#include <SoftwareSerial.h>

#include <SPI.h>

#include <MFRC522.h>

#define SS\_PIN 10

#define RST\_PIN 9

MFRC522 mfrc522(SS\_PIN, RST\_PIN); // Create MFRC522 instance.

SoftwareSerial espSerial = SoftwareSerial(8,7);

String ssid="iot1";

String password ="iot12345";

String line="";

char c[100];

boolean DEBUG=true;

void setup()

{

Serial.begin(9600); // Initiate a serial communication

espSerial.begin(9600);

espSerial.println("AT+CWMODE=1");

showResponse(1000);

espSerial.println("AT+CWJAP=\""+ssid+"\",\""+password+"\"");

showResponse(5000);

espSerial.println("AT+RST"); // Enable this line to reset the module;

showResponse(1000);

if (DEBUG) Serial.println("Setup completed");

SPI.begin(); // Initiate SPI bus

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

Serial.println("Approximate your card to the reader...");

Serial.println();

}

void mydemoWriteRead(String value1)

{

String cmd = "AT+CIPSTART=\"TCP\",\""; // TCP connection

cmd += "my-demo.in"; // api.thingspeak.com

cmd += "\",80";

espSerial.println(cmd);

if (DEBUG) Serial.println(cmd);

if(espSerial.find("Error")){

if (DEBUG) Serial.println("AT+CIPSTART error");

return false;

}

//http://my-demo.in/parentchild/request1.aspx?id=1D%20D4%2041%2073

String getStr = "GET /parentchild/request1.aspx?";

getStr +="id=";

getStr += String(value1);

getStr += " HTTP/1.1\r\n";

getStr += "Host: my-demo.in\r\n";

getStr += "User-Agent: Arduino\r\n";

getStr += "Connection: close\r\n\r\n";

cmd = "AT+CIPSEND=";

cmd += String(getStr.length());

espSerial.println(cmd);

if (DEBUG) Serial.println(cmd);

if(espSerial.find(">")){

espSerial.println(getStr);

Serial.println(getStr);

while(espSerial.available()!=0){

if(espSerial.find("success:")){

line = espSerial.readString();

}

}

Serial.println(c[0]);

}

if(c[0]=='1'){

digitalWrite(13, HIGH); // turn the LED on (HIGH is the voltage level)

}

else{

digitalWrite(13, LOW); // turn the LED on (HIGH is the voltage level)

}

espSerial.flush();

}

void loop()

{

// Look for new cards

if ( ! mfrc522.PICC\_IsNewCardPresent())

{

return;

}

// Select one of the cards

if ( ! mfrc522.PICC\_ReadCardSerial())

{

return;

}

//Show UID on serial monitor

Serial.print("UID tag :");

String content= "";

byte letter;

for (byte i = 0; i < mfrc522.uid.size; i++)

{

Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ");

Serial.print(mfrc522.uid.uidByte[i], HEX);

content.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " "));

content.concat(String(mfrc522.uid.uidByte[i], HEX));

}

Serial.println();

Serial.print("Message : ");

content.toUpperCase();

if (content.substring(1) == "1D D4 41 73") //change here the UID of the card/cards that you want to give access

{

Serial.println("Authorized access");

Serial.println();

mydemoWriteRead("1D%20D4%2041%2073");

delay(3000);

}

else if (content.substring(1) == "77 73 7D B2") //change here the UID of the card/cards that you want to give access

{

Serial.println("Authorized access");

Serial.println();

mydemoWriteRead("77%2073%207D%20B273");

delay(3000);

}

else {

Serial.println(" Access denied");

delay(3000);

}

}

void showResponse(int waitTime){

long t=millis();

char c;

while (t+waitTime>millis()){

if (espSerial.available()){

c=espSerial.read();

if (DEBUG) Serial.print(c);

}

}

}