EAST WEST UNIVERSITY
Department of Computer Science and Engineering

# Project Proposal On

## IoT-Based Fingerprint Attendance System

**Course Title: Internet of Things Course**
**Code: CSE406**



| Submitted By | Student's Id |
|---|---|
| **Saifur Rahman** | **2021-3-60-033** |
| **Mumtahina Akter** | **2021-3-60-137** |
| **Md Tafsidul Islam** | **2022-1-60-271** |
| **Farhana Hoque Nishi** | **2022-1-60-296** |



**Submitted To:**
**Dr. Raihan Ul Islam (DRUI)**
Associate Professor, Dept. of CSE East West University

Date : 25/12/2025

## 1. Abstract

Attendance management is a critical activity in educational institutions and organizations. Traditional attendance methods are often inefficient, time-consuming, and prone to human error and proxy attendance. This project presents an **IoT-Based Fingerprint Attendance System** that uses biometric authentication to ensure accuracy and reliability.

The system is developed using the **ESP8266 NodeMCU** microcontroller and an **R307 fingerprint sensor** for secure user identification. Once a fingerprint is successfully verified, attendance information such as user ID, date, and time is automatically transmitted to **Google Sheets** through a **Google Apps Script Web API** using Wi-Fi connectivity. An **OLED display** provides real-time status messages, while a **buzzer** offers audio feedback for user interaction.

The proposed system reduces manual effort, eliminates proxy attendance, and enables real-time cloud based data storage. It is cost-effective, scalable, and suitable for deployment in schools, offices, and organizations.

## 2. Introduction

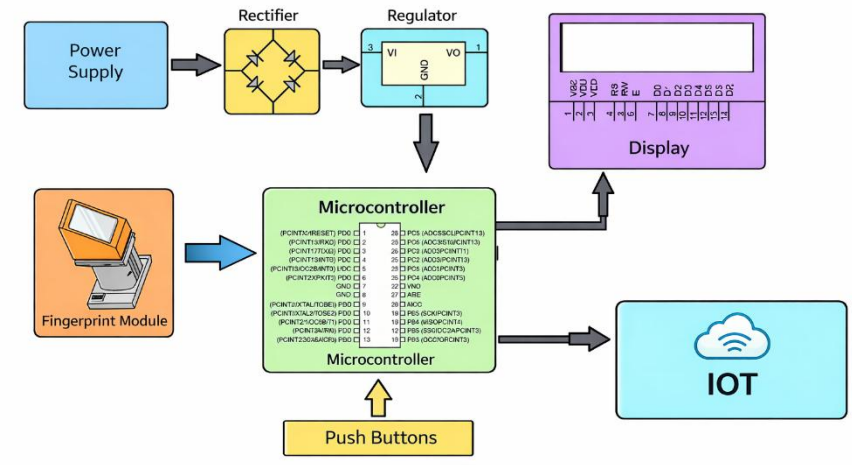
Attendance management is an essential task in educational institutions, offices, and organizations. Traditional attendance systems such as manual registers or card-based systems are time-consuming, error prone, and vulnerable to proxy attendance. To overcome these limitations, this project presents an **IoT Based Fingerprint Attendance System** that ensures secure, accurate, and automated attendance recording.

The system uses biometric fingerprint authentication combined with Internet of Things (IoT) technology. Once a valid fingerprint is detected, attendance data is automatically uploaded to a cloud-based platform, eliminating manual effort and ensuring real-time data availability.

## 3. Block Diagram of IoT-Based Fingerprint Attendance System

The block diagram represents the overall architecture of the IoT-based fingerprint attendance system.

- **Fingerprint Sensor (R307):** Captures the user's fingerprint and sends biometric data to the controller.
- **ESP8266 NodeMCU:** Acts as the central processing unit. It verifies fingerprints, controls output devices, and manages Wi-Fi communication.
- **OLED Display:** Displays system messages such as attendance status.
- **Buzzer (via BC547 Transistor):** Provides audio feedback to the user.
- **Wi-Fi & Cloud (Google Sheets):** Stores attendance data (User ID, Date, Time) in real time.




Colour-coded Block Diagram of IOT-Based Firgnt Attendce System

## 4. Circuit Diagram

The circuit diagram shows the physical interconnection of all hardware components used in the system.

- The **R307 fingerprint sensor** is connected to the ESP8266 using UART communication (TX, RX).
- The **OLED display** communicates with ESP8266 via the I2C protocol (SDA, SCL).
- The **buzzer** is driven through a **BC547 transistor** to protect the microcontroller from excess current.
- **Resistors** are used for current limiting and transistor biasing.
- All components are powered using the ESP8266's regulated power supply.

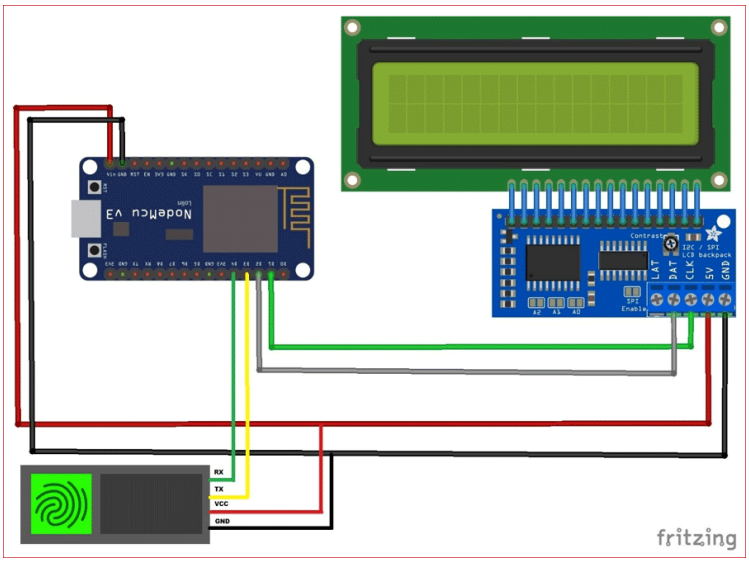



Figure : Circuit Diagram

## 5. Flowchart of System Operation

Flowchart Explanation:

1. Start the system
2. Initialize ESP8266, fingerprint sensor, OLED display, and Wi-Fi
3. Wait for fingerprint input
4. Scan fingerprint
5. Compare fingerprint with stored templates
6. If match found:
    - Send data to Google Sheets
    - Display "Attendance Recorded"
    - Beep buzzer once
7. If no match found:
    - Display "Access Denied"
    - Beep buzzer twice
8. Return to fingerprint scanning mode

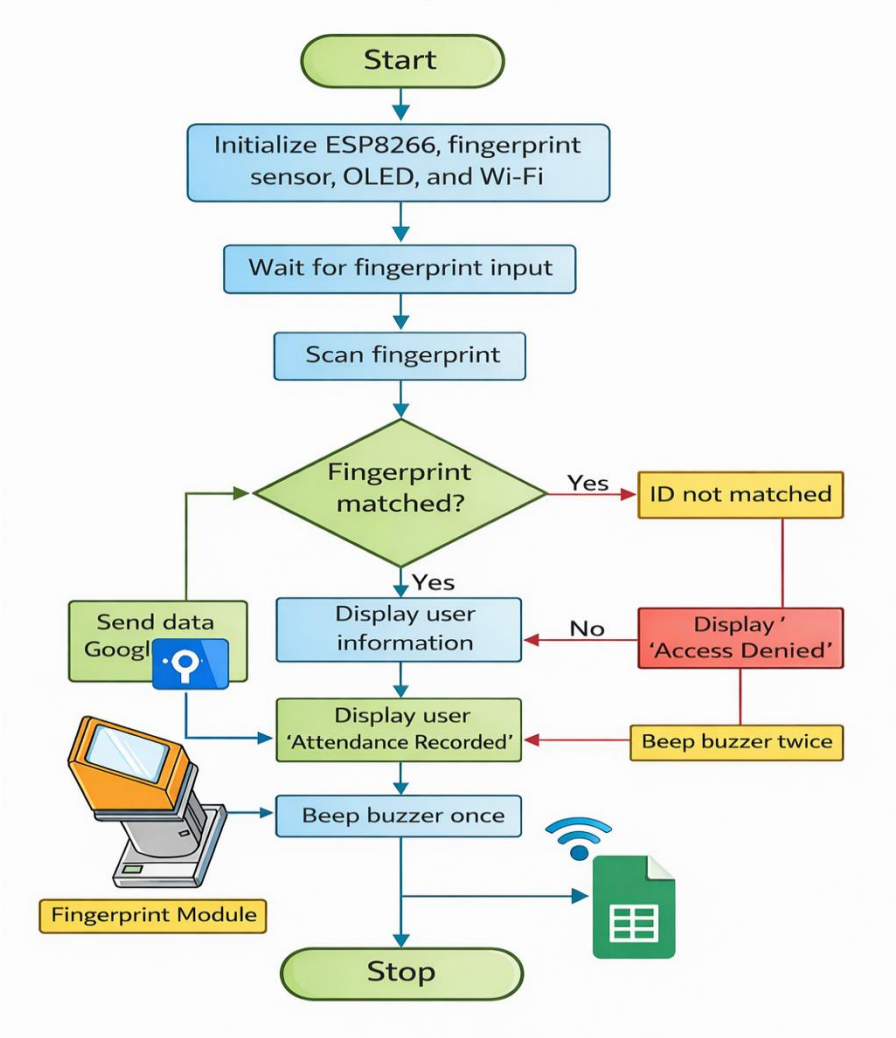



**Figure :** Flowchart of System Operation diagram

## 6. Project Objective

The main objectives of this project are:

- To design an automated attendance system using fingerprint authentication
- To eliminate proxy and manual attendance errors
- To store attendance data securely in the cloud
- To provide real-time feedback using OLED display and buzzer
- To implement an IoT-based solution using ESP8266 and Google Sheet

## 7. Project Overview

This project focuses on developing an **IoT-based fingerprint attendance system** using the **ESP8266 NodeMCU** microcontroller and the **R307 fingerprint sensor**.
Users place their finger on the fingerprint sensor for attendance. After successful verification, the system automatically sends the user ID along with date and time to **Google Sheets** via the internet using **Google Apps Script Web API**.

An **OLED display** shows system messages such as "Attendance Recorded" or "Access Denied", while a **buzzer** provides audio feedback.
This system is suitable for schools, colleges, offices, and organizations where reliable and automated attendance tracking is required.

## 8. Hardware Components

| Sl. No | Component Name | Description |
|---|---|---|
| **1** | ESP8266 NodeMCU | Main controller and Wi-Fi connectivity |
| **2** | R307 Fingerprint Sensor | Captures and verifies fingerprints |
| **3** | OLED Display (0.96") | Displays system messages |
| **4** | BC547 Transistor | Drives buzzer safely |
| **5** | Buzzer | Provides audio feedback |
| **6** | Resistors | Current limiting and protection |
| **7** | Veroboard | Compact soldered circuit mounting |
| **8** | Male & Female Headers | Easy module connections |
| **9** | Jumper Wires | Interconnections between modules |

## 9. Software Requirements

- Arduino IDE
- ESP8266 Board Package
- Adafruit Fingerprint Sensor Library
- Adafruit OLED Display Library
- Google Apps Script
- Google Sheets

## 10. System Architecture

The system consists of four major sections:

- Input Section: Fingerprint sensor captures biometric data
- Processing Section: ESP8266 verifies fingerprint and processes data
- Output Section: OLED display and buzzer provide feedback
- Cloud Section: Google Sheets stores attendance data

## 11. Working Principle

1. The system powers on and connects to Wi-Fi using ESP8266
2. User places finger on the R307 fingerprint sensor
3. The sensor scans and compares the fingerprint with stored templates
4. If fingerprint matches:
    - User ID, date, and time are sent to Google Sheets
    - OLED displays "Attendance Recorded"
    - Buzzer beeps once
5. If fingerprint does not match:
    - OLED displays "Access Denied"
    - Buzzer produces two short beeps
6. Attendance data is stored in real time for monitoring

**12. IoT Platform and Cloud Integration**

- **Google Sheets** is used as the cloud database
- **Google Apps Script Web API** acts as a bridge between ESP8266 and Google Sheets
- Data is transmitted over Wi-Fi using HTTP requests
- Attendance records can be accessed anytime from anywhere

**13. Expected Output**

- Successful fingerprint authentication
- Automatic attendance entry in Google Sheets
- Real-time date and time logging
- Visual confirmation on OLED display
- Audio feedback using buzzer

**14. Advantages**

- Eliminates proxy attendance
- Fully automated system
- Real-time cloud data storage
- Low-cost and scalable solution
- Easy data management and analysis

**15. Applications**

- Schools and colleges
- Offices and corporate organizations
- Factories and industries
- Examination halls
- Secure access control systems

**16. Future Improvements**

- Integration with mobile application
- Face recognition support
- SMS or email notification system
- Admin dashboard with analytics
- RFID + fingerprint hybrid system

**17. Conclusion**

The IoT-Based Fingerprint Attendance System provides a secure, efficient, and automated solution for attendance management. By integrating biometric authentication with cloud technology, the system ensures accurate data recording and eliminates manual errors. This project demonstrates the practical implementation of IoT in real-world applications and can be further enhanced for large-scale deployment.

**18. Algorithm**

Algorithm for IoT-Based Fingerprint Attendance System

1. Start the system
2. Initialize ESP8266 NodeMCU
3. Initialize OLED display
4. Initialize R307 fingerprint sensor
5. Connect ESP8266 to Wi-Fi network
6. Display “System Ready” on OLED
7. Wait for fingerprint input

8. Capture fingerprint image
9. Convert fingerprint image to template
10. Compare fingerprint with stored templates
11. If match found:
    - Fetch user ID
    - Read current date and time
    - Send attendance data to Google Sheets
    - Display “Attendance Recorded”
    - Activate buzzer once
12. Else:
    - Display “Access Denied”
    - Activate buzzer twice
13. Return to fingerprint scanning mode
14. Repeat continuously

**19. Source Code Explanation**

The source code is written using Arduino IDE and controls fingerprint verification, Wi-Fi communication, cloud data transfer, and output devices.

**Key Sections**

- Library Inclusion: Wi-Fi, fingerprint sensor, OLED, HTTP libraries
- Variable Declaration: Wi-Fi credentials, script URL, sensor and display objects
- Setup Function: Initializes hardware, connects Wi-Fi, displays status
- Loop Function: Handles fingerprint scanning, verification, data upload
- Cloud Upload: Sends User ID, Date, Time to Google Sheets
- Output Control: OLED messages and buzzer alerts

**20. Circuit Pin Table**

Pin Configuration of ESP8266 NodeMCU

| Sl. No | Component | Pin Name | ESP8266 Pin | Description |
|---|---|---|---|---|
| 1 | R307 Fingerprint Sensor | TX | D7 (GPIO13) | Data from sensor |
| 2 | R307 Fingerprint Sensor | RX | D8 (GPIO15) | Data to sensor |
| 3 | R307 Fingerprint Sensor | VCC | 5V / VIN | Power supply |
| 4 | R307 Fingerprint Sensor | GND | GND | Ground |
| 5 | OLED Display | SDA | D2 (GPIO4) | I2C data |
| 6 | OLED Display | SCL | D1 (GPIO5) | I2C clock |
| 7 | OLED Display | VCC | 3.3V | Power |
| 8 | OLED Display | GND | GND | Ground |
| 9 | Buzzer | +ve | D5 (GPIO14) | Control pin |
| 10 | BC547 | Base | Via resistor | Driver |
| 11 | BC547 | Collector | Buzzer –ve | Load |
| 12 | BC547 | Emitter | GND | Ground |