#include <SPI.h>

#include <MFRC522.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] = {

  {0xa3, 0x53, 0x30, 0x11},  // UID of Abhay

  {0x23, 0x82, 0xb6, 0x92},  // UID of Jaish

  {0x03, 0x9a, 0xf7, 0x12}   // UID of Jaish

};

// Student names

const char\* studentNames[] = {

  "Abhay",     // Name of Abhay

  "Jaish",     // Name of Jaish

  "Jaish"  // Name of Jaish

};

void setup() {

  Serial.begin(9600); // Initialize serial communication

  SPI.begin();        // Initiate SPI bus

  mfrc522.PCD\_Init(); // Initiate MFRC522

  Serial.println("Ready to read RFID cards");

}

void loop() {

  // Look for new cards

  if (mfrc522.PICC\_IsNewCardPresent() && 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, "True");

      delay(3000); // Display attendance status for 3 seconds

    } else {

      // UID doesn't match, perform other actions (e.g., display error message)

      Serial.println("ID doesn't match");

      delay(2000);

    }

    delay(1000); // Delay to avoid reading the card multiple times in a short period

  }

}

// 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 serial monitor

  Serial.print("Student Name: ");

  Serial.println(studentNames[studentID - 1]); // -1 to adjust for array indexing

  Serial.print("Book Issued: ");

  Serial.println(status);

}









