ATTENDRO: SMART BIOMETRIC + APP-BASED ATTENDANCE MANAGEMENT SYSTEM USING AI & IOT

A Project Report Submitted by

[STUDENT NAME]

In partial fulfillment of the requirements for the Diploma in

APPLIED AI & ML

At

Rajarambapu Institute of Technology, Islampur

2025–2026

Under the Guidance of

[GUIDE NAME]

CERTIFICATE

This is to certify that the project titled **"ATTENDRO: Smart Biometric + App-Based Attendance Management System using AI & IoT"** has been carried out by **[Student Name]** under my guidance and supervision in partial fulfillment of the requirements for the award of the Diploma in **Applied AI & ML** at **Rajarambapu Institute of Technology, Islampur**, during the academic year **2025–2026**.

____________________	____________________	____________________
Guide	**H.O.D.**	**Principal**

Date: _______________
Place: Islampur

ACKNOWLEDGEMENT

I express sincere gratitude to **[Guide Name]** for guidance and feedback, to the **Department of Applied AI & ML** for providing laboratory resources, to my group members for their dedicated collaboration, and to my parents for their unwavering support throughout this project.

TABLE OF CONTENTS

ABSTRACT

This project presents **Attendro**, an intelligent, portable, and offline-first biometric attendance system designed for educational institutes. Addressing the limitations of fixed biometric terminals and proxy-prone manual registers, Attendro introduces a **session-controlled** architecture where attendance can only be marked during active, authorized lecture sessions. The system integrates an **ESP32-based portable device** with a fingerprint sensor, a **Supabase (Cloud)** backend for real-time synchronization, and a **Faculty Web App** for session management.

Key innovations include **Time-Variant Batch Locking** (ensuring students only mark attendance for their specific batch), **Offline-First Synchronization** (queueing scans when Wi-Fi is unavailable), and **Context-Aware AI Rules** that validate scans against subject, class, and schedule constraints locally. This system satisfies AIML diploma requirements by leveraging biometric pattern recognition and rule-based decision intelligence to ensure data integrity and operational efficiency.

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 INTRODUCTION

1.1 PROBLEM STATEMENT

Polytechnic institutes and colleges today face significant challenges in attendance management: 1. **Proxy Attendance:** Manual rolls are easily manipulated, and QR-code systems can be shared remotely. 2. **Lack of Context:** Standard biometric machines allow "punching in" at any time, even if the student skips the actual lecture. 3. **Data Fragmentation:** Attendance data is often siloed in physical registers, delaying the generation of defaulter lists and compliance reports. 4. **Fixed Infrastructure:** Wall-mounted biometric devices are expensive to install in every classroom and laboratory.

1.2 OBJECTIVES

The primary objective is to design and build a **Portable, Session-Controlled Biometric Attendance System** that ensures attendance is taken only during an active lecture session for the correct subject, class, and batch. Specific sub-objectives include: - **Portable Hardware:** Develop an ESP32-based battery-powered device that moves with the faculty. - **Session Control:** Implement "Session Tokens" to lock attendance to specific time windows. - **Offline Resilience:** Ensure the system works without active Internet, syncing data when connectivity returns. - **AI Integration:** Incorporate biometric pattern recognition and rule-based AI for context verification. - **Real-Time Analytics:** Provide instant access to attendance stats via a web dashboard.

1.3 CONSTRAINTS AND ASSUMPTIONS

- The device relies on periodic Wi-Fi connectivity for synchronization.
- Fingerprint templates are stored securely and matched either on-device or on-server depending on mode.

- Faculty are responsible for charging the portable devices and initiating sessions via their smartphones.

CHAPTER 2 LITERATURE SURVEY

- **Biometric Systems:** Traditional fixed systems provide high accuracy but lack schedule awareness (Ross & Jain, 2021). Students can mark attendance and leave class.

- **IoT Attendance:** Recent IEEE papers discuss IoT-enabled attendance, but many lack offline queuing mechanisms, leading to data loss in unstable networks.

- **Session-Based Tokenization:** Secure systems in banking use time-limited tokens; we adapt this for attendance, generating a unique token for every lecture slot.

- **Gap Analysis:** Existing solutions are either purely software (easy to spoof) or purely hardware (dumb terminals). Attendro bridges this by making the hardware "context-aware" via a cloud connection.

CHAPTER 3 SCOPE OF THE PROJECT

3.1 OVERVIEW

The project scope covers the end-to-end development of hardware, firmware, cloud infrastructure, and a web application.

3.2 IN-SCOPE DELIVERABLES

1. **Portable Device Firmware (ESP32):**
    - Wi-Fi connectivity and Session Polling.
    - Fingerprint acquisition and Template generation.
    - Local Rule Engine (Context checks).
    - Offline Queue Management and Synchronization logic.
2. **Central Cloud (Supabase):**
    - SQL Schema for Users, Classes, and Attendance.
    - Edge Functions for `start_session`, `mark_attendance`, and `sync_offline`.
    - Row-Level Security (RLS) to protect flexible data.
3. **Faculty/Admin Web Interface:**
    - Dashboard for Timetable and Analytics.
    - Session Control (Start/End).
    - Reporting Module (Defaulters, PDF Export).

3.3 OUT-OF-SCOPE

- Advanced face recognition or iris scanning.

- Payroll processing integration.

- Hardware enclosure manufacturing (prototype uses acrylic/3D print case).

CHAPTER 4 METHODOLOGY / APPROACH

4.1 SYSTEM OVERVIEW

The system comprises three main interconnected components: 1. **Portable Device:** Handheld unit carried by faculty to class. It serves as the capture point. - *Hardware:* ESP32 Controller, R307 Fingerprint Sensor, 0.96" OLED Display, Rechargeable Battery. - *Identity:* Each device has a burned-in, unique `device_code`. 2. **Central Cloud (Supabase):** The brain of the system. - Stores all Master Data (Students, Subjects) and Transaction Data (Sessions, Logs). - Runs "Edge Functions" (Serverless API) to handle complex logic securely. 3. **Faculty App (Web):** The control center. - Allows faculty to log in, select their current class/subject, and "Start Session" on a specific device code.

4.2 DEVICE LOGIC & WORKFLOW

The device operates in a continuous state machine loop: 1. **Boot:** Connects to Wi-Fi and displays its `device_code`. 2. **Poll:** Periodically checks the Cloud for an "Active Session" linked to its ID. 3. **Session Active:** Once a session starts, it locks to that Class/Subject. The OLED displays "Marketing: [Subject]". 4. **Scan & Capture:** - Student places finger. - Device captures image -> extracts Template. - Device verifies **Context Rules** locally (Is this batch allowed? Duplicate scan?). 5. **Sync:** - **Online:** Pushes data immediately to Cloud Edge Function. - **Offline:** Saves scan to onboard memory (Queue) with timestamp. Pushes when Wi-Fi returns.

4.3 OFFLINE-FIRST & SYNC STRATEGY

To ensure robustness in campus environments with spotty Wi-Fi: - **Local Queue:** The ESP32 maintains a circular buffer of attendance records. - **Deferred Matching:** If the server cannot be reached, the scan is stored. Upon reconnection, the queue is flushed in batches. - **Integrity Check:** Records are only deleted from the device queue after receiving a positive `200 OK` acknowledgement from the server. This guarantees zero data loss.

4.4 AI & AIML JUSTIFICATION

This project meets AIML diploma requirements through: 1. **Biometric Pattern Recognition:** The core sensor performs feature extraction (minutiae points) and template matching, a fundamental pattern recognition task. 2. **Context-Aware Rule-Based Intelligence:** The device implements a "Classical AI" rule engine (If-Then logic) to make autonomous decisions: - *If* Current Time > End Time → Reject. - *If* Roll No is not in Batch A → Reject. - *If* Student already marked → Reject. 3. **Behavioral Analysis (Future/Server-side):** collected timestamp data can be analyzed to detect anomalies, such as "Rapid Fire Scans" indicating a potential bypass attempt.

CHAPTER 5 DESIGNS, WORKING AND PROCESSES

5.1 HARDWARE DESIGN

- **Controller:** ESP32 DevKit V1 (Dual-core, Wi-Fi/BLE).
- **Biometric Sensor:** R307 / Waveshare Optical Sensor (UART interface).
- **Display:** SSD1306 I2C OLED (Visual feedback for students).
- **Power:** Li-Ion Battery with TP4056 charging module.

5.2 DATABASE DESCRIPTION (SUPABASE)

The database is normalized to ensure consistency: - `devices`: Stores `device_code`, `status`, and `last_seen_at`. - `lecture_sessions`: The core entity connecting `faculty_id`, `device_code`, `subject_id`, `class_id`, `batch_id`, and `session_token`. - `students`: Contains `roll_no`, `fingerprint_template` (optional), and mapping to multiple classes. - `attendance_records`: Use `session_id` and `roll_no` as unique composite keys to prevent duplicates. - `finger_templates`: Central repository for "Enroll Once, Use Anywhere".

5.3 SESSION & CONTEXT RULES (THE VERIFICATION LOGIC)

A scan is only accepted if it passes the following **Context Rules**: 1. **Time Window:** `Current Time` must be between `session.start` and `session.end`. 2. **Device Match:** The provided `session_token` matches the device's current active session. 3. **Batch Lock:** If the session is for "Batch A" (e.g., Roll 1-20), a student with Roll 25 is rejected. 4. **De-Duplication:** Checks if `attendance_records` already contains this `roll_no` for this `session_id`.

5.4 USE CASE SCENARIOS

Scenario A: Normal Lecture 1. Faculty enters class, opens Web App. 2. Selects "Third Year", "Data Structures", "Theory". 3. Clicks "Start Session". 4. Device beeps and shows "Data Structures Active". 5. Device is passed around; students scan. 6. Faculty clicks "End Session". Report is generated instantly.

Scenario B: No Internet 1. Session starts (while online or cached). 2. Internet cuts out. 3. Students continue scanning; device shows "Saved Offline". 4. Faculty returns to staff room (Wi-Fi restored). 5. Device automatically flushes queue; server processes records.

CHAPTER 6 RESULTS AND APPLICATIONS

6.1 INNOVATION & NOVELTY

1. **Session-Controlled Attendance:** Unlike standard biometrics that accept any finger at any time, Attendro only unlocks for specific, authorized windows.
2. **Portable & Wire-Free:** The battery-powered design allows it to be used in labs, seminar halls, or playgrounds without wiring.
3. **Smart Batch Locking:** Enforces attendance rules strictly (e.g., stopping Batch B students from attending Batch A labs).
4. **Hybrid AI Approach:** Combines edge-based rule intelligence with cloud-based analytics.

6.2 APPLICATION AREAS

- **Polytechnic & Engineering Colleges:** For theory lectures, practical labs, and workshops.
- **Seminar Halls:** For tracking attendance at guest lectures.
- **Examination Halls:** For verifying student identity before exams.

6.3 EXPECTED RESULTS

- **Accuracy:** >98% True Acceptance Rate on fingerprints.
- **Speed:** <2 seconds per student scan-to-verify.
- **Efficiency:** Reduces attendance taking time from 10 minutes (manual) to ~2 minutes (parallel scanning).

CHAPTER 7 CONCLUSION

Attendro successfully demonstrates how IoT and AI can modernize a legacy process. By decoupling the biometric sensor from the wall and connecting it to a session-aware cloud brain, we eliminate proxy attendance and automate the tedious task of compliance reporting. The system is offline-tolerant, secure, and user-friendly, making it a viable product for educational institutions.

REFERENCES

[1] Ross, A., & Jain, A. (2021). "Biometric sensor interoperability". *IEEE Trans. on PAMI*.

[2] Espressif Systems. (2023). "ESP32 Technical Reference Manual".

[3] Supabase. (2024). "Realtime & Edge Functions Documentation".

[4] "R307 Optical Fingerprint Sensor Datasheet", Hangzhou Grow Technology.

Diagrams referenced in List of Figures are available in the project documentation folder.