PROJECT REPORT

Submitted by

KAUSHIK UMESH CHANDRA (RA2211003011530)
SEJAL SOMANI (RA2211003011527)
APARIJIT CHAKRABORTY (RA2211003011540)

Topic:

"AUTOMATIC ATTENDANCE RECORDER USING RFID CARD"

Under the Guidance of

DR. M. KARTHIKEYAN

Assistant Professor, Computing Technologies Department

In partial satisfaction of the requirements for the degree of

BACHELOR OF TECHNOLOGY
In
COMPUTER SCIENCE ENGINEERING
with specialization in: CORE

SRM INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be **University** *u/s 3 of UGC Act, 1956)*

SCHOOL OF COMPUTING
COLLEGE OF ENGINEERING AND TECHNOLOGY
SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
(May 2023)

SRM INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR – 603203

BONAFIDE CERTIFICATE

Certified that this project report title **"Automatic Attendance Recorder using RFID Card"** is the Bonafide work done by **Kaushik Umesh Chandra (RA2211003011530); Sejal Somani (RA2211003011527); Aparijit Chakraborty (RA2211003011540)** Who completed the project under my Supervision. Certified further, Get to the best of my knowledge the work reported herein does not form part of any other work.

SIGNATURE
DR. M. KARTHIKEYAN
COA – Course Faculty
Assistant Professor
Department Of Computing Technology
SRMIST

SIGNATURE
DR. M. Pushpalatha
Professor & Head
Department Of Computing Technology
SRMIST

Index:

Team Members:

- ✓ Kaushik Umesh Chandra (RA2211003011530)
- ✓ Sejal Somani (RA2211003011527)
- ✓ Aparijit Chakraborty (RA2211003011540)

Automatic Attendance Recorder using RFID Card

❖ Introduction:

In an era characterized by technological advancements and the relentless pursuit of efficiency, traditional methods of taking attendance in educational institutions and workplaces have become increasingly cumbersome and error-prone. Recognizing the need for a more streamlined and accurate solution, the Automatic Attendance Recorder using RFID (Radio-Frequency Identification) card technology has emerged as a game-changer.

This innovative system harnesses the power of RFID technology to transform attendance management into a hassle-free, automated process. By providing individuals with RFID cards that emit unique identification signals, the system is able to record attendance data swiftly and accurately, eliminating the need for manual record-keeping and reducing the risk of errors.

In this introduction, we will delve deeper into the concept of the Automatic Attendance Recorder using RFID card technology, exploring its key features, benefits, and its potential to revolutionize attendance tracking in a variety of environments, from educational institutions to corporate offices. This system is poised to enhance productivity, accountability, and data security while simplifying the lives of both administrators and attendees.

❖ Abstract:

Efficient attendance management is a fundamental aspect of ensuring productivity and accountability in educational institutions. Traditional paper-based attendance tracking methods are prone to errors, time-consuming, and often lack real-time insights. In this digital age, the Automatic Attendance Recorder using RFID card technology has emerged as a transformative solution, offering a reliable and automated approach to recording attendance for students and faculty members alike.

At the heart of this system is RFID technology, which provides a seamless and contactless method for tracking attendance. Each student or faculty member is assigned a unique RFID card embedded with a tiny chip that emits radio waves. When the card is brought into proximity with an RFID reader, the reader captures the unique signal emitted by the card, instantly recording the individual's attendance. The technology operates in the ultra-high frequency (UHF) range, allowing for rapid data collection without the need for direct physical contact.

Key Features and Functionality

The Automatic Attendance Recorder using RFID card technology offers a range of features that make it a compelling solution for modern educational institutions:

1. **Real-time Data Capture**: Attendance records are updated in real-time, enabling both students and instructors to monitor attendance status instantly. This real-time access to attendance data provides immediate insights into class participation and helps in early intervention for at-risk students.

2. **Accuracy and Elimination of Proxy Attendance**: Since RFID cards are unique to individuals and require physical presence to record attendance, the system significantly reduces the possibility of proxy attendance, ensuring the integrity of attendance records.

3. **Ease of Integration**: The system can seamlessly integrate with existing school management software and databases, making it a straightforward addition to the existing infrastructure.

4. **Data Security**: RFID technology provides an additional layer of data security. Attendance records are stored securely in a centralized database, protected from unauthorized access and tampering.

5. **Analytics and Reporting**: The system allows for the generation of attendance reports and analytics, facilitating data-driven decision-making for educational institutions. This can be particularly valuable for identifying trends and improving educational outcomes.

❖ Objective:

The primary goal of the Automated Attendance Recorder System is to create a seamless and efficient method for tracking attendance in an educational setting. This innovative system is designed to:

Student Identification: Utilize RFID card technology to accurately read and identify each student by their unique card data. This ensures a quick and error-free method for recognizing students as they enter the class or designated area.

Date and Time Logging: Record the precise date and time at which students scan their RFID cards. This timestamp serves as a crucial component in monitoring attendance, helping to track punctuality and participation.

Punctuality Status: Determine whether the student's arrival is on time or late, providing instant feedback to both the student and the instructor. This real-time assessment encourages punctuality and allows educators to take appropriate action if necessary.

Data Integration with Google Sheets: Seamlessly upload the attendance data through a secure Wi-Fi connection to a dedicated Google Sheets database, which is stored in the cloud. This integration ensures that attendance records are stored securely and are easily accessible to authorized personnel from anywhere with an internet connection.

Web Page Creation: Utilize Google Sheets to generate a user-friendly web page that provides a clear and organized overview of attendance records. This web page can be accessed by administrators, instructors, and students, promoting transparency and accountability within the educational institution.

By incorporating these features into the Automated Attendance Recorder System, we are ushering in a new era of attendance management. This system not only simplifies the process but also enhances data accuracy, promotes punctuality, and facilitates data accessibility through cloud-based storage and web-based interfaces. It represents a significant advancement in educational technology, offering a comprehensive solution for attendance tracking and management.

❖ **Code:**

```
 #include <WiFi.h>
#include <HTTPClient.h>
#include <vector>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

/**********************************************************************
***********/
// Things to change
const char * ssid = "Galaxy A34 5G 4F72";
const char * password = "rfid1234";
String GOOGLE_SCRIPT_ID = "AKfycbwAoHiaoAtAOa4pD-
L1fL1t5taHA5WkN9CxdahIOoEy91ZL9YzO3dWw4L8DkbRjLdL1";

const int sendInterval = 500;
/**********************************************************************
***********/
int lcdColumns = 16;
int lcdRows = 2;
LiquidCrystal_I2C lcd(0x27, lcdColumns, lcdRows);

WiFiClientSecure client;

#include "SPI.h"      // SPI library
#include "MFRC522.h"  // RFID library (https://github.com/miguelbalboa/rfid)

const int buzzer = 2;
const int pinRST = 0;
const int pinSDA = 5;

MFRC522 mfrc522(pinSDA, pinRST);  // Set up mfrc522 on the Arduino
```

```
#include <RTClib.h>
#include <Wire.h>
RTC_DS3231 rtc;
char d[32];
char t[32];
String Attendance;

// Define check in time
const int checkInHour = 14;
const int checkInMinute = 20;
//Variable to hold user check in
int userCheckInHour;
int userCheckInMinute;

//String Date, Time, Attendance;

// Define the I2C pins for each device

void sendData(String params) {
  HTTPClient http;
  String url="https://script.google.com/macros/s/AKfycbwAoHiaoAtAOa4pD-
L1fL1t5taHA5WkN9CxdahIOoEy91ZL9YzO3dWw4L8DkbRjLdL1/exec?action=writ
e&"+params;
  //Serial.print(url);
  //Serial.print("\nMaking a request");
  http.begin(url);//, root_ca); //Specify the URL and certificate
  int httpCode = http.GET();
  http.end();
  //Serial.println(": done "+httpCode);
}

void removeBracketsAndSpaces(String &input) {
 input.replace("[", "");
 input.replace("]", "");
 input.trim();
}
```

```
void spreadsheet_comm(std::vector<String> &numbers, std::vector<String> &names)
{
  HTTPClient http;
  String url = "https://script.google.com/macros/s/AKfycbwAoHiaoAtAOa4pD-
L1fL1t5taHA5WkN9CxdahIOoEy91ZL9YzO3dWw4L8DkbRjLdL1/exec?action=rea
d&Column=A";
  Serial.println("\n\nScanning...");
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Scanning...");

  http.begin(url.c_str()); // Specify the URL and certificate
  http.setFollowRedirects(HTTPC_STRICT_FOLLOW_REDIRECTS);
  int httpCode = http.GET();
  String payload = http.getString();
  //Serial.println(payload);

  //Serial.println("function has been reached");

  int tokenCount = 0;
  while (payload.length() > 0) {
    //Serial.println("while loop has been cleared");
    int openBracketIndex = payload.indexOf('[');
    int closeBracketIndex = payload.indexOf(']');
    if (openBracketIndex != -1 && closeBracketIndex != -1) {
      //Serial.println("first if loop has been cleared");
      String token = payload.substring(openBracketIndex + 1, closeBracketIndex);
      int commaIndex = token.indexOf(',');
      if (commaIndex != -1) {
        // Remove square brackets and spaces from numbers
        String numberStr = token.substring(0, commaIndex);
        removeBracketsAndSpaces(numberStr);

        //Serial.print("Number: ");
        //Serial.println(numberStr);
        numbers.push_back(numberStr);
```

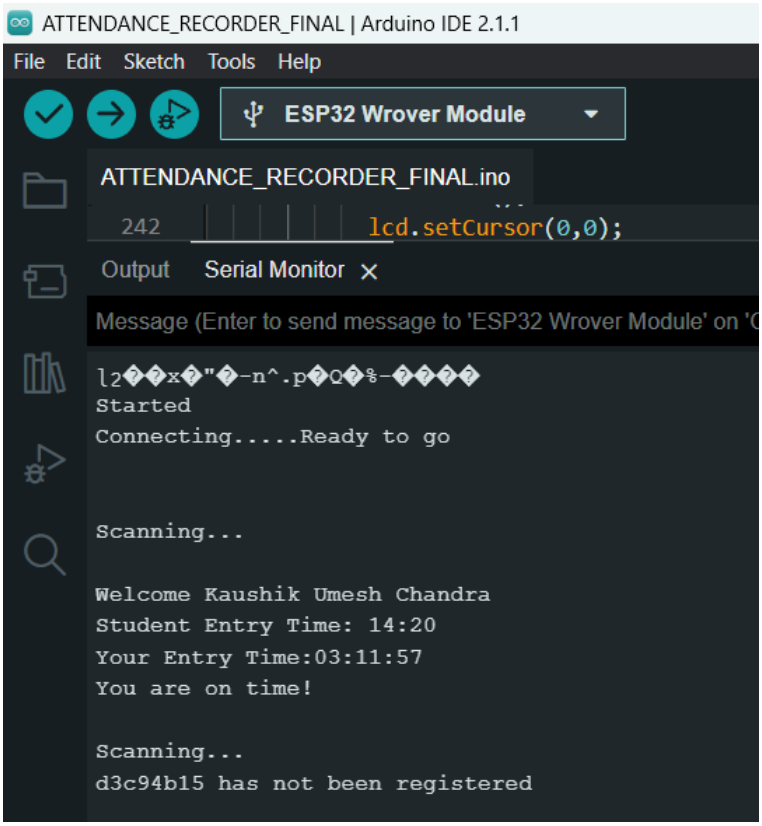
```
        //Serial.print("Name: ");
        //Serial.println(token.substring(commaIndex + 1));
        names.push_back(token.substring(commaIndex + 1));
      }
      payload = payload.substring(closeBracketIndex + 1);
    } else {
      break;
    }
  }
}

String removeDoubleQuotes(String input) {
  // Check if the input string starts with a double quote
  if (input.startsWith("\"")) {
    input = input.substring(1);
  }

  // Check if the input string ends with a double quote
  if (input.endsWith("\"")) {
    input = input.substring(0, input.length() - 1);
  }

  return input;
}

void setup() {
  Wire.begin();
  SPI.begin();        // open SPI connection
  mfrc522.PCD_Init();  // Initialize Proximity Coupling Device (PCD)
  Serial.begin(9600);  // open serial connection

  rtc.begin();

  pinMode(buzzer,OUTPUT);
  delay(10);
  lcd.init();
  lcd.backlight();
```

```
  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password);

  Serial.println("\nStarted");
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Started");
  Serial.print("Connecting");
  lcd.setCursor(0, 1);
  lcd.print("Connecting...");
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("Ready to go");
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Ready to go");
}

void loop() {
  DateTime now = rtc.now();
  if (mfrc522.PICC_IsNewCardPresent()) {
    if (mfrc522.PICC_ReadCardSerial()) {
      digitalWrite(buzzer, HIGH);
      delay(250);
      digitalWrite(buzzer, LOW);
      std::vector<String> numbers;
      std::vector<String> names;
      spreadsheet_comm(numbers, names);
      String UID = "";

      for (byte i = 0; i < mfrc522.uid.size; ++i) {
        UID += String(mfrc522.uid.uidByte[i], HEX);
      }
```

```
    bool found = false;
    int sheet_length = numbers.size();
    for (int i = 0; i < sheet_length; i++) {
      String stored_UID = removeDoubleQuotes(numbers[i]);
      //Serial.println(stored_UID);

      if (stored_UID == UID) {
        Serial.print("\nWelcome ");

        lcd.setCursor(0, 0);
        lcd.print("Welcome");

        Serial.println(removeDoubleQuotes(names[i]));
        lcd.setCursor(0, 1);
        lcd.print(removeDoubleQuotes(names[i]));
        found = true;
        userCheckInHour = now.hour();
        userCheckInMinute = now.minute();
        sprintf(d, "%02d/%02d/%02d", now.day(), now.month(), now.year());
        sprintf(t, "%02d:%02d:%02d", now.hour(), now.minute(), now.second());
        String Date = String(d);
        String Time = String(t);
        Serial.print("Student Entry Time: 14:20");
        Serial.print("\nYour Entry Time:");
        Serial.print(Time);
        if ((userCheckInHour < checkInHour) || ((userCheckInHour == checkInHour)
&& (userCheckInMinute <= checkInMinute))) {
          Attendance = "PRESENT";
          Serial.print("\nYou are on time!");
          lcd.clear();
          lcd.setCursor(0,0);
          lcd.print(removeDoubleQuotes(names[i]));
          lcd.setCursor(0,1);
          lcd.print("You are on time!");
        } else {
          Attendance = "LATE";
          Serial.print("\nYou are late...");
```

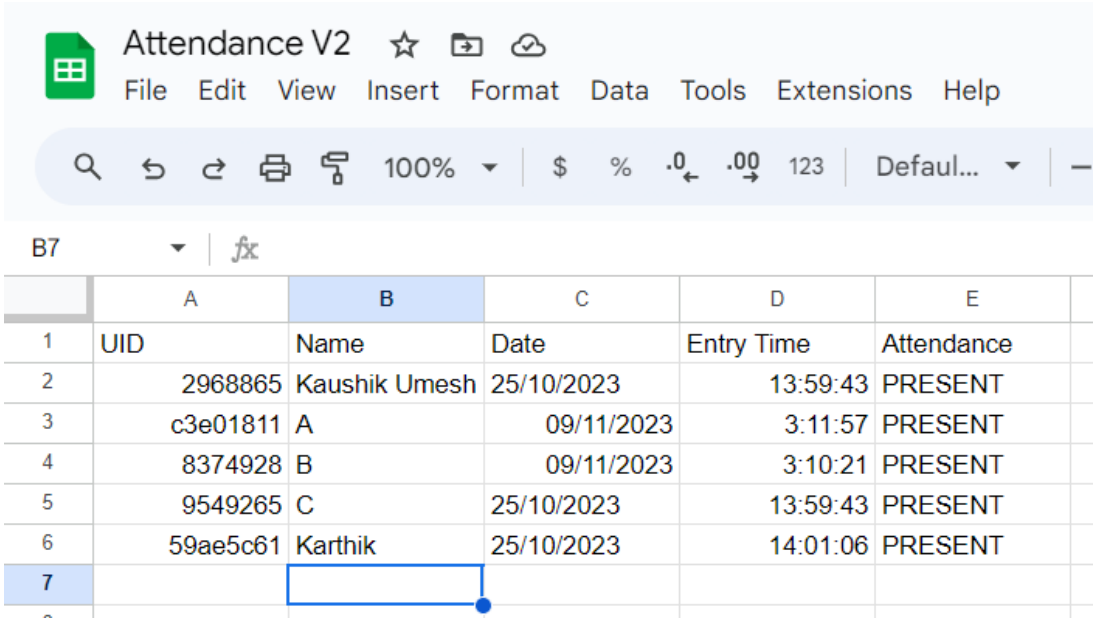
```
          lcd.clear();
          lcd.setCursor(0,0);
          lcd.print(removeDoubleQuotes(names[i]));
          lcd.setCursor(0,1);
          lcd.print("You are late...");
        }
        sendData("Row=" + String(i+2) + "&Column=E&Value=" + Attendance);
        //delay(300);
        sendData("Row=" + String(i+2) + "&Column=D&Value=" + Time);
        //delay(300);
        sendData("Row=" + String(i+2) + "&Column=C&Value=" + Date);
        break;
      }

    }
    if (!found){
      Serial.print(UID);
      Serial.print(" has not been registered");
      lcd.clear();
      lcd.setCursor(0,0);
      lcd.print(UID);
      lcd.setCursor(0,1);
      lcd.print("has not been registered");
    }
   }
  }
}
```

❖ **Output:**



Serial Monitor Output:

a. **WiFi Connection has been established**
b. **Student Kaushik Umesh Chandra's RFID Card has been scanned**
c. **Kaushik Umesh Chandra has been marked present due to entering before student entry time**
d. **User d3c94b15 who isn't part of the class has not been recognized**



	A	B	C	D	E
1	UID	Name	Date	Entry Time	Attendance
2	2968865	Kaushik Umesh	25/10/2023	13:59:43	PRESENT
3	c3e01811	A	09/11/2023	3:11:57	PRESENT
4	8374928	B	09/11/2023	3:10:21	PRESENT
5	9549265	C	25/10/2023	13:59:43	PRESENT
6	59ae5c61	Karthik	25/10/2023	14:01:06	PRESENT
7					

Google Sheets Database containing all the students in the class with their assigned RFID UID values.

Columns Date, Entry Time, and Attendance are automatically filled by attendance recorder upon RFID scan if user is recognized.

❖ Explanation & Conclusion:

This Automatic Attendance Recorder using RFID card uses the following components:

INPUT:
RFID READER MODULE
RTC DS3231 MODULE

MICROCONTROLLER:
ESP32

OUTPUT:
BUZZER + LED MODULE

This code is an Arduino sketch designed for an attendance recording system using RFID cards. The code integrates various libraries and hardware components to accomplish the task. Here's an overview of how the code works:

1. **Library and Hardware Setup**: The code begins by including necessary libraries for RFID, Wi-Fi connectivity, and a real-time clock module (RTC). It also initializes variables, pins, and sets up the LCD display.
2. **Wi-Fi Connection**: The code establishes a Wi-Fi connection to the specified network (SSID and password). It waits until a successful connection is made before proceeding.
3. **Main Loop**: The core of the program resides in the **loop** function, which runs continuously.
4. **RFID Card Detection**: It checks if a new RFID card is present near the RFID reader using the **mfrc522.PICC_IsNewCardPresent()** and **mfrc522.PICC_ReadCardSerial()** functions. If a card is detected, a buzzer sound is triggered, and the program proceeds to read the card's unique UID (Unique Identifier).
5. **Data Retrieval**: The code communicates with a Google Sheets spreadsheet using a web request. It retrieves data from the spreadsheet, specifically student information and their respective UIDs, and stores them in vectors.
6. **UID Comparison**: It then compares the UID of the detected RFID card with the UIDs stored in the spreadsheet. If a match is found, the system records the student's entry time, date, and attendance status (present or late). The student's name and status are displayed on the LCD.
7. **Data Upload**: The system uploads the attendance data, including time, date, and attendance status, back to the Google Sheets spreadsheet using a web request.
8. **Feedback**: The LCD provides feedback to the user, indicating whether they are on time or late for attendance.
9. **Repeat**: The code continues to loop, waiting for the next RFID card to be scanned, and the process is repeated.

This code essentially allows for the automatic recording of student attendance based on their RFID card scans. It interfaces with Google Sheets to keep track of attendance data and provides real-time feedback to students. The system also has a time-based check to determine whether a student is on time or late for their attendance, which is crucial for monitoring punctuality.

❖ GitHub Link:

https://github.com/Kaush7040/Automatic-Attendance-Recorder-through-RFID-Card-using-ESP32