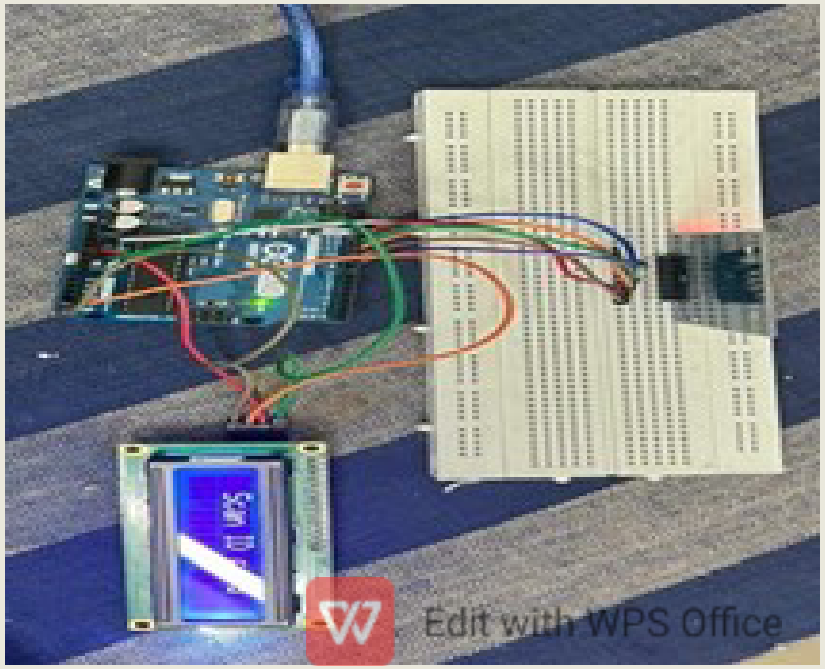
SMART PORTABLE ATTENDANCE SYSTEM USING RFID READER

PG1

- ADITYA PAL(2022CEB1001)
- ALAPATI RAMYASRI(2022CEB1003)
- APUM DUPAK(2022CEB1004)
- AYUSHI RAUT(2022CEB1005)
- DEV BHARGAVA(2022CEB1007)
- DHEERAJ THALOUR(2022CEB1008)

OBJECTIVE:

- Automatic attendance recording system that allow us to simply fill the attendance just by swiping or moving our ID card on RFID reader.



INTRODUCTION

- In this project, we have designed RFID Based Attendance system using Arduino. RC522 is a very simple yet effective module. It is an RFID module and is used for scanning RFID cards.
- It’s a new technology and is expanding day by day. Nowadays it is extensively used in offices where employees are issued an RFID card and their attendance is marked when they touch their card to the RFID reader.

COMPONENTS REQUIRED

- Arduino Uno
- RFID reader RC522 Module
- UID cards
- I2C module
- LCD display
- Breadboard
- Jumper wires

- **RFID Reader:** The core component of the system, the RFID reader, communicates with RFID tags to capture unique identifiers and transmit this information to the central database.
- **RFID tags:** Each individual has a unique RFID card or tag containing a unique identifier that is read by the RFID card reader RFID card reader. These cards are assigned to individuals to track their attendance.
- **Database:** A database stores information related to the RFID cards and the individuals they represent. It maintains records of attendance history, linking unique IDs from the cards with the individual they belong to.
- **Arduino uno:** Acts as the brain if the system, responsible for reading RFID data and managing the attendance system.

WORKING

- In this project, we have designed an RFID based attendance system using Arduino. First, we store a set of RFID card data in our system.
- When the person with correct RFID card comes and swipes his RFID card, it emits radio waves. The RFID reader captures these waves and detects the unique ID of the card/tag.
- The Arduino processes this data and reads the unique ID stored on the RFID card/tag and compares the read ID with a pre-stored list of valid IDs or an existing database of authorized users.
- If the ID matches an authorized user, the system marks their attendance as present and show attendance status through an LCD screen

CODING PART

```
#include <SPI.h>
#include <MFRC522.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

#define SS_PIN 10
#define RST_PIN 9

MFRC522 mfrc522(SS_PIN, RST_PIN);   // Create
MFRC522 instance

// Valid UIDs of students
byte validUIDs[][4] = {
  {0x8A, 0x5E, 0xB8, 0x89},  // UID of student 1
  {0x53, 0x15, 0x57, 0xC8},  // UID of student 2
  {0x12, 0x34, 0x56, 0x78},  // UID of student 3
  {0xAB, 0xCD, 0xEF, 0x12},  // UID of student 4
};

// Student names
const char* studentNames[] = {
  "SAGAR BISWAS",  // Name of student 1
  "ANKIT BISWAS",    // Name of student 2
  "ELON MUSK",       // Name of student 3
  "ARIJIT SINGH",     // Name of student 4
};


// LCD I2C display
LiquidCrystal_I2C lcd(0x27, 16, 2);  // Adjust the
address and size according to your display
```



```
void setup() {
  Serial.begin(9600);   // Initialize serial
communication
  SPI.begin();          // Initiate SPI bus
  mfrc522.PCD_Init();   // Initiate MFRC522
  lcd.begin(16, 2);     // Initialize the LCD display
  lcd.init();
  lcd.backlight();      // Turn on the backlight
  lcd.clear();
  lcd.setCursor(2, 0);
  lcd.print("Scan ID card");
  Serial.println("Ready to read RFID cards");
}

void loop() {
  // Look for new cards
  if (mfrc522.PICC_IsNewCardPresent()) {
    if (mfrc522.PICC_ReadCardSerial()) {
      // Show UID on serial monitor
      Serial.print("UID tag: ");
      for (byte i = 0; i < mfrc522.uid.size; i++) {
        Serial.print("0x");
        if (mfrc522.uid.uidByte[i] < 0x10)
Serial.print("0");
        Serial.print(mfrc522.uid.uidByte[i], HEX);
        if (i < mfrc522.uid.size - 1) Serial.print(", ");
      }
      Serial.println();
      Serial.print("UID Number: ");
      String content = "";
      byte letter;
      for (byte i = 0; i < mfrc522.uid.size; i++) {
        content.concat(String(mfrc522.uid.uidByte[i] <
0x10 ? "0" : ""));
        content.concat(String(mfrc522.uid.uidByte[i],
HEX));
      }
      content.toUpperCase();
      Serial.println(content);
```

```
    // Check if the UID matches any of the valid
UIDs
    bool uidMatched = false;
    int studentIndex = -1;
    for (int i = 0; i < sizeof(validUIDs) /
sizeof(validUIDs[0]); i++) {
      if (memcmp(mfrc522.uid.uidByte, validUIDs[i],
mfrc522.uid.size) == 0) {
        uidMatched = true;
        studentIndex = i;
        break;
      }
    }

    // Perform actions based on UID match
    if (uidMatched) {
      // UID matches, perform attendance action for
the corresponding student

      // Update the attendance status in the
database
      // Example: Update the attendance status for
student studentIndex+1 to "Present"
      updateAttendance(studentIndex + 1,
"Present");

      delay(3000);  // Display attendance status for 3
seconds
    } else {
      // UID doesn't match, perform other actions
(e.g., display error message)

      lcd.clear();
      lcd.setCursor(0, 0);
      lcd.print("ID doesn't match");
      delay(2000);
    }
```



```
    // Clear the LCD display and reset for the next
scan
    lcd.clear();
    lcd.setCursor(2, 0);
    lcd.print("Scan ID card");

    delay(1000);   // Delay to avoid reading the card
multiple times in a short period
  }
  mfrc522.PICC_HaltA();   // Stop reading
  mfrc522.PCD_StopCrypto1();  // Stop encryption
on PCD
 }
}

// Function to update attendance in the database
void updateAttendance(int studentID, const char*
status) {
 // Implement your code to update the attendance
status in the database
 // based on the studentID and status parameters

 // Display the name of the student on the LCD
 lcd.clear();
 lcd.setCursor(0, 0);
 lcd.print(studentNames[studentID - 1]);  // -1 to
adjust for array indexing
 lcd.setCursor(0, 1);  // Set cursor to the second
line
 lcd.print("~");
 lcd.setCursor(2, 1);  // Set cursor position after 4
pixels
 lcd.print(status);
}
```

APPLICATIONS

- **Education:** Schools and colleges can use this system to automate attendance tracking for students and faculty. It reduces manual data entry and provides accurate attendance records.
- **Time and Attendance Management:** The system is valuable for tracking the attendance of workers in manufacturing facilities or offices, making it easier to manage working hours.
- **Events:** Event organizers can use RFID-based attendance systems for conferences, trade shows, and concerts to monitor participant engagement and session attendance.
- **Security:** The system can enhance access control and security by monitoring who enters and exits specific areas or buildings.
- **Healthcare:** Hospitals and healthcare facilities can use RFID attendance systems to track the presence of medical staff and ensure timely response to emergencies.