Asian Journal of Computer Science and Technology
ISSN: 2249-0701 (P); 2583-7907 (E)
Vol.14 No.1, 2025, pp.28-46

www.crijournals.org
DOI: https://doi.org/10.70112/ajcst-2025.14.1.4323

Design and Implementation of a University Attendance Management System Using Geo-Fencing

Ibukun O. Eweoya[1]*, Oluwabamise J. Adeniyi[2], Amos O. Awoniyi[3], Emmanuel Mgbeahuruike[4], Joseph O. Adewuyi[5], Taiwo O. Adigun[6] and Yaw Asa Mensah[7]
[1,2,3,4&6]Department of Software Engineering, [5]Department of Computer Science,
Babcock University, Ilishan-Remo, Ogun State, Nigeria
[7]Department of Information Technology, Valley View University, Ghana
E-mail: adeniyiolu@babcock.edu.ng, awoniyia@babcock.edu.ng, mgbeahuruikee@babcock.edu.ng,
adewuyij@babcock.edu.ng, adigunt@babcock.edu.ng, yaw.mensah@vvu.edu.gh
*Corresponding Author: eweoyai@babcock.edu.ng
(Received 22 December 2024; Revised 29 January 2025, Accepted 19 February 2025; Available online 2 March 2025)

Abstract - **Efficient attendance management is crucial in educational institutions to ensure student participation and compliance with policies. Traditional methods, such as physical registers and manual inputs, are often inefficient, inaccurate, and susceptible to manipulation, limiting their effectiveness. Modern technologies provide innovative solutions to address these challenges. This study aimed to design and implement a web-based attendance management system using geofencing technology to enhance the accuracy, security, and efficiency of attendance tracking at Babcock University. A mixed-methods approach was employed. Qualitative data were gathered through interviews with students and lecturers to identify flaws in the current system. A literature review informed the system's design, with an emphasis on geofencing technologies. The system was developed using Python for backend logic and JavaScript for the frontend. It supports iOS and Android devices, offering functionalities such as secure user authentication, geofence creation for lecture halls, and automated attendance monitoring. The system effectively addressed the shortcomings of traditional methods by providing accurate and automated attendance tracking, minimizing errors, and ensuring compliance with university policies. Lecturers and students found the system intuitive, efficient, and reliable for managing class attendance. The geofencing-based attendance management system offers a scalable and user-friendly solution for modernizing attendance tracking in educational institutions. Its integration of geolocation and automation improves operational efficiency, data reliability, and policy adherence, serving as a model for similar systems in other educational contexts.**
Keywords: **Attendance Management, Geofencing Technology, Automated Tracking, Educational Institutions, User Authentication**

I. INTRODUCTION

A. Attendance and Geofencing-Based Systems

Attendance refers to the act of registering and tracking individuals' presence or absence in a group context, such as a class, meeting, or event. This process is often undertaken to monitor and control participation, maintain records for administrative purposes, and ensure accountability. Attendance is recorded using various methods, such as verbal checks, class sheets, attendance activities, and applications. These methods can be broadly classified into two categories: manual and technical methods [1], [2]. Class attendance is considered a critical activity for students, as absenteeism, particularly during their first year, negatively impacts their academic performance [3]-[5]. Many universities, including Babcock University, implement minimum attendance policies. At Babcock University, students are required to attend at least 75% of their classes to be eligible to take final exams [6]. This policy is strictly enforced, with students who fail to meet the attendance requirement receiving grades reflective of their attendance performance.

Digital attendance systems leverage technology to record the presence or absence of individuals at a specific location [1], [7]-[9]. These systems have proven to be more efficient and reliable compared to manual methods, which have demonstrated inefficiencies over time [10], [11]. Geofencing is a location-based service where a mobile device or RFID tag triggers an action upon entering or exiting a predefined virtual boundary, known as a geofence [12], [13]. Incorporating geofencing with attendance systems offers numerous advantages, including improved accuracy, efficiency, and security.

A geofencing-based attendance system is a technological solution that monitors and records attendance by using geofencing-a virtual boundary defined by geographic coordinates [13]-[15]. Such systems have been implemented in various domains, including education, business, sports, and entertainment.

B. Statement of the Problem

At Babcock University, as in many institutions, attendance is a critical criterion for students to participate in final examinations. This policy emphasizes that class participation is essential before exams. Students failing to meet the minimum 75% attendance requirement are assigned an FA grade, indicating failure due to attendance.

However, the current system for recording attendance is inefficient and inaccurate. Attendance is typically taken manually, using physical lists or manual inputs by lecturers.

These methods are prone to manipulation, as students can bypass the system through proxy attendance. Consequently, the university's attendance policy cannot be effectively enforced.

Several institutions have implemented geofencing-based attendance systems, either developed in-house or provided by third-party vendors. This research seeks to address the inefficiencies of the current system by designing and developing a web-based attendance management system.

Using geofencing technology, lecturers can define the geographic boundaries of a class. Students outside this boundary during the specified time will be marked absent. The system will calculate total attendance points for each student at the end of the semester, ensuring accurate attendance records and enforcing the university's policies.

C. Objective of the Study

The aim of this study is to design and implement a web-based attendance management system using geofencing technology for a private institution, with Babcock University as a case study. The specific objectives are to:

1. Gather requirements for the attendance management system.
2. Design a web-based attendance management system.
3. Implement the web-based attendance management system.
4. Evaluate the effectiveness of the web-based attendance management system.

D. Scope of the Study

This study focuses on Babcock University and aims to improve the accuracy and efficiency of the attendance collation process for lecturers. The project involves developing an application that accurately tracks student presence in class using geo-mapping technology. It will calculate the total attendance marks for each student at the end of the semester.

The scope is constrained to Babcock University. Expanding the system to other institutions would require additional resources and collaboration. The success of this system relies on active participation from students and lecturers. Students must attend classes with a functioning smartphone equipped with a front camera for facial detection, while lecturers need to be familiar with using the web application to set up location, class, and time constraints. The web application will be developed using the Streamlit framework, ensuring accessibility on both iOS and Android platforms. JavaScript will be used for development, providing swift functionality and compatibility across devices.

II. REVIEW OF LITERATURE

Throughout history, the practice of attendance monitoring has evolved alongside changes in educational and organizational systems. In ancient times, educators used informal methods to track student presence. During the medieval and Renaissance periods, more formalized attendance records were introduced in official institutions. With the industrial revolution in the nineteenth century, attendance monitoring became more systematic, and manual methods such as paper registers became prevalent.

The introduction of punch cards and mechanical time clocks in workplaces during the twentieth century marked the incorporation of technology. The late twentieth century witnessed a transition to automated attendance systems, paving the way for more sophisticated administration. Attendance systems in the twenty-first century have integrated modern technologies such as biometrics, RFID, and GPS to improve accuracy and efficiency [1], [16]. Cloud-based and mobile applications are now commonly used in modern systems to provide flexibility and accessibility.

In some existing studies, authors have presented a system utilizing GPS and standard coordinates, seamlessly integrating Google Maps APIs [14]. Attendance calculation employed the Haversine formula, classifying users as instructors and students, who were authenticated through Google Maps APIs.

The geofencing system targeted improved efficiency, concurrent processing, and reduced errors in attendance records. Challenges identified included GPS dependency, security issues in user authentication, scalability concerns for larger student populations, and a lack of comparison with alternative attendance tracking technologies.

In another study, the authors introduced an attendance management system tailored for blended learning in tertiary institutions, incorporating geofencing, biometrics, and Bluetooth technology [17]. The system employs a unified database, specialized algorithms for both lecturer and student attendance, and ensures security through a RESTful API. The primary goals include improving engagement tracking, early identification of disengagement, and streamlining report evaluation. However, existing research gaps include the need to address scenarios involving both face-to-face and online learning, as well as overcoming limitations related to indoor dimensions.

Additionally, some studies propose an innovative Automated Time and Attendance System that utilizes location-based services for enhanced employer control over working hours [18], [19]. The system replaces traditional ID cards with a GPS-enabled mobile app for attendance tracking.

The methodology integrates the app, GPS technology, and attendance management software, depicted in a basic block diagram. Results compare the new system with traditional methods, emphasizing cost, maintenance, and automation. However, the study acknowledges incomplete implementation due to time constraints, with limited discussions on challenges and potential improvements. In conclusion, while the approach shows promise, a more thorough evaluation is necessary to assess its practicality and effectiveness.

Gedam *et al.,* developed a GPS-based staff attendance system using geofencing, which proved efficient, cost-effective, and user-friendly, eliminating the need for expensive biometric devices. The system addressed concerns such as GPS signal dependency, privacy issues, device compatibility, scalability, and integration challenges. However, refinements are needed for broader applicability [20].

Makhtar *et al.,* developed a Mobile Attendance Application employing the geofence technique to track off-site staff attendance using GPS-enabled smartphones [21]. The Android-based system aims for efficiency by automating attendance recording through a single tap on the application. The methodology involves implementing a unique algorithm that uses geofence queries to establish virtual boundaries for specific locations. Staff locations, identified by IMEI numbers, are recorded along with the date and time upon tapping the application. The administration then verifies the accuracy of the information and the legitimacy of the device.

The research emphasizes the importance of this technology in providing an efficient and cost-effective approach to managing staff attendance. The implementation results indicate the application's successful functioning, recording location, date, and time for staff members using GPS-enabled smartphones, even outside the office building.

Galgale *et al.,* presented a GPS-based Location-Based Attendance System using smartphones, employing a geofence technique through an Android application for off-site staff [22]. The Mobile Attendance System (MAS) minimizes user input for attendance tracking, ensuring effective real-world testing. The article suggests enhancing security discussions, conducting a detailed user experience analysis, exploring scalability to other platforms and smartphone models, and providing a more comprehensive comparison with existing systems for a robust understanding.

Another study introduced an Intelligent Attendance Management System that combines geofencing and facial recognition, leveraging technologies such as Python, Firebase, and Google API [23]. The system achieves effective user registration, login, geofence configuration, and facial recognition for efficient attendance monitoring, focusing on reducing manual efforts and curbing proxy attendance.

Despite its successes, the article fails to explicitly address potential limitations, and the future work section outlines enhancements without acknowledging associated challenges.

The Students Attendance Management System, as proposed by Tasleem *et al.,* (2023), utilizes OpenCV for face detection, database creation, training data gathering, and model training with LBPH Face Recognizer [24]. The system, catering to administrators, lecturers, and students, features a user-friendly interface. It effectively captures and recognizes faces, enabling students to mark attendance through face recognition. A valuable addition would be a thorough examination of face recognition accuracy in educational environments to enhance the system's overall evaluation.

Babatunde *et al.,* presented a mobile-based student attendance system using Google services and face recognition [17]. The system, developed in Android Studio, combines face recognition for authentication and geofencing to monitor real-time student presence, ensuring accurate attendance records.

Results demonstrate its success in addressing identification and accuracy issues, automatically recording attendance within the designated classroom area. The study emphasizes the need for innovative solutions to traditional attendance challenges, introducing a novel timing geofence application to improve attendance monitoring.

Some studies propose contactless attendance systems using facial recognition, while others focus solely on facial recognition [23], [25], [26], [27]. One study, however, leverages geofencing in conjunction with facial recognition, along with an admin portal, to address challenges posed by the pandemic and traditional attendance methods [13]. The methodology involves faculty and admin portals, utilizing Python-based face recognition and the Haversine formula for geofencing.

Results indicate successful face recognition and geofencing, albeit with some errors. However, the article lacks detailed mitigation strategies for hardware issues and geofencing errors. The proposed microservices architecture and notification system are mentioned but lack specific implementation details. The overall scalability and adaptability claims need further elaboration.

III. METHODOLOGY

The research design for the attendance system using geofencing is a combination of qualitative and quantitative methods to achieve the objectives of the study. The project aimed to design an attendance system that would simplify the process of taking attendance for students at Babcock University, while also addressing issues such as manipulation, forgery, lack of a proper database, and lack of access control within the classrooms.

The first step in the research design was to conduct oral interviews among students and lecturers to gather information on the challenges faced with the current attendance system. The data obtained from the interviews were used to define the system requirements, which formed the basis for the design of the attendance system. In addition, a study of closely related works was conducted to identify appropriate methods for achieving the objectives of the project.

The findings from this study were used to inform the design of the attendance system. The system was designed to be easy to use, reliable, and secure, incorporating geofencing technology to ensure accurate attendance tracking. Overall, the research design involved a carefully planned and executed process to design and implement the attendance system for the classes at Babcock University. The methodology employed during this study allowed for the derivation of relevant information from related works, outlining system requirements, and implementing the system design using Java and Python programming languages in a sequential approach.

A. Functional Requirements

1. Lecturer

The Lecturer shall be able to:

- *a.* Login using the university-provided credentials or created credentials.
- *b.* Select existing geofenced lecture halls for attendance taking.
- *c.* Create geofences for new lecture halls.
- *d.* Save and name geofenced lecture halls for future use.
- *e.* Generate a unique code for each attendance session.
- *f.* Set a start time for each attendance session.
- *g.* Set a stop time for each attendance session.
- *h.* View real-time attendance based on the generated attendance code.
- *i.* See an overview of attendance per class.

2. Student

The student shall be able to:

- *a.* Login using their sign-up credentials.
- *b.* Mark attendance by entering the unique class code generated by the lecturer within the set time frame.
- *c.* View marked class attendance.

3. General Functional Requirements

The system shall:

- *a.* Maintain logs of user activities for auditing purposes.
- *b.* Ensure compliance with university policies and regulations regarding student data and attendance.
- *c.* Generate accurate geofences to ensure students can only mark attendance when they are physically within the specified lecture hall.

B. Non-functional Requirements

1. Lecturer

- *a.* *Security:* The application should ensure data security and privacy for lecturer accounts and attendance records.
- *b.* *Usability:* The user interface should be intuitive and user-friendly for lecturers to create geofences and generate class codes.
- *c.* *Performance:* The system should handle concurrent requests from multiple lecturers without significant delays.
- *d.* *Reliability:* The system should be highly available and reliable, ensuring that lecture attendance data is accurately recorded.

2. Student

- *a.* *Security:* Student data and attendance records should be protected to prevent unauthorized access.
- *b.* *Usability:* The web interface should be easy for students to use to mark their attendance.
- *c.* *Geofencing Accuracy:* The geofencing system should have a high degree of accuracy to ensure that students can only mark their attendance when physically present within the specified area.
- *d.* *Performance:* The application should be responsive, even when many students are marking their attendance simultaneously.

3. Hardware Requirements

- *a.* 1 GHz or faster 32-bit or 64-bit processor.
- *b.* 2 GB RAM.
- *c.* 16 GB available disk space.
- *d.* Mobile device with GPS capability.

C. System Model

The proposed model design for the geofencing-based attendance system at Babcock University is an application. The system will consist of the following components:

1. *User Interface:* The user interface will be web-based and accessible from any mobile device with an internet connection. The interface will be simple and easy to use, with clear menus and options for users to navigate through the system.
2. *Database:* The system will store user data and geofenced data, including student details and attendance history, in a database.
3. *Lecturer Panel:* The system will provide lecturers with a web-based panel for creating and viewing attendance, as well as geofencing new venues.
4. *Reporting Module:* The system will have a reporting module that generates reports and logs on attendance taken.
5. *Admin Panel:* The system will have an admin panel that will allow the university to access and generate logs and attendance reports.

The proposed system will effectively mark attendance and curb the issue of failure due to attendance at the university.

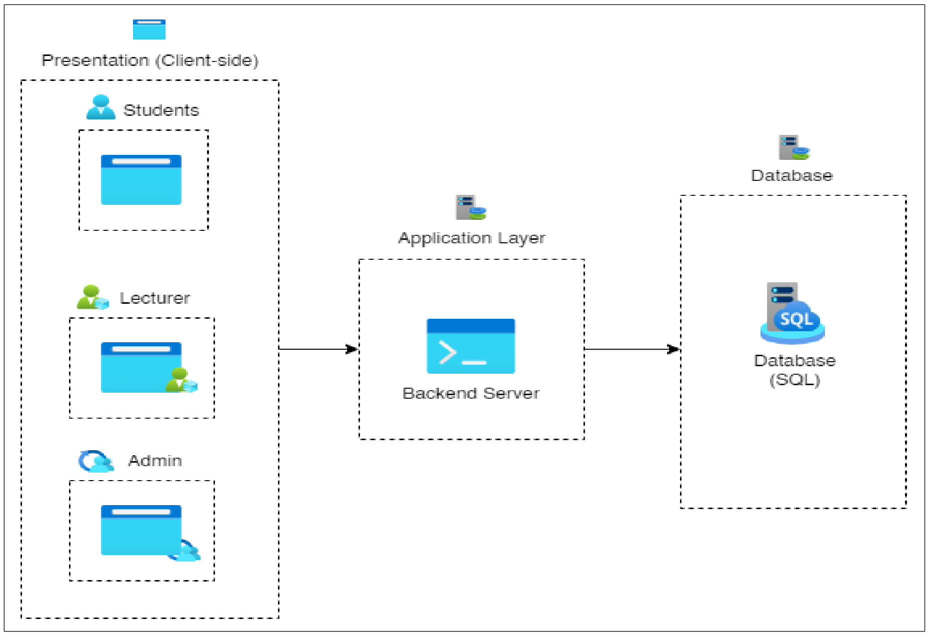

Fig. 1 System Architectural Diagram

D. System Architecture

1. *Presentation Tier (Client):* Web-Application: The client-side application for both lecturers and students is developed using JavaScript and Python. This tier handles the user interface, user interactions, and communicates with the server-side for data retrieval and updates.
2. *Application Tier (Server):* Backend Server (Python): The server-side logic, responsible for geofencing, authentication, attendance tracking, and business rules, is implemented in Python. This tier handles the application's core functionality and acts as an intermediary between the client and the database.
3. *Data Tier (Database):* Database (e.g., SQLite): The data tier stores and manages information such as user credentials, geofenced locations, lecture sessions, attendance records, and other relevant data. The choice of the specific database system depends on factors like scalability, data structure requirements, and the overall system architecture.

E. Use Case Diagram, Class Diagram, and Sequence Diagram

A design model in software engineering is an object-based representation or series of representations that depict the use cases for a system. In other words, it is a means of describing a system's implementation and source code in a diagrammatic format.

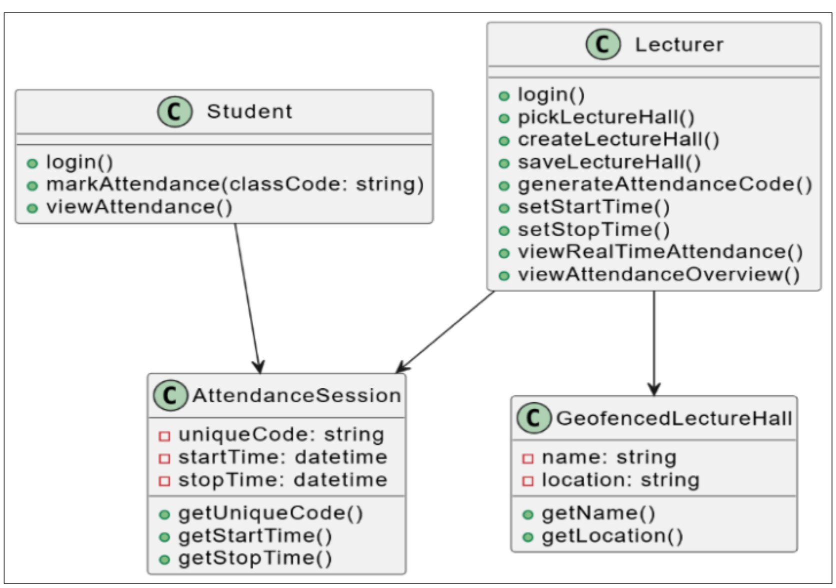

Fig. 2 Class Diagram

1. Class Diagram

The class diagram is used to represent the class model. It displays the class name, followed by its attributes, and then the functions or methods associated with the class object. The goal of constructing a class model is to capture the real-world concepts that are important to the application.

2. Use Case Diagram

A use case diagram is a graphical representation of a user's possible interactions with a system. It illustrates various use cases and the different types of users within the system. Use case diagrams are often accompanied by other diagram types. The use cases are represented by circles or ellipses, while the actors are typically depicted as stick figures.

Fig. 3 Use Case Diagram for Attendance System

3. Sequence Diagram

A sequence diagram illustrates process interactions arranged in a time sequence within the field of software engineering. It depicts the processes involved and the sequence of messages exchanged between the processes required to execute the functionality.

F. Development Tools

The software development methodology adopted for this project is the Agile model. Agile is known for its iterative and incremental development approach, allowing for continuous refinement of the software throughout the development process. Given the nature of the project, where requirements may evolve or need adjustments, this iterative approach provides flexibility and the ability to adapt to changes smoothly.

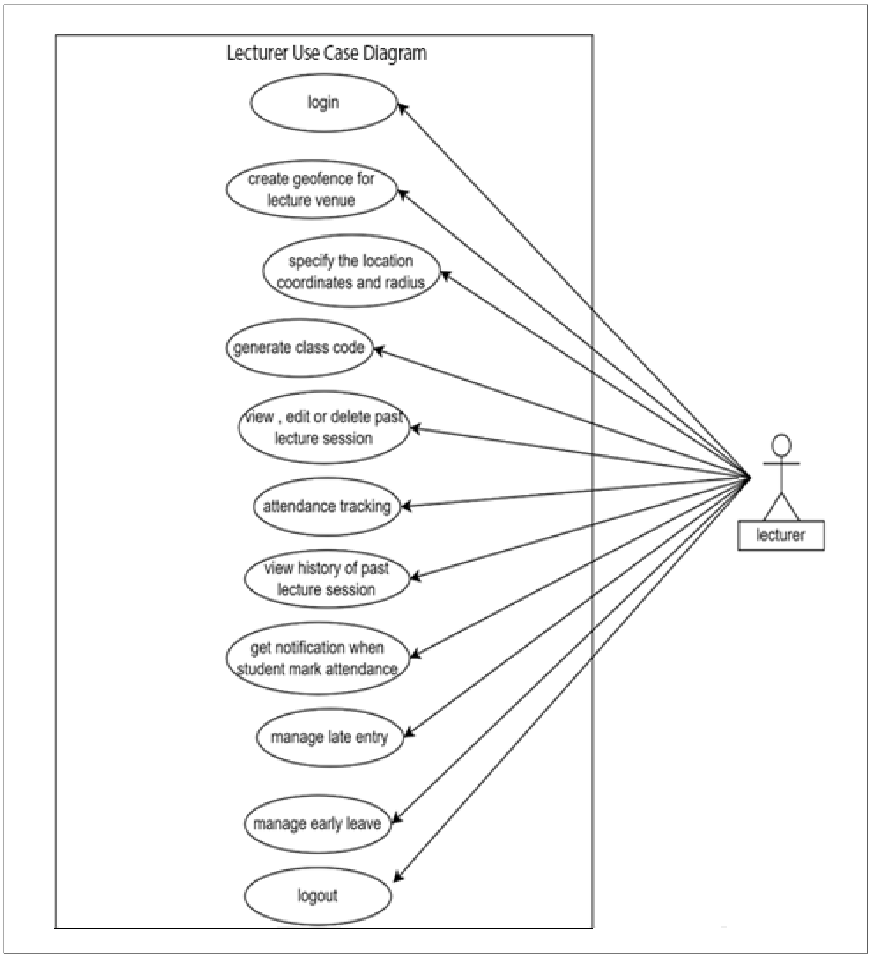

Fig. 4 Use Case Diagram for Lecturer

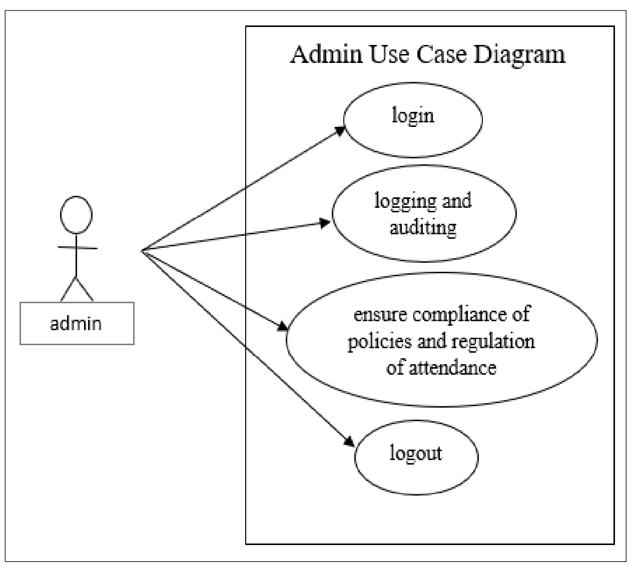

Fig. 5 Use Case Diagram for Admin

In the Agile model, a basic working system is initially released to the clients. Subsequent modules and functionalities are then implemented, released, and added to the existing project. This approach allows for greater flexibility in accommodating changes in requirements, as the team focuses on implementing the core functionalities of the system at each stage. Feedback is obtained from the client with each successive iteration, and the modified requirements are incorporated into the new module.

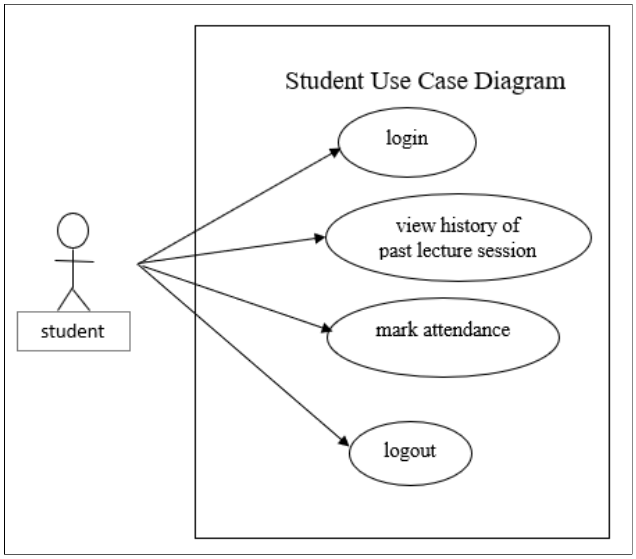

Fig. 6 Use Case Diagram for Students

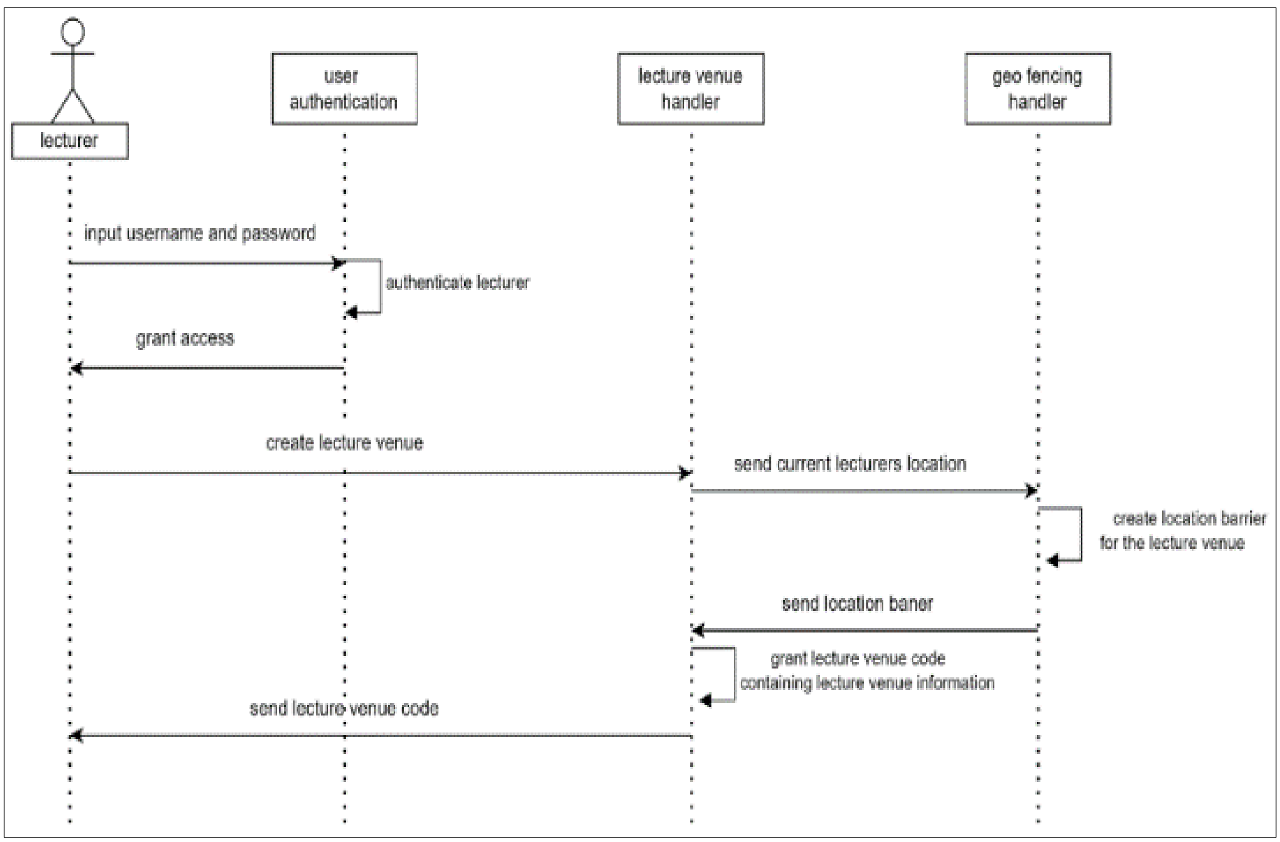

Fig. 7 Sequence Diagram for Lecturer

These are the tools used in the development of this research software:

1. *Visual Studio Code:* Visual Studio Code is a streamlined code editor with support for development operations such as debugging, task running, and version control. It is designed to provide the necessary tools for a quick code-build-debug cycle. It was used to program the client-side aspect of the application.
2. *Azure App Service:* Azure App Service enables the building and hosting of web applications in the programming language of your choice, without the need to manage infrastructure. Web apps in Azure allow for easy publication and management of websites without working with the underlying servers, storage, or network assets. It was used to host the software.

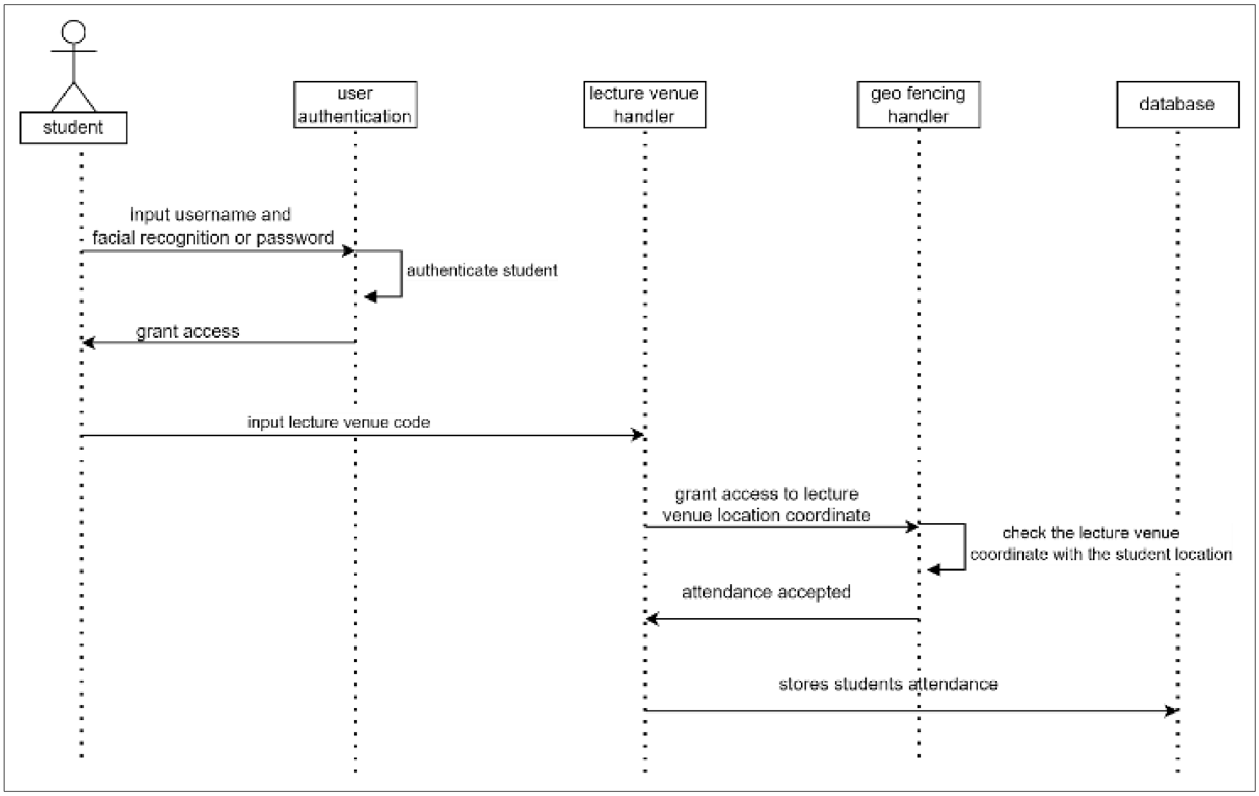

Fig. 8 Sequence Diagram for Student

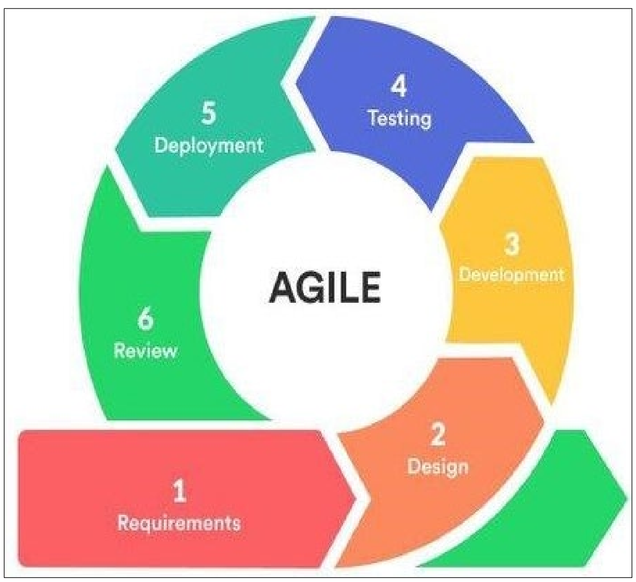

Fig. 9 Agile Development Methodology

3. *SQLite:* SQLite is a lightweight, self-contained, serverless, and open-source relational database management system. It allows users to create, read, update, and delete data in a relational database using a SQL-based interface. SQLite is commonly used in embedded systems, mobile applications, and small-scale database applications due to its simplicity, reliability, and portability.

4. *Microsoft Visio:* Microsoft Visio is a diagramming and vector graphics application, part of the Microsoft Office family. It was used to create diagrams such as the use case and class diagrams.

The programming languages used include:

1. *JavaScript:* JavaScript was used because it has access to the Geolocation API, allowing the browser to retrieve the user's current location. This functionality is essential for tracking when a user enters or exits a defined geofence area.
2. *Python:* The geofencing logic is implemented using Python libraries like Shapely for geometric calculations and defining geofence shapes (e.g., circles, polygons). The database system, such as MongoDB, is used to store geofence data and associated information, including geofence shapes, metadata, and user-related data.
3. *Server Deployment:* The Python backend is deployed on a server or cloud platform, such as AWS or Heroku, for scalability and reliability.

G. Communication Between JavaScript and Python

In the proposed research project, a geofencing attendance system is being developed using JavaScript for the frontend, Streamlit as the web framework, Python for the backend, and SQLite for the database. The communication between these components is established through HTTP requests and responses. The frontend, implemented in JavaScript with Streamlit, handles user interactions and sends HTTP requests to the backend when data needs to be exchanged. The backend, developed in Python, receives these requests, processes them, interacts with the SQLite database to retrieve or store data, and generates appropriate responses. SQLite serves as the database management system, storing the necessary data for the application. This communication flow enables a clear separation of concerns and facilitates the development of a robust and efficient geofencing attendance system.

H. Additional Tools and Services

Google Places API: If the geofencing app involves working with specific locations or points of interest, consider integrating the Google Places API for location-related data.

IV. IMPLEMENTATION AND EVALUATION

The implementation of this system is a testament to the application's modular and extensible architecture. It seamlessly integrates various components, leveraging the power of SQLAlchemy, an Object-Relational Mapping (ORM) library for Python, to facilitate communication with the underlying database.

At the core of the system are several models, each representing a critical aspect of the application. The Lecturer model defines the structure and properties of lecturers, including their name, username, and password, ensuring secure authentication and identification. The student model encapsulates similar attributes for students, enabling secure access and tracking of their attendance. The Class model lies at the heart of the application, storing essential class details such as course name, venue, class ID, number of students, and the associated lecturer's ID.

This model establishes a strong association between classes and their respective lecturers, ensuring data integrity and privacy. The system's functionality is implemented within the Streamlit application, a powerful open-source Python library for building interactive web applications. Streamlit provides an intuitive and responsive user interface, allowing both lecturers and students to seamlessly navigate through the application's various features.

For lecturers, the application offers a dedicated class module where they can create and manage their classes. The class creation process is facilitated through a user-friendly form, allowing lecturers to input relevant class information, including the longitude and latitude coordinates of the class location. These coordinates are stored in the Class model, leveraging SQLAlchemy's support for floating-point data types, enabling precise attendance tracking based on student proximity to the designated area.

Once a class is created, the application stores the class information in the database, establishing a robust link between the class and the lecturer who created it. This association ensures that lecturers can only access and manage the classes they have created, maintaining data integrity and privacy. Additionally, the lecturer module provides a comprehensive overview of all the classes a lecturer has created, dynamically generated by querying the database using SQLAlchemy's powerful query capabilities.

For students, the application offers a seamless attendance marking process. Students can view the available classes and mark their attendance by entering the unique class ID provided by the lecturer. Upon entering the correct class ID, students are prompted to take a picture using their device's camera, serving as a visual record and verification of their presence.

Leveraging the power of geofencing technology, the application verifies the student's proximity to the designated class location. If the student is within the predefined geofence radius, typically set to 100 meters, their attendance is marked as successful. The attendance data, including the student's ID, class ID, timestamp, location coordinates, and captured image, are securely stored in the Attendance model within the database.

The implementation of this system showcases the application's adherence to best practices in software development, leveraging industry-standard libraries and frameworks to deliver a secure, scalable, and user-friendly experience for both lecturers and students.

A. Lecturer Module

The lecturer module plays a pivotal role in the attendance management system, serving as the gateway for lecturers to manage and oversee their respective classes. This module is designed to streamline the process of class creation, attendance tracking, and overall class administration, leveraging the power of geofencing technology.

At the core of the lecturer module lies the Lecturer model, which defines the structure and properties of a lecturer within the system. The model encapsulates essential attributes such as name, username, and password, enabling secure authentication and identification of lecturers. The unique username constraint ensures that each lecturer has a distinct identity, preventing duplication and potential conflicts.

Upon accessing the application, lecturers are presented with a user-friendly interface that guides them through the signup and login processes. The signup functionality allows new lecturers to register by providing their name, username, and password. This information is securely stored in the database, ensuring the privacy and integrity of user data.

Once registered, lecturers can seamlessly log in to the system using their credentials. The login process involves validating the provided username and password against the stored data in the database. If the credentials are valid, the lecturer is granted access to the class module, where they can create and manage their classes.

```python
if choice == "Home":
    st.subheader("Home")
    st.write("Welcome to the Class Attendance using Geofencing app!")

elif choice == "Signup":
    st.subheader("Lecturer Signup")
    name = st.text_input("Name")
    username = st.text_input("Username")
    password = st.text_input("Password", type='password')

    if st.button("Signup"):
        # Create a new lecturer
        lecturer = Lecturer(name=name, username=username, password=password)
        session = Session()
        session.add(lecturer)
        session.commit()
        session.close()
        st.success("Signup successful!")

elif choice == "Login":
    st.subheader("Lecturer Login")
    username = st.text_input("Username")
    password = st.text_input("Password", type='password')

    if st.button("Login"):
        # Validate the login credentials
        session = Session()
        lecturer = session.query(Lecturer).filter_by(username=username, password=password).first()
        session.close()

        if lecturer:
            st.success("Login successful!")
            st.session_state.logged_in_lecturer = lecturer
            st.experimental_rerun()
        else:
            st.error("Invalid username or password")
```

Fig. 10 Lecturer Module

```python
elif choice == "Class":
    if 'logged_in_lecturer' not in st.session_state:
        st.warning("Please login to access the class module")
    else:
        st.subheader("Class Module")
        course_name = st.text_input("Course Name")
        longitude = st.number_input("Longitude")
        latitude = st.number_input("Latitude")
        class_id = st.text_input("Class ID")
        venue = st.text_input("Venue")
        num_students = st.number_input("Number of Students", min_value=0, step=1)

        if st.button("Create Class"):
            # Create a new class
            class_ = Class(
                course_name=course_name,
                longitude=longitude,
                latitude=latitude,
                class_id=class_id,
                attendance=0,
                venue=venue,
                num_students=num_students,
                lecturer_id=st.session_state.logged_in_lecturer.id
            )
            session = Session()
            session.add(class_)
            session.commit()
            session.close()
            st.success("Class created successfully!")

        # Display the list of classes
        session = Session()
        classes = session.query(Class).filter_by(lecturer_id=st.session_state.logged_in_lecturer.id).all()
        session.close()
```

Fig. 11 Lecturer Module Session Code

The class module is a comprehensive section dedicated to facilitating the creation and administration of classes. Lecturers can input essential class details, such as course name, venue, class ID, and the number of students enrolled. Leveraging the power of geofencing, lecturers are required to provide the longitude and latitude coordinates of the class location, enabling precise attendance tracking based on student proximity to the designated area. Upon successful class creation, the application stores the class information in the database, associating it with the respective lecturer. This association ensures that lecturers can only access and manage the classes they have created, maintaining data integrity and privacy.

```python
# Display the list of classes
session = Session()
classes = session.query(Class).filter_by(lecturer_id=st.session_state.logged_in_lecturer.id).all()
session.close()

st.subheader("Classes")
for class_ in classes:
    st.write(f"Course Name: {class_.course_name}")
    st.write(f"Class ID: {class_.class_id}")
    st.write(f"Venue: {class_.venue}")
    st.write(f"Number of Students: {class_.num_students}")
    st.write(f"Attendance: {class_.attendance}")
    st.write("---")
```

Fig. 12 Lecturer Module Creating a Class code

The class module further enhances the lecturer experience by providing a seamless overview of all the classes they have created. The application displays a concise list of classes, along with relevant details such as course name, class ID, venue, number of students, and current attendance count. This overview empowers lecturers to stay informed about their classes and monitor attendance levels easily.

Furthermore, the lecturer module lays the foundation for future enhancements, such as attendance tracking algorithms, geofencing implementation, and integration with mobile applications or wearable devices. By building a robust and modular architecture, the application can adapt and evolve to meet the ever-changing needs of educational institutions and lecturers alike. In summary, the lecturer module serves as the backbone of the system, enabling lecturers to effortlessly manage their classes, track attendance, and leverage the power of geofencing technology.

Its user-friendly interface, secure authentication, and robust database integration ensure a seamless experience for lecturers, while paving the way for future enhancements and expansions.

B. Student Module

The student module is an integral component of the Attendance Management System, enabling students to seamlessly interact with the system and mark their attendance for respective classes. This module is designed with a user-friendly interface and incorporates advanced geofencing technology to ensure accurate attendance tracking. The classes module is a comprehensive section dedicated to displaying the available classes and enabling students to mark their attendance. The application presents a list of classes, including relevant details such as course name and venue. Students are required to enter the unique class ID provided by the lecturer to ensure proper identification and authentication. Upon entering the correct class ID, students are prompted to take a picture using their device's camera. This picture serves as a visual record and verification of the student's presence during the attendance marking process. Additionally, students are required to input their current location by providing latitude and longitude coordinates.

Leveraging the power of geofencing technology, the application verifies the student's proximity to the designated class location. If the student is within the predefined geofence radius, typically set to 100 meters, their attendance is marked as successful. This approach ensures the integrity of the attendance data and prevents potential abuse or fraudulent activity.

The attendance data, including the student's ID, class ID, timestamp, location coordinates, and captured image, is securely stored in the Attendance model within the database. This information can later be assessed and analyzed by authorized personnel for attendance tracking, reporting, and other administrative purposes.

Furthermore, the student module lays the foundation for future enhancements, such as real-time attendance notifications, integration with mobile applications or wearable devices, and advanced data analytics. By building a modular and extensible architecture, the application can adapt and evolve to meet the ever-changing needs of educational institutions and students alike.

In summary, the student module plays a crucial role in the system, empowering students to mark their attendance seamlessly while leveraging the power of geofencing technology. Its user-friendly interface, secure authentication, and robust database integration ensure a smooth experience for students while paving the way for future enhancements and expansions.

```
elif choice == "Signup":
    st.subheader("Student Signup")
    name = st.text_input("Full Name")
    username = st.text_input("Username")
    password = st.text_input("Password", type='password')

    if st.button("Signup"):
        # Create a new student
        student = Student(name=name, username=username, password=password)
        session = Session()
        session.add(student)
        session.commit()
        session.close()
        st.success("Signup successful!")

elif choice == "Login":
    st.subheader("Student Login")
    username = st.text_input("Username")
    password = st.text_input("Password", type='password')

    if st.button("Login"):
        # Validate the login credentials
        session = Session()
        student = session.query(Student).filter_by(username=username, password=password).first()
        session.close()

        if student:
            st.success("Login successful!")
            st.session_state.logged_in_student = student
            st.experimental_rerun()
        else:
            st.error("Invalid username or password")
```

Fig. 12 Student Module

```
for class_ in classes:
    st.write(f"Course Name: {class_.course_name}")
    st.write(f"Venue: {class_.venue}")
    class_id = st.text_input(f"Enter Class ID for {class_.course_name}", key=f"class_id_{class_.id}")
    if st.button(f"Mark Attendance for {class_.course_name}", key=f"mark_attendance_{class_.id}"):
        if class_id == class_.class_id:
            # Take student's picture
            picture = st.camera_input("Take a picture")
            if picture:
                # Get student's location
                latitude = st.number_input("Enter your latitude")
                longitude = st.number_input("Enter your longitude")

                # Check if student is within the class geofence
                class_location = (class_.latitude, class_.longitude)
                student_location = (latitude, longitude)
                if distance(class_location, student_location).meters <= 100:  # Assuming a 100-meter radius
                    # Save attendance
                    attendance = Attendance(
                        student_id=st.session_state.logged_in_student.id,
                        class_id=class_id,
                        timestamp=datetime.datetime.now(),
                        latitude=latitude,
                        longitude=longitude,
                        image=picture.name
                    )
                    session = Session()
                    session.add(attendance)
                    session.commit()
                    session.close()
                    st.success("Attendance marked successfully!")
                else:
                    st.warning("You are not within the class geofence.")
        else:
            st.error("Invalid Class ID")
    st.write("---")
```

Fig. 13 Student Attendance Module Code

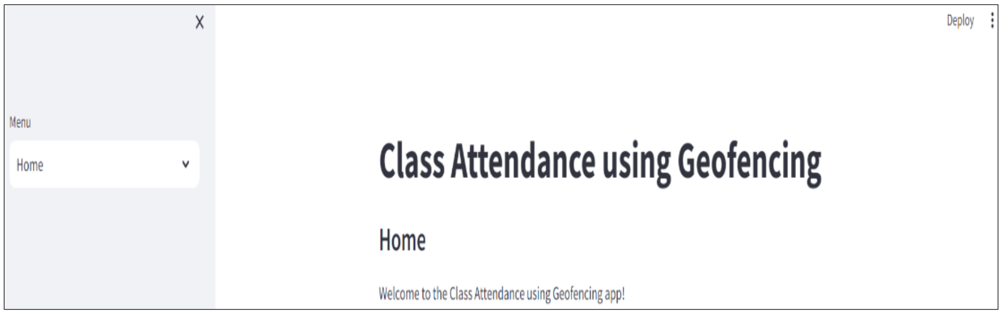

Fig. 14 System Home Page

C. System Evaluation

The signup process is a crucial step for both lecturers and students to gain access to the system. For lecturers, the signup module allows them to register by providing their name, desired username, and a secure password. This process ensures that each lecturer has a unique identity within the system, preventing potential conflicts or duplication.

Similarly, the student signup module enables new students to enrol by providing their full name, preferred username, and a secure password. This not only creates an account for the student but also serves as a means of authentication, ensuring that only registered individuals can access the system and mark their attendance.

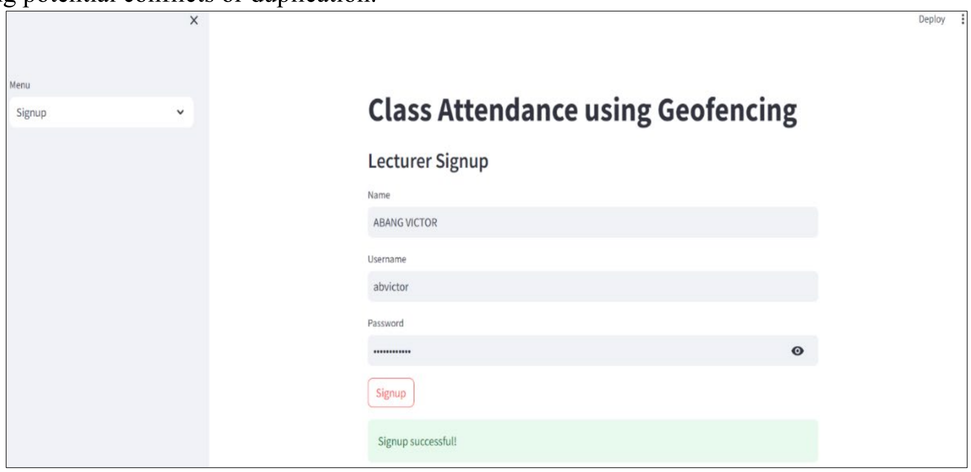

Fig. 15 Lecturer Signup

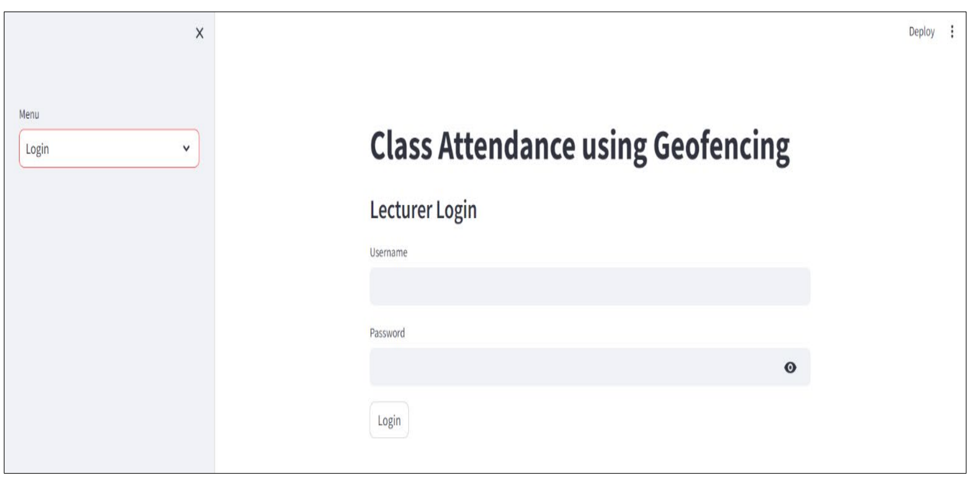

Fig. 16 Lecturer Sign in

Once registered, both lecturers and students can securely access the system through the login function. The lecturer login module prompts users to enter their username and password, which are then validated against the stored credentials in the database. Upon successful authentication, lecturers gain access to the class module, where they can manage and oversee their respective classes.

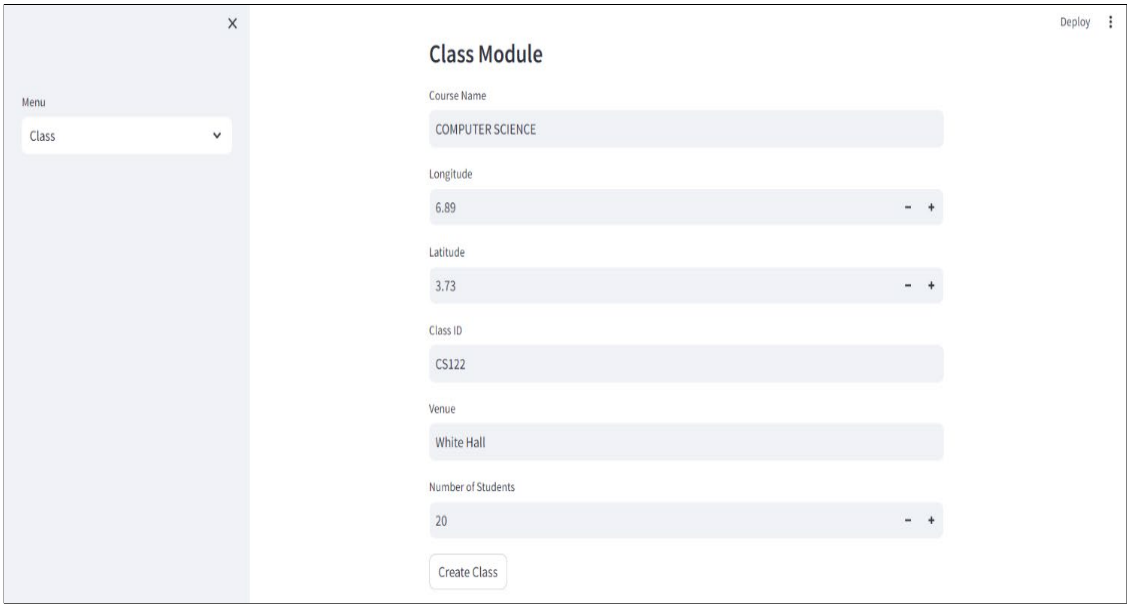

Fig. 17 System Class Module

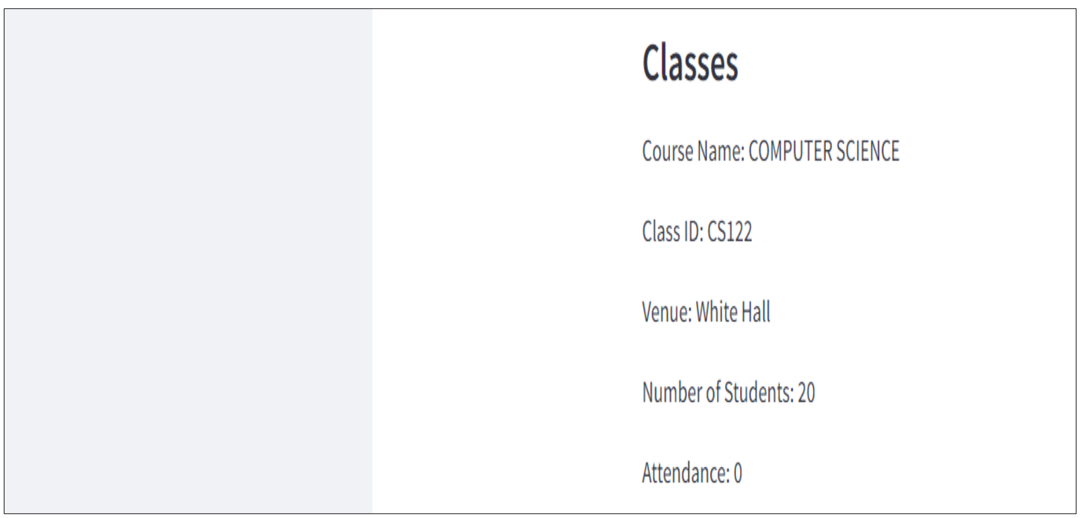

Fig. 18 System Class Schedule

Similarly, the student login module requires students to enter their unique username and password. If the provided credentials are valid, students are granted access to the classes module, where they can view and mark their attendance for the respective courses.

The class module is the heart of the lecturer's experience within the system. This module empowers lecturers to create and manage their classes, providing a user-friendly interface for inputting essential class details. Lecturers can specify the course name, venue, a unique class ID, and the number of students enrolled.

One of the standout features of the class module is the integration of geofencing technology. Lecturers are required to provide the longitude and latitude coordinates of the class location, enabling precise attendance tracking based on student proximity to the designated area. This innovative approach ensures the integrity of the attendance data and prevents potential abuse or fraudulent activity.

Once a class is created, the system securely stores the class information in the database, associating it with the respective lecturer. This association ensures that lecturers can only access and manage the classes they have created, maintaining data privacy and integrity. Furthermore, the class module provides lecturers with a comprehensive overview of all their created classes, displaying relevant details such as course name, class ID, venue, number of students, and current attendance count. This overview empowers lecturers to stay informed about their classes and monitor attendance levels.

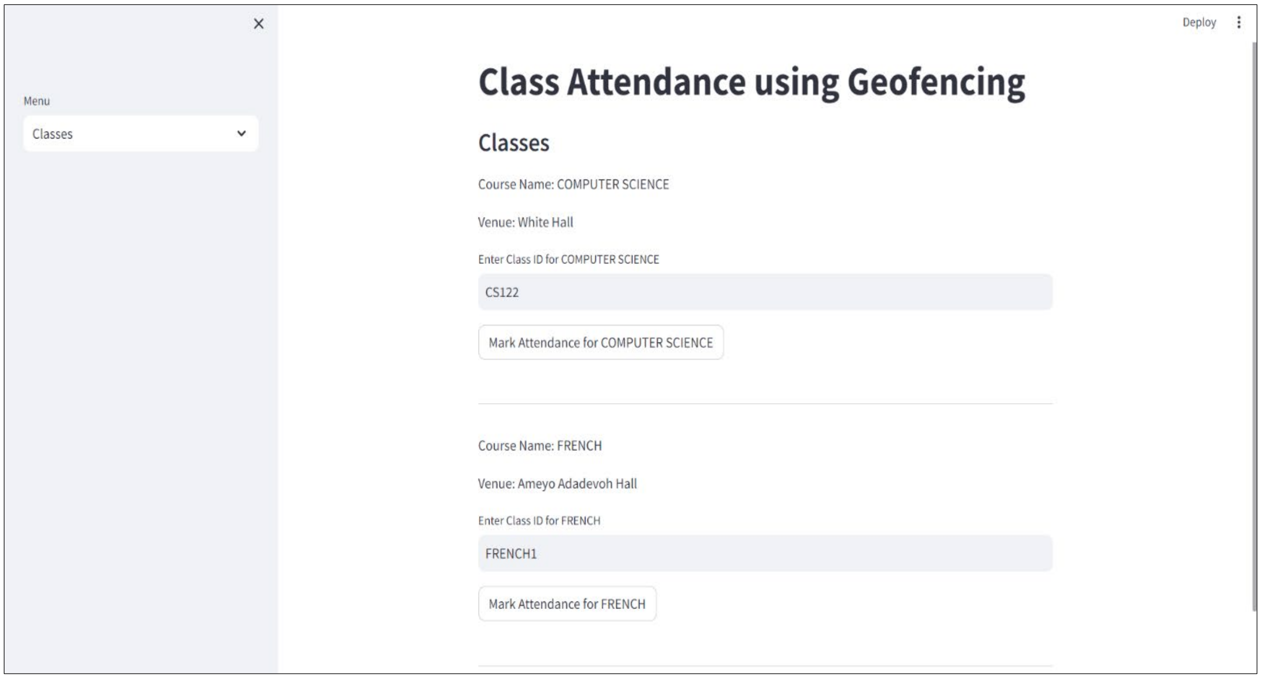

Fig. 19 System Class Attendance Module

From the student's perspective, the classes module serves as a gateway to accessing and marking their attendance for the respective courses. This module presents a list of available classes, allowing students to easily identify the courses they are enrolled in.

To mark their attendance, students are required to enter the unique class ID provided by the lecturer. This step ensures proper identification and authentication, preventing unauthorized individuals from marking attendance for classes they are not enrolled in.

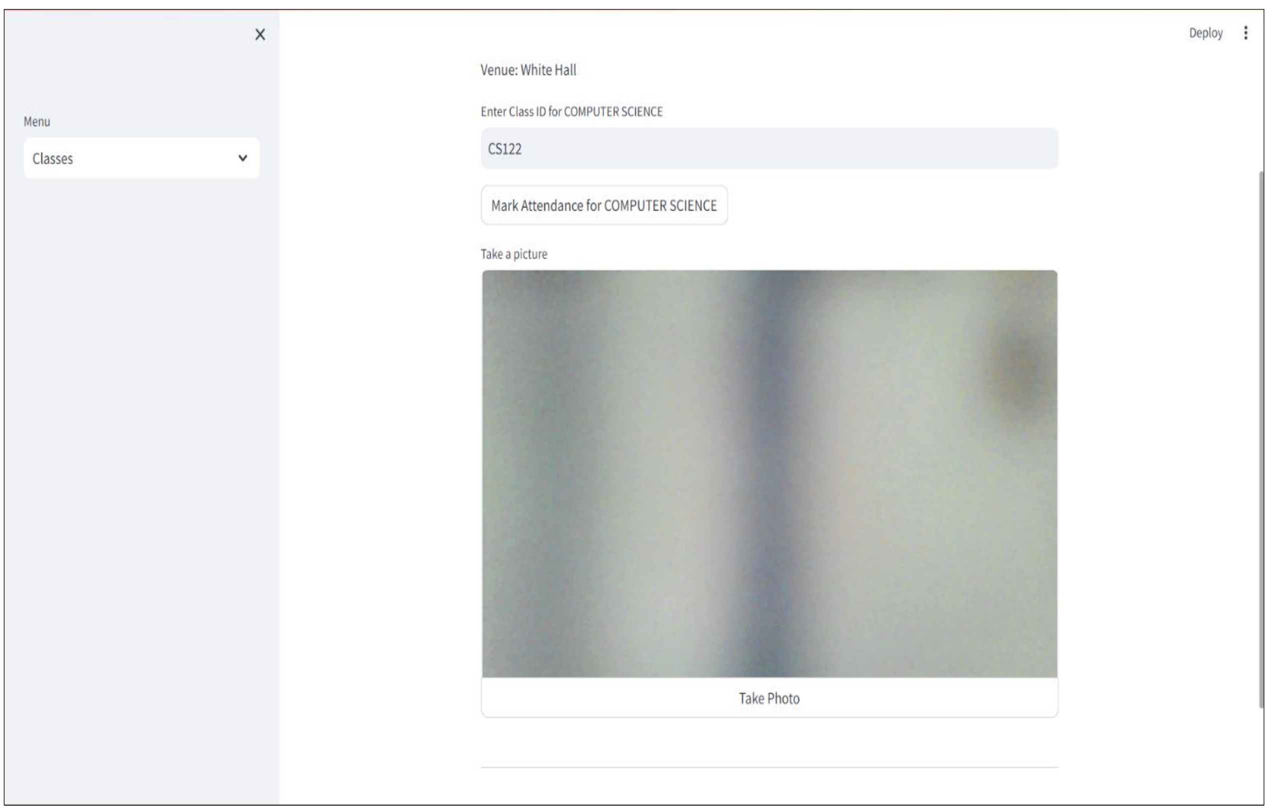

Fig. 20 System Class Attendance Module

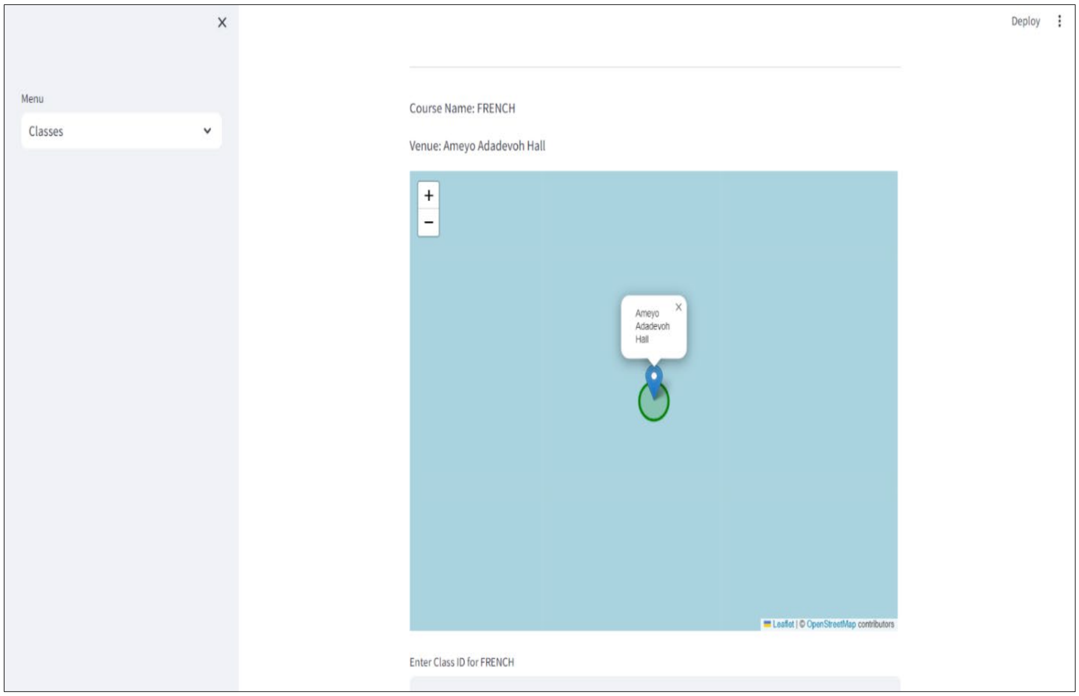

Fig. 21 System Class Attendance Module

The attendance module is the core functionality that enables students to mark their presence for a particular class. Upon entering the correct class ID, students are prompted to take a picture using their device's camera. This picture serves as a visual record and verification of the student's presence during the attendance marking process. Additionally, students are required to input their current location by providing latitude and longitude coordinates. Leveraging the power of geofencing technology, the system verifies the student's proximity to the designated class location. If the student is within the predefined geofence radius, typically set to 100 meters, their attendance is marked as successful.

The attendance data, including the student's ID, class ID, timestamp, location coordinates, and captured image, is securely stored in the database. This information can later be accessed and analyzed by authorized personnel for attendance tracking, reporting, and other administrative purposes.

Overall, the Attendance Management System offers a comprehensive set of functions that cater to the needs of both lecturers and students. From secure registration and authentication to class management and attendance tracking, the system leverages modern technologies and innovative approaches to streamline the attendance process while ensuring data integrity and privacy.

V. CONCLUSION

This chapter offers a comprehensive evaluation of the system, highlighting its achievements, strengths, and potential impact on educational institutions. It emphasizes the project's adherence to best practices in software development, data security, and user-friendly design. Additionally, this chapter discusses the project's scalability and potential for future enhancements, positioning it as a valuable resource for educational institutions seeking to embrace the latest technologies and improve the overall learning experience. Throughout these chapters, the project demonstrates a comprehensive approach to addressing the challenges of class attendance tracking, leveraging modern technologies and innovative solutions.

By combining secure authentication, geofencing capabilities, and a user-friendly interface, the Attendance Management System offers a reliable, accurate, and efficient solution for educational institutions, streamlining the attendance process while ensuring data integrity and privacy. The Attendance Management System is a remarkable achievement in leveraging modern technology to revolutionize the way class attendance is tracked and managed. By seamlessly integrating geofencing capabilities with a robust and user-friendly system, this project has successfully addressed the longstanding challenges associated with traditional attendance-taking methods.

Throughout the development process, a strong emphasis was placed on security, ease of use, and the incorporation of cutting-edge technologies. The implementation of secure authentication mechanisms, such as unique usernames and encrypted passwords, ensures that only authorized individuals can access the system, safeguarding the integrity and privacy of the attendance data. The modular and extensible architecture of the application, built upon industry-standard libraries and frameworks like SQLAlchemy and Streamlit, allows for seamless integration of various components while adhering to best practices in software development. This approach not only enhances the system's scalability and maintainability but also paves the way for future enhancements and expansions. One of the standout features of this project is the integration of geofencing technology, which enables precise attendance tracking based on the student's proximity to the designated class location. By leveraging location-based services and advanced algorithms, the system verifies the student's presence within a predefined radius, ensuring the accuracy and reliability of attendance data.

This innovative approach eliminates the potential for abuse or fraudulent activities, addressing a long-standing challenge in traditional attendance-taking methods. The user-friendly interface, implemented through the powerful Streamlit library, provides a seamless experience for both lecturers and students. Lecturers can effortlessly create and manage classes, input essential class details, and monitor attendance levels. Students, on the other hand, can easily mark their attendance by providing the required information and visual verification through captured images. Furthermore, the application's comprehensive data storage and retrieval capabilities, facilitated by SQLAlchemy's powerful query features, enable authorized personnel to access and analyze attendance data for reporting, auditing, and other administrative purposes. This feature enhances the overall efficiency and transparency of the attendance management process. In conclusion, this system is a testament to the power of innovation and technological advancements in the field of education.

By addressing the longstanding challenges of traditional attendance-taking methods, this project offers a secure, accurate, and user-friendly solution that streamlines the attendance process while ensuring data integrity and privacy. Its modular design and adherence to best practices pave the way for future enhancements and expansions, positioning it as a valuable resource for educational institutions seeking to embrace the latest technologies and improve the overall learning experience [28]. It is recommended to explore the possibility of integrating the system with existing Learning Management Systems (LMS) used by educational institutions. By establishing a seamless connection between the attendance tracking system and the LMS, valuable attendance data can be automatically synchronized and made available to both lecturers and students within the familiar environment of the LMS.

This integration would enhance the overall user experience by eliminating the need to switch between multiple applications, providing a centralized platform for accessing attendance records alongside other course materials, assignments, and grades. Additionally, it would enable lecturers to easily monitor and analyze attendance data in conjunction with student performance, potentially identifying correlations and taking appropriate actions to support struggling students. Furthermore, integration with the LMS could facilitate automated attendance reporting and notifications, streamlining administrative tasks and ensuring timely communication with students regarding their attendance status. This feature could be particularly beneficial for institutions with strict attendance policies, allowing for early intervention and support mechanisms to be implemented when attendance issues arise. By leveraging the existing infrastructure and user base of the LMS, the system could seamlessly integrate into the educational ecosystem, enhancing its adoption rate and providing a comprehensive solution for attendance tracking and overall student engagement.

Declaration of Conflicting Interests

The authors declare no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

Use of Artificial Intelligence (AI)-Assisted Technology for Manuscript Preparation

The authors confirm that no AI-assisted technologies were used in the preparation or writing of the manuscript, and no images were altered using AI.

REFERENCES

[1] N. S. Ali, A. H. Alhilali, H. D. Rjeib, H. Alsharqi, and B. Al Sadawi, "Automated attendance management systems: systematic literature review," *Int. J. Technol. Enhanc. Learn.*, vol. 14, no. 1, p. 37, 2022, doi: 10.1504/IJTEL.2022.120559.

[2] A. Ardebili, A. Latifian, C. F. Aziz, R. H. BinSaeed, S. M. Alizadeh, and E. V. Kostyrin, "A comprehensive and systematic literature review on the employee attendance management systems based on cloud computing," *J. Manag. Organ.*, vol. 29, no. 4, pp. 679-696, Jul. 2023, doi: 10.1017/jmo.2022.63.

[3] P. Bijsmans and A. H. Schakel, "The impact of attendance on first-year study success in problem-based learning," *High. Educ.*, vol. 76, no. 5, pp. 865-881, Nov. 2018, doi: 10.1007/s10734-018-0243-4.

[4] A. Puckdeevongs, N. K. Tripathi, A. Witayangkurn, and P. Saengudomlert, "Classroom attendance systems based on Bluetooth low energy indoor positioning technology for smart campus," *Information*, vol. 11, no. 6, p. 329, Jun. 2020, doi: 10.3390/info110 60329.

[5] O. J. Adeniyi, O. D. Adekola, B. G. Akwaronwu, A. G. Abiodun, and I. O. Eweoya, "Exploring the link between roommate compatibility and academic outcomes: A systematic review of personality-based matching systems," *Asian Journal of Computer Science and Technology*, vol. 13, no. 2, pp. 29-40, Oct. 2024, doi: 10.70112/ajcst-2024.13.2.4275.

[6] A. A. Abiona, M. O. Olarinde, and M. O. Ajinaja, "Enhanced donkey and smuggler optimization algorithm for holistic student admissions in polytechnics," *Asian Journal of Computer Science and Technology*, vol. 13, no. 2, pp. 56-60, Nov. 2024, doi: 10.70112/ajcst-2024.13.2.4302.

[7] K. Alagasan, M. H. Alkawaz, A. Iqbal Hajamydeen, and M. N. Mohammed, "A review paper on advanced attendance and monitoring systems," in *2021 IEEE 12th Control and System Graduate Research Colloquium (ICSGRC)*, IEEE, Aug. 2021, pp. 195-200, doi: 10.1109/ICSGRC53186.2021.9515249.

[8] 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 & Digital Technologies (IC2SDT)*, IEEE, Aug. 2024, pp. 137-141, doi: 10.1109/IC2SDT62152.2024.10696345.

[9] T.-W. Chiang et al., "Development and evaluation of an attendance tracking system using smartphones with GPS and NFC," *Appl. Artif. Intell.*, vol. 36, no. 1, pp. 1118-1137, Dec. 2022, doi: 10.1080/088395 14.2022.2083796.

[10] B. Suraj Aravind, A. Vandana, R. Bharath Kumar Reddy, S. Patchigolla, and G. Harshitha, "Custom reporting framework for Moodle LMS based e-learning," *Asian Journal of Computer Science and Technology*, vol. 12, no. 1, pp. 7-10, Apr. 2023, doi: 10.51983/ ajcst-2023.12.1.3521.

[11] K. A. Yousif Yaseen, "Digital education: The cybersecurity challenges in the online classroom (2019-2020)," *Asian Journal of Computer Science and Technology*, vol. 11, no. 2, pp. 33-38, Dec. 2022, doi: 10.51983/ajcst-2022.11.2.3450.

[12] K. Tobin, O. Heidari, C. Volpi, S. Sodder, and D. Duncan, "Use of geofencing interventions in population health research: A scoping review," *BMJ Open*, vol. 13, no. 8, p. e069374, Aug. 2023, doi: 10.1136/bmjopen-2022-069374.

[13] A. Rao, A. Vaidya, and Y. Narayanan, "Faculty attendance system using geofence, face-recognition and admin portal," *Turkish J. Comput. Math. Educ.*, vol. 13, no. 03, pp. 819-829, 2022.

[14] Enikuomehin and Dosumu, "Geofencing based attendance monitoring system," *Res. Inven. Int. J. Eng. Sci.*, vol. 11, no. 1, pp. 2319-6483, 2021, [Online]. Available: www.researchinventy.com.

[15] R. Shinde, A. Nilose, and P. Chandankhede, "Design and development of geofencing based attendance system for mobile application," in *2022 10th International Conference on Emerging Trends in Engineering and Technology - Signal and Information Processing (ICETET-SIP-22)*, IEEE, Apr. 2022, pp. 1-6, doi: 10.1109/ICETET-SIP-2254415.2022.9791781.

[16] S. H. V. K. Ashok, N. D. Divya, F. Samuel, and D. U. Mande, "RFID based attendance system," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 10, no. 12, pp. 2239-2248, Dec. 2022, doi: 10.22214/ijraset.2022.48092.

[17] A. N. Babatunde, A. Oke, R. Babatunde, O. Ibitoye, and E. Jimoh, "Mobile based student attendance system using geo-fencing with timing and face recognition," *Adv. Multidiscip. Sci. Res. J. Publ.*, vol. 10, no. 1, pp. 75-90, 2022, doi: 10.22624/aims/maths/v10n1p8.

[18] M. Kohana and S. Okamoto, "A location-based attendance confirming system," in *2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA)*, IEEE, Mar. 2016, pp. 816-820, doi: 10.1109/AINA.2016.119.

[19] M. S. Uddin, S. M. Allayear, N. C. Das, and F. A. Talukder, "A location based time and attendance system," *Int. J. Comput. Theory Eng.*, vol. 6, no. 1, pp. 36-38, 2014, doi: 10.7763/IJCTE.2014.V6.832.

[20] A. Gedam, V. Chalwadi, V. Dangre, S. Musale, and P. Mane, "Implementation of GPS based staff attendance system using geo-fence technique," *Int. J. Innov. Res. Sci. Eng. Technol.*, vol. 10, no. 6, 2021, doi: 10.15680/IJIRSET.2021.1006044.

[21] M. Makhtar, R. Rosly, S. A. Fadzli, S. N. W. Shamsuddin, and A. A. Jamal, "Implementation of mobile attendance application using geo-fence technique," *ARPN J. Eng. Appl. Sci.*, vol. 11, no. 5, pp. 3391-3395, 2016.

[22] S. Sanjay Galgale, J. Sayyad Yamakanamardi, S. R. Koli, and M. J. Desai, "The location based attendance system," *Int. Res. J. Mod. Eng. Technol. Sci.*, vol. 5, no. 4, pp. 924-929, 2023, [Online]. Available: www.irjmets.com.

[23] M. S. Mubarak Alburaiki, G. Md Johar, R. A. Abbas Helmi, and M. Hazim Alkawaz, "Mobile based attendance system: Face recognition and location detection using machine learning," in *2021 IEEE 12th Control and System Graduate Research Colloquium (ICSGRC)*, IEEE, Aug. 2021, pp. 177-182, doi: 10.1109/ICSGRC53186.2021.9515221.

[24] S. Tasleem, P. Bano, and H. Rahman, "Students attendance management system based on face recognition," *Int. J. STEM Educ.*, vol. 1, no. 3, pp. 66-74, 2020.

[25] N. K. Jayant and S. Borra, "Attendance management system using hybrid face recognition techniques," in *2016 Conference on Advances in Signal Processing (CASP)*, IEEE, Jun. 2016, pp. 412-417, doi: 10.1109/CASP.2016.7746206.

[26] K. Alhanaee, M. Alhammadi, N. Almenhali, and M. Shatnawi, "Face recognition smart attendance system using deep transfer learning," *Procedia Comput. Sci.*, vol. 192, pp. 4093-4102, 2021, doi: 10.1016/j.procs.2021.09.184.

[27] S. M. Bah and F. Ming, "An improved face recognition algorithm and its application in attendance management system," *Array*, vol. 5, no. Mar., p. 100014, Mar. 2020, doi: 10.1016/j.array.2019.100014.

[28] K. Mohammed, A. S. Tolba, and M. Elmogy, "Multimodal student attendance management system (MSAMS)," *Ain Shams Eng. J.*, vol. 9, no. 4, pp. 2917-2929, Dec. 2018, doi: 10.1016/j.asej.2018.08.002.