.10819242

Sandhya, K., Kumari, G., Niharika, J. & Goud, K. P. (2025). Geolocation based student attendance management system using QR-code and Google Maps API. *Journal of Engineering Sciences, 16*(5), 164–172.

Shahab, A. S., & Sarno, R. (2020). Android application for presence recognition based on face and geofencing. *2020 International Seminar on Application for Technology of Information and Communication (iSemantic), Semarang, Indonesia*. Retrieved from https://doi.org/10.1109/iSemantic50169.2020.9234253

Syamaidzar, F. N. & Sutopo, J. (2024). Implementasi location-based service pada sistem presensi geolokasi berbasis web. *Journal of Information System Research (JOSH), 5*(2), 459–467. Retrieved from https://doi.org/10.47065/josh.v5i2.4743

Valdez, C. J. H., Nivales, Y. M., Uy, R. J. R. & Montañez, J. J. F. (2024). RFID-based student monitoring system for accessing institutional campus. *In 2024 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies* (pp.427–434). IEEE. Retrieved from https://doi.org/10.1109/3ICT64318.2024.10824565