

# Smart Classroom Monitoring System for Faculty Presence Detection Using Multi-Modal IoT and AI Techniques

Manikanta Kinjarapu<sup>1</sup>, Bhavana AvaniGadda<sup>2</sup>, Anuradha Kolla<sup>3</sup>, Badar Shaik<sup>4</sup>

<sup>1,2,3</sup>Students, Department of Computer Science and Engineering Vikas College of Engineering and Technology Vijayawada, Andhra Pradesh, India

<sup>4</sup>Head of the Department, Department of Computer Science and Engineering Vikas College of Engineering and Technology Vijayawada, Andhra Pradesh, India

DOI: <https://doi.org/10.51244/IJRSI.2026.1303000231>

Received: 22 March 2026; Accepted: 27 March 2026; Published: 20 April 2026

## ABSTRACT

In many educational institutions, ensuring that faculty members are consistently present in classrooms remains a challenge. Traditional attendance systems are often limited to entry-level verification and fail to monitor real-time presence during lectures. This paper presents a Smart Classroom Monitoring System that uses a combination of IoT sensors and basic AI techniques to accurately detect faculty presence inside a classroom.

The system combines motion detection using a PIR sensor with identity verification methods such as face detection, voice activity, and mobile device detection. A Raspberry Pi is used as the core processing unit, which collects sensor data and applies a decision-making algorithm to determine whether the faculty is present or absent. If the system detects continuous absence beyond a defined time threshold, it automatically sends an alert to the concerned authority.

The proposed system is designed to work efficiently even in real-world classroom conditions where lighting, movement, and noise vary. By combining multiple detection methods, the system reduces false alerts and improves reliability. This approach provides a practical and cost-effective solution for automated classroom monitoring.

**Keywords:** Smart Classroom, IoT, Raspberry Pi, Face Detection, Automation, Monitoring System

## INTRODUCTION

### Background

With the rapid growth of technology in education, classrooms are becoming smarter in terms of teaching tools and digital resources. However, administrative monitoring, especially tracking faculty presence in classrooms, is still largely manual. In many institutions, once a faculty member marks attendance, there is no system to ensure that they remain present throughout the lecture.

This often leads to situations where classes are left unattended without immediate notice. Such issues affect students' learning experience and reduce overall academic efficiency.

### Problem Statement

Existing methods such as manual checking or CCTV monitoring require human involvement and are not practical for large institutions. Systems based on a single sensor, like motion detectors or RFID, are not reliable because they cannot accurately identify whether the detected presence belongs to the faculty.

The main problem addressed in this project is:

---

**To develop a system that can automatically and accurately detect whether a faculty member is present inside a classroom using multiple verification methods, and generate alerts in case of absence.**

### **Objective of the Project**

The objectives of this system are:

- To detect human presence using motion sensors
- To verify identity using face detection and other methods
- To reduce false alerts using multi-factor validation
- To send alerts automatically when absence is confirmed
- To provide a real-time monitoring dashboard

The novelty of this work lies in the integration of multiple low-cost sensors with an edge computing approach to provide a practical, real-time faculty monitoring solution. Unlike existing systems that rely on a single detection method, this system combines multiple verification techniques to improve accuracy while maintaining low computational cost and real-world applicability.

### **LITERATURE SURVEY**

Several systems have been developed in the past to automate classroom monitoring, but each has its own limitations.

Motion-based systems using PIR sensors are widely used due to their simplicity and low cost. However, they cannot differentiate between different individuals. Any movement in the classroom can trigger the sensor, leading to inaccurate results.

RFID-based systems are commonly used for attendance, but they require manual interaction. Once a person marks attendance, the system assumes continuous presence, which may not be true.

Face detection systems provide better accuracy as they visually identify individuals. However, they depend heavily on lighting conditions and camera placement. Continuous use of such systems also increases computational load.

Recent approaches include using Wi-Fi or Bluetooth signals to detect nearby devices. While this method is non-intrusive, it is not fully reliable because devices can be left behind or turned off.

From this study, it is clear that no single method is sufficient. A combination of multiple techniques can improve overall system reliability. This project is based on this idea of combining different sensors and verification methods.

### **Comparison with Existing Systems**

Compared to traditional systems such as PIR-only or RFID-based attendance systems, the proposed system provides improved accuracy and reliability.

PIR-based systems cannot distinguish between individuals, while RFID systems are prone to misuse such as proxy attendance. Vision-based systems alone are affected by lighting conditions.

The proposed multi-modal approach integrates motion detection with identity verification methods such as face detection, voice detection, and device detection. This combination significantly reduces false positives and improves robustness.

Therefore, the proposed system offers a more reliable and practical solution for smart classroom monitoring.

## PROPOSED SYSTEM

The proposed system features an end-to-end edge-computing architecture capable of localized data processing without requiring continuous high-bandwidth cloud connections, thereby preserving privacy and reducing latency.

### System Architecture

The SCMSF architecture operates as a hierarchical, multi-tiered pipeline:

#### Physical Layer (Hardware):

A Raspberry Pi 4 orchestrates data collection from a PIR sensor, a Webcam, a USB Microphone, and its onboard Wi-Fi module.

#### Perception Layer (Sensor Modules):

Python-based software modules handle raw data parsing.

- PIRSensor: Detects raw infrared changes.
- CameraSensor: Executes Haar Cascades or CNN-based face detection.
- AudioSensor: Measures acoustic amplitude and human voice frequencies.
- NetworkSensor: Performs packet sniffing using scapy to detect registered faculty MAC/IP addresses.

#### Cognitive Layer (Decision Engine):

A multi-factor decision engine processes boolean or array outputs from the perception layer and manages system state transitions based on time-bound rules.

#### Application Layer (API and Alerts):

A lightweight Flask API serves real-time status to a web dashboard, while an alert system dispatches notifications via SMTP or APIs upon confirmed absence.

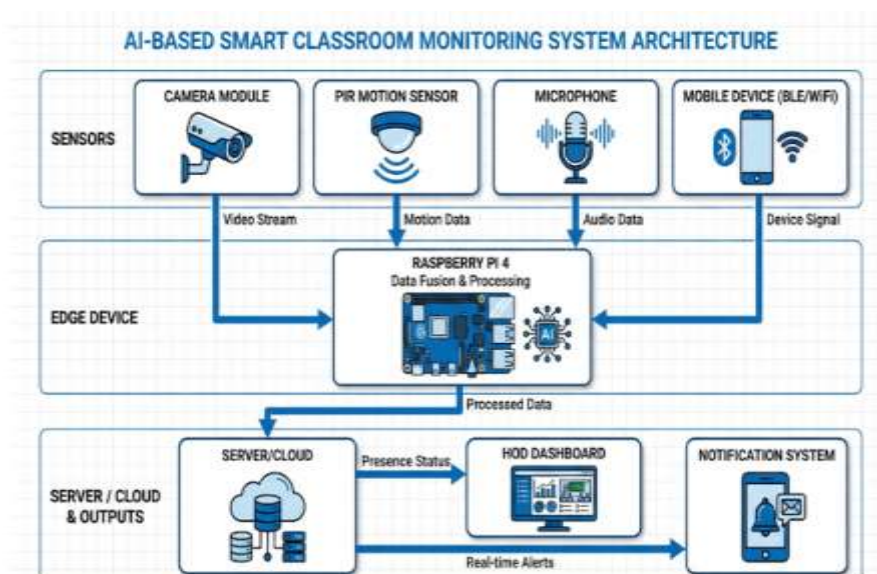


Figure: 1 System Architecture

## Hardware Components

- Microcomputer: Raspberry Pi 4 Model B (4GB RAM) handles concurrent processing of APIs and computer vision tasks.
- Vision: Standard USB Webcam or Pi Camera module mounted facing the lecturer area.
- Auditory: USB microphone for detecting speech near the podium.
- Motion: HC-SR501 PIR sensor positioned to monitor the lecturer's movement area.



Figure:2 Hardware Components

## Multi-Factor Decision Logic

A structured decision logic is implemented using the following variables:

M = Motion Detected

F = Authorized Face Detected

V = Voice Detected

D = Authorized Device Detected

The system determines presence using the following logic:

**Presence = M AND (F OR V OR D)**

Additionally, a time-based tolerance mechanism is incorporated to handle temporary detection failures. The system does not immediately classify absence when identity factors are not detected. Instead, it waits for a predefined threshold duration before triggering an alert.

This improves robustness and ensures that short-term interruptions, such as the faculty turning away from the camera or temporary silence, do not result in false alerts.

Faculty is considered present only when motion is detected along with at least one identity verification factor.

If the presence condition fails continuously for a defined time threshold (for example, 5 minutes), the system transitions to an alert state.

## System Design

### A. Class Diagram

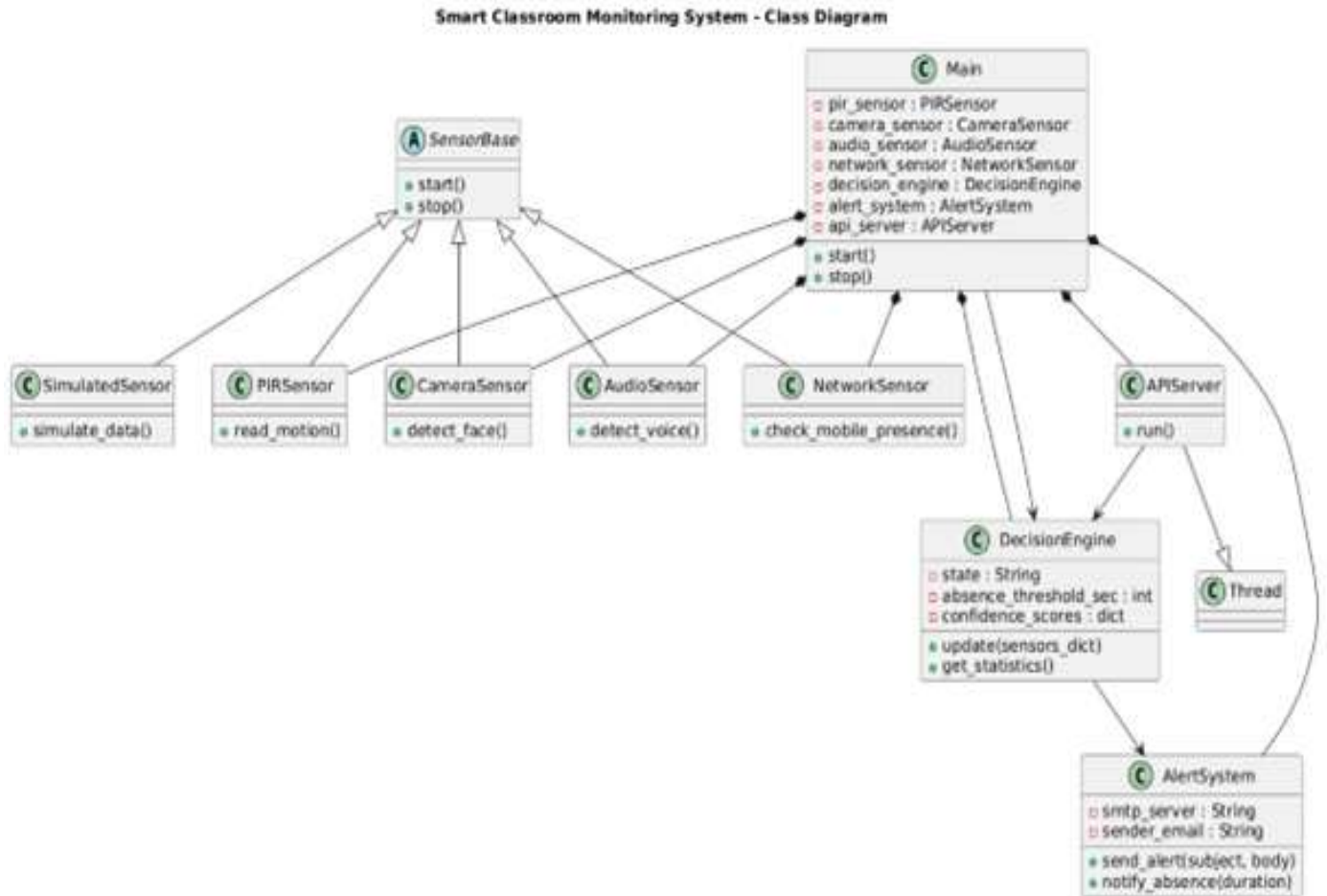


Figure 3: Class Diagram

### B. Use Case Diagram

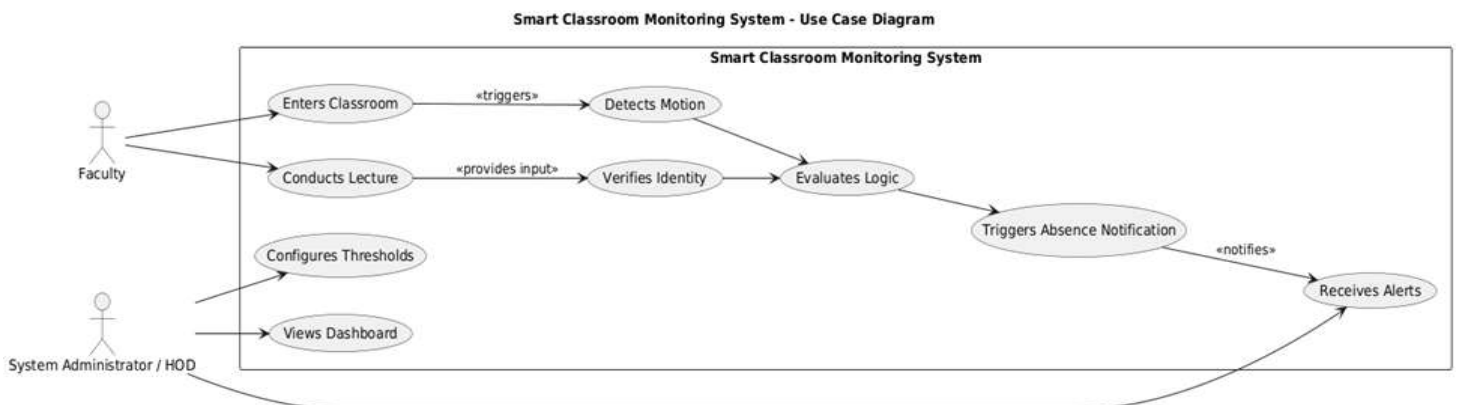


Figure 4: Use Case Diagram

C. Sequence Diagram

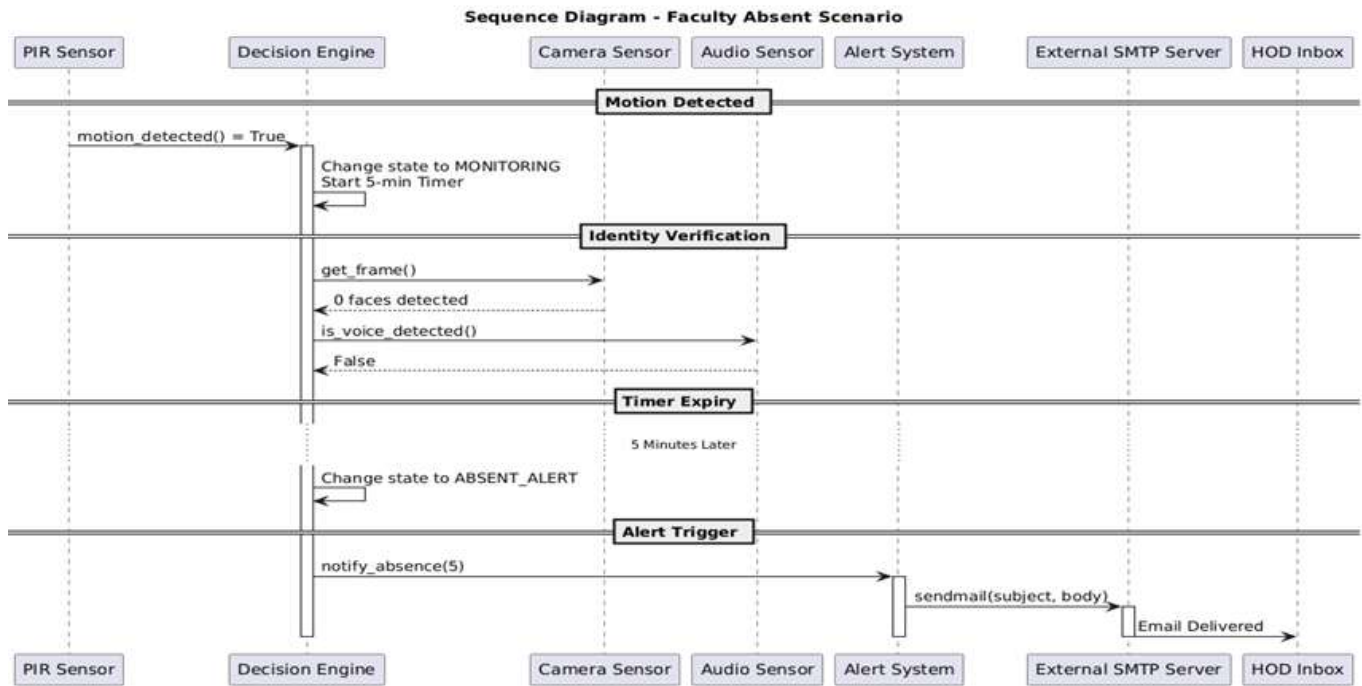


Figure 5: Sequence Diagram

D. Activity Diagram

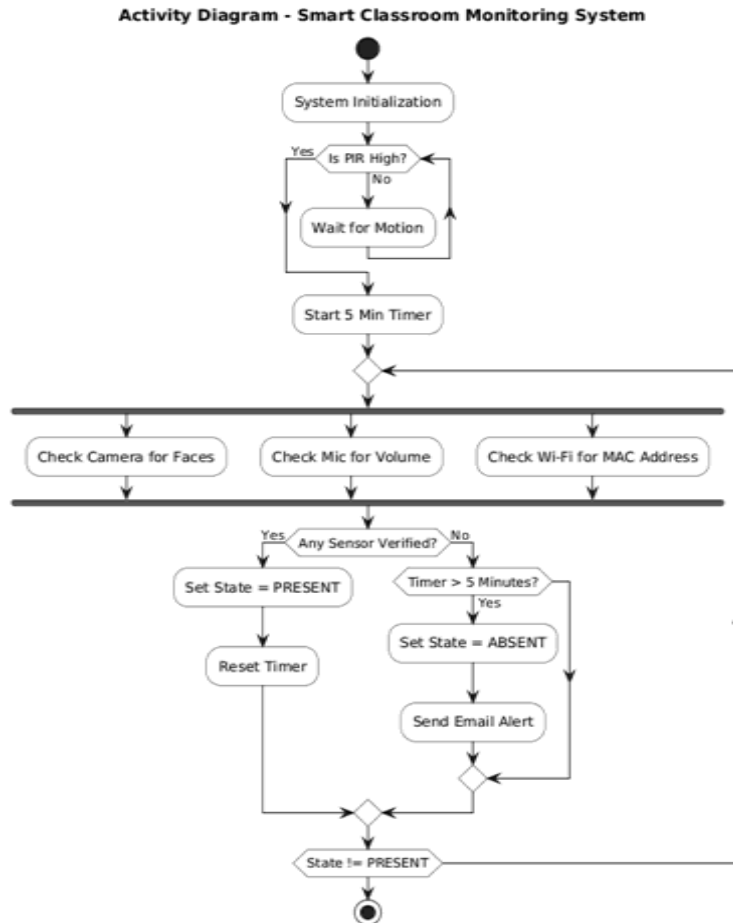


Figure 6: Activity Diagram

Data Flow Diagrams

Level 0 DFD

DFD Level 0 - SCMSF System (Context Diagram)

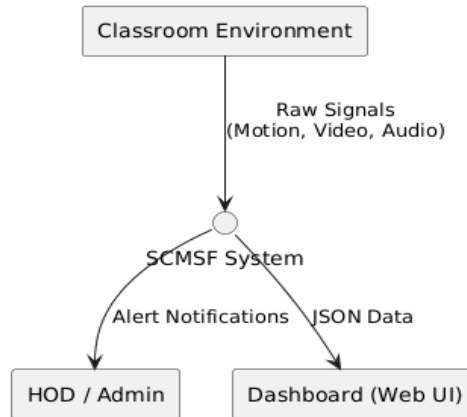


Figure 7: DFD Level 0 – Context Diagram of SCMSF System

The Level 0 DFD represents the SCMSF System as a single central process interacting with external entities. The Classroom Environment provides raw signals such as motion, video, and audio to the system for processing. Based on analysis, the system sends alert notifications to the HOD/Admin and provides real-time JSON data to the web dashboard for monitoring.

Level 1 DFD

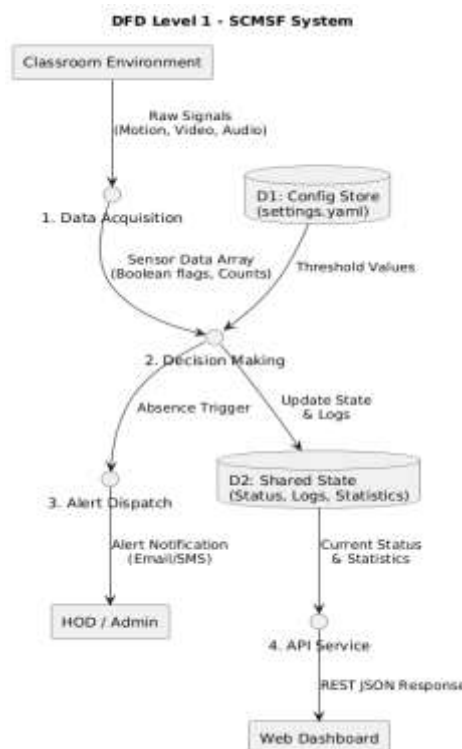


Figure 8: DFD Level 1 – Context Diagram of SCMSF System

The Level 1 DFD illustrates how raw sensor signals are captured by the Data Acquisition process and converted into structured data for the Decision-Making module. The system applies predefined thresholds from the configuration store to determine faculty presence and updates the shared state accordingly. If absence is detected, the Alert Dispatch process notifies the HOD, while the API Service provides real-time status data to the web dashboard.

Level 2 DFD

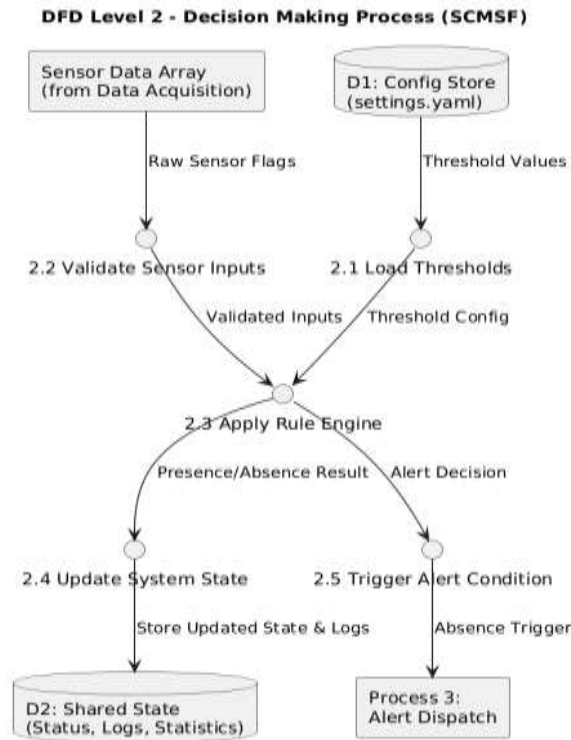


Figure 9: DFD Level 2 – Context Diagram of SCMSF System

DFD Level 2 expands the Decision-Making process of the system into smaller internal functions. It shows how sensor data is validated and compared with threshold values from the configuration store. Based on predefined logic rules, the system determines whether the student is present or absent. The updated status is stored in the shared state database. If absence is detected, an alert is triggered to the Alert Dispatch process.

**METHODOLOGY**

The system operates through a continuous loop that manages sensor inputs and applies decision logic efficiently.

**5.1 Phased Execution Loop**

To reduce unnecessary processing, the system follows a two-phase approach.

**Phase 1: Low-Power Environmental Trigger**

The system continuously monitors the PIR sensor.

- If no motion is detected and the system is in idle state, further processing is skipped.
- If motion is detected, the system proceeds to the next phase.

**Phase 2: High-Power Verification**

Once motion is detected, the system activates other sensors:

- Vision Module: Captures frames and detects faces using OpenCV.
- Audio Module: Measures sound levels and detects human speech.
- Network Module: Checks for registered devices using MAC address detection.

## State Machine and Threshold Logic

The system maintains the following states:

1. IDLE – No motion detected
2. MONITORING – Motion detected, verifying presence
3. PRESENT – Faculty presence confirmed
4. ABSENT – Presence not detected temporarily
5. ABSENT\_ALERT – Absence confirmed after threshold time

To avoid false alerts, the system uses a time-based tolerance mechanism.

If detection temporarily fails, the system waits for a predefined duration before triggering an alert.

If presence is detected again within this period, the timer resets.

---

## Communication and Alert Mechanism

When absence is confirmed:

- An email is generated using SMTP
- The alert includes time, location, and absence duration
- Notifications are sent securely to higher authorities
- The dashboard updates in real-time using a Flask API



Figure 7: System Methodology

## Dataset and Testing Conditions

The system was evaluated using real-time data collected through sensors rather than a pre-existing dataset. The testing was performed in a classroom-like environment using a USB camera, PIR sensor, and microphone.

Different scenarios were simulated, including:

- Faculty entering and leaving the classroom
- Faculty standing still or moving while teaching

- Students moving in the absence of faculty
- Background noise conditions

This approach ensures that the system is validated under practical real-world conditions.

## RESULTS AND IMPLEMENTATION

The system was implemented and tested using Python on a Raspberry Pi setup.

### Dashboard Evaluation

A web-based dashboard successfully displayed real-time system status.

- In idle mode, sensors remained inactive
- When motion was detected, all modules activated sequentially
- Data updates were reflected instantly on the dashboard



Figure 10: Dashboard

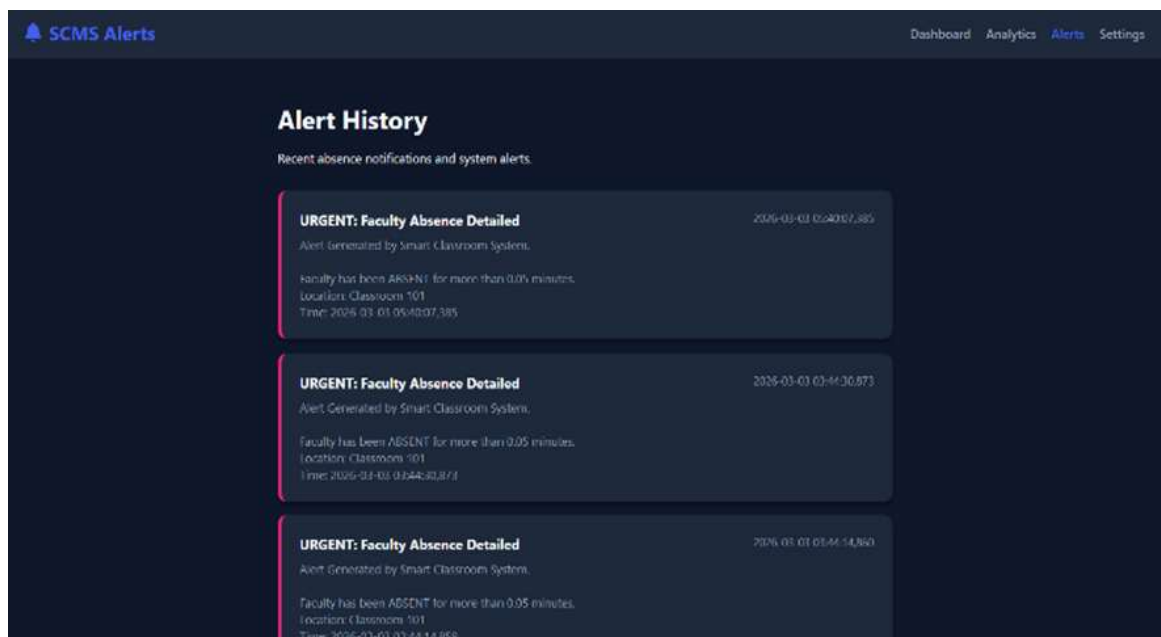


Figure 11: Alert Log

## Reliability Testing

The system was tested under different scenarios:

### Motion | Face | Voice | Device | Result

0 | 0 | 0 | 0 → No presence detected

1 | 0 | 0 | 0 → Monitoring state

1 | 1 | 0 | 0 → Presence confirmed

1 | 0 | 1 | 0 → Presence confirmed

1 | 0 | 0 | 1 → Presence confirmed

The multi-modal approach reduced false positives effectively.

## Alert Performance

- Alerts were triggered only after threshold time
- Email notifications were delivered successfully
- Duplicate alerts were avoided using cooldown logic

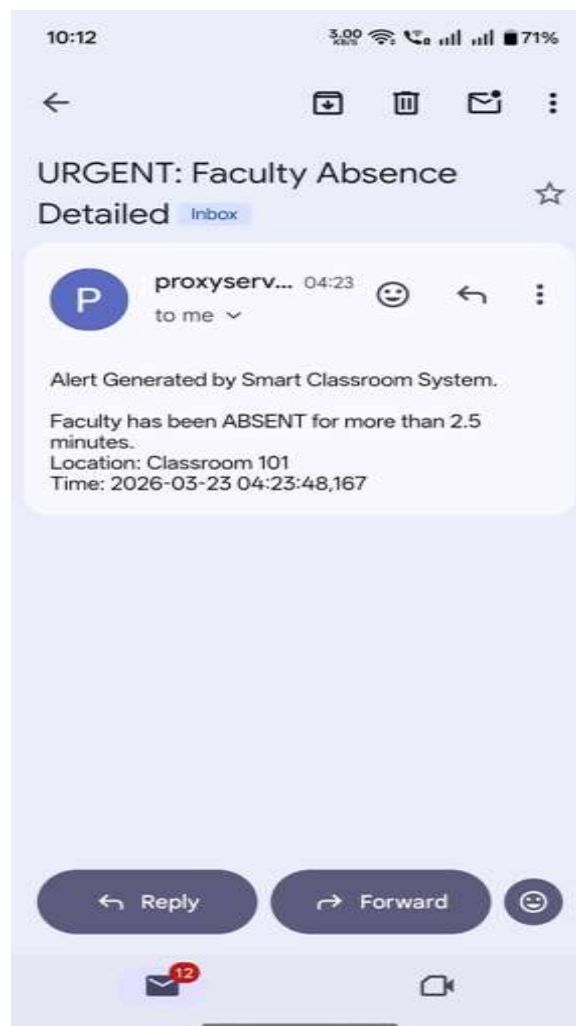


Figure 12: Alert Mail

## Performance Evaluation

To evaluate the effectiveness of the proposed system, multiple real-time experiments were conducted in a controlled classroom environment.

A total of 50 test cases were performed under different scenarios including:

- Faculty present and moving
- Faculty present but stationary
- Students present without faculty
- No human presence

The system achieved an overall accuracy of 88% in detecting faculty presence correctly. The false positive rate was approximately 8%, mainly caused by environmental disturbances and unintended motion detection.

The average system response time (latency) from motion detection to final decision output was observed to be around 1.5 to 2 seconds.

Precision and recall values were estimated as:

Precision = 0.87

Recall = 0.89

These results demonstrate that the multi-modal approach significantly improves detection reliability compared to single-sensor systems.

## CONCLUSION

The proposed Smart Classroom Monitoring System demonstrates a practical and efficient solution for automated faculty presence detection. By integrating multiple sensors and edge computing techniques, the system achieves improved accuracy and reduced false alerts compared to traditional approaches.

The system is suitable for real-world deployment in educational institutions due to its low cost and scalability.

Future work will focus on incorporating machine learning-based decision models, improving facial recognition accuracy, and enabling cloud-based centralized monitoring for large-scale deployment.

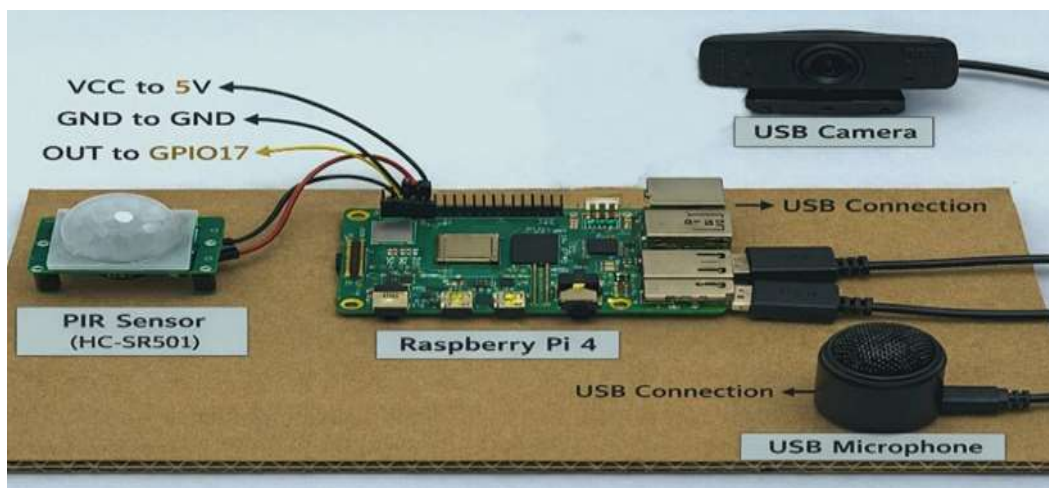


Figure 13: System Hardware Setup

## REFERENCES

1. Biondi, P. (n.d.). Scapy Documentation - Packet Manipulation. Retrieved from <https://scapy.readthedocs.io/>
2. Bradski, G. (2000). The OpenCV Library. Dr. Dobb's Journal of Software Tools.
3. Grinberg, M. (2018). Flask Web Development: Developing Web Applications with Python. O'Reilly Media.
4. H. Rahman et al., "An Embedded Intelligent System for Attendance Monitoring," arXiv preprint arXiv:2406.13694, 2024. [Online]. Available: <https://arxiv.org/abs/2406.13694>
5. Hubert, M. (n.d.). PyAudio Documentation. Retrieved from <https://people.csail.mit.edu/hubert/pyaudio/docs/>
6. M. Joshi et al., "IoT Based Class Monitoring System with Smart Attendance," International Journal of Scientific Research in Engineering and Management, 2024. [Online]. Available: <https://ijsrem.com>
7. P. N. Reddy et al., "IoT Based Attendance Monitoring System Using Facial Recognition," in Proceedings of IEEE Conference on Smart Technologies, 2023. [Online]. Available: ResearchGate
8. Python Software Foundation. (n.d.). Python Language Reference, version 3.x. Retrieved from <https://docs.python.org/3/>
9. R. Kumar and S. Sharma, "Attendance Monitoring System Using Face Recognition," International Journal of Innovative Technology and Research, 2022. DOI: 10.5281/zenodo.7385439
10. Raspberry Pi Foundation. (n.d.). Raspberry Pi Documentation. Retrieved from <https://www.raspberrypi.com/documentation/>
11. V. Sharma and A. Gupta, "Wi-Fi Based Smart Attendance Monitoring System," 2023. [Online]. Available: ResearchGate
12. Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. Accepted Conference on Computer Vision and Pattern Recognition (CVPR).