

```

// #define BLYNK_PRINT Serial
#include <FS.h>
#include <ESP8266WiFi.h>
#include <MFRC522.h>
#include <BlynkSimpleEsp8266.h>

// Pinos
#define RST_PIN D3
#define SS_PIN D8
#define PIRpin D1
#define LEDpin D0
#define BUZZERpin 1
#define RELEpin D2

// Autentificação
char auth[] = "sua_identificação";
char ssid[] = "seu_ssid";
char pass[] = "sua_senha_wifi";

// Variáveis
boolean alarme;
boolean ativo;
boolean disparo;
boolean estadoAnterior;
boolean estadoLED = false;
boolean estadoRele = true;
unsigned long timerLeituraRFID = 0;

WidgetLCD lcd(V1);
BlynkTimer timer;
MFRC522 mfrc522(SS_PIN, RST_PIN);
File arquivo;

int timerPIR = 1;
int timerRFID = 2;
int timerLED = 3;
int timerBUZZER = 4;
int timerATIVAR = 5;

BLYNK_CONNECTED() {
  Blynk.syncVirtual(V0);
  Blynk.syncVirtual(V2);
}

BLYNK_WRITE(V0) {
  int alarmeInt = param.asInt();
  alarme = (bool)alarmeInt;
  alarme = !alarme;
  arquivo = SPIFFS.open("/alarme", "w");
  arquivo.write(alarme);
  arquivo.close();
  if (alarme != estadoAnterior) {
    mudarAlarme();
    lcd.print(0, 1, " via aplicativo ");
  }
  estadoAnterior = alarme;
  arquivo = SPIFFS.open("/anterior", "w");
  arquivo.write(estadoAnterior);
  arquivo.close();
}

```

```

BLYNK_WRITE(V2) {
    estadoRele = !estadoRele;
    arquivo = SPIFFS.open("/rele", "w");
    arquivo.write(estadoRele);
    arquivo.close();
    digitalWrite(RELEpin, !estadoRele);
}

void setup() {
    //Serial.begin(9600);
    boolean fileBegin = false;
    while (fileBegin != true) {
        fileBegin = SPIFFS.begin();
        delay(500);
    }
    arquivo = SPIFFS.open("/alarme", "r");
    alarme = arquivo.read();
    arquivo.close();
    arquivo = SPIFFS.open("/anterior", "r");
    estadoAnterior = arquivo.read();
    arquivo.close();
    arquivo = SPIFFS.open("/ativo", "r");
    ativo = arquivo.read();
    arquivo.close();
    arquivo = SPIFFS.open("/disparo", "r");
    disparo = arquivo.read();
    arquivo.close();
    arquivo = SPIFFS.open("/rele", "r");
    estadoRele = arquivo.read();
    arquivo.close();
    pinMode(PIRpin, INPUT);
    pinMode(LEDpin, OUTPUT);
    pinMode(BUZZERpin, OUTPUT);
    pinMode(RELEpin, OUTPUT);
    digitalWrite(LEDpin, ativo);
    digitalWrite(BUZZERpin, disparo);
    digitalWrite(RELEpin, !estadoRele);
    Blynk.begin(auth, ssid, pass);
    SPI.begin();
    mfrc522.PCD_Init();
    lcd.clear();
    timerRFID = timer.setInterval(50L, leituraRFID);
    delay(25);
    timerPIR = timer.setInterval(100L, leituraPIR);
    timerBUZZER = timer.setInterval(20000L, buzzerOn); // Tempo para
desativar o alarme após movimento antes de disparar o Buzzer
    timerATIVAR = timer.setInterval(20000L, ativarAlarme); // Tempo para
sair do local antes de disparar o alarme;
    timerLED = timer.setInterval(200L, LEDblink); // Faz piscar o LED
apenas
    timer.disable(timerBUZZER);
    timer.disable(timerATIVAR);
    timer.disable(timerLED);
    mudarAlarme();
    mudarRele();
    if (disparo == HIGH) {
        buzzerOn();
    }
}

```

```

void loop() {
    timer.run();
    Blynk.run();
}

void LEDblink () {
    estadoLED = !estadoLED;
    digitalWrite(LEDpin, estadoLED);
}

void buzzerOn () {
    digitalWrite(BUZZERpin, HIGH);
    Blynk.email("ALARME DISPARADO!", "ALARME DISPARADO!");
    Blynk.notify("ALARME DISPARADO!");
    timer.disable(timerBUZZER);
}

void ativarAlarme () {
    timer.disable(timerLED);
    timer.disable(timerATIVAR);
    digitalWrite(LEDpin, HIGH);
    ativo = true;
    arquivo = SPIFFS.open("/ativo", "w");
    arquivo.write(1);
    arquivo.close();
}

void leituraPIR () {
    if (ativo == false || disparo == true) {
        return;
    }
    if (digitalRead(PIRpin) == HIGH) {
        lcd.clear();
        lcd.print(0, 0, "    Movimento    ");
        lcd.print(0, 1, "    detectado    ");
        disparo = true;
        arquivo = SPIFFS.open("/disparo", "w");
        arquivo.write(1);
        arquivo.close();
        timer.enable(timerLED);
        timer.restartTimer(timerBUZZER);
        timer.enable(timerBUZZER);
    }
}

void leituraRFID () {
    if (millis() < timerLeituraRFID) {
        return;
    }
    // Procura por cartao RFID
    if (!mfrc522.PICC_IsNewCardPresent()) {
        return;
    }
    // Seleciona o cartao RFID
    if (!mfrc522.PICC_ReadCardSerial()) {
        return;
    }
    // Mostra UID na serial
    String conteudo = "";

```

```

byte letra;
for (byte i = 0; i < mfrc522.uid.size; i++) {
    // Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ");
    // Serial.print(mfrc522.uid.uidByte[i], HEX);
    conteudo.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " "));
    conteudo.concat(String(mfrc522.uid.uidByte[i], HEX));
}
// Serial.println();
conteudo.toUpperCase();

if (conteudo.substring(1) == "86 C1 17 C5") {
    alarme = !alarme;
    arquivo = SPIFFS.open("/alarme", "w");
    arquivo.write(alarme);
    arquivo.close();
    mudarAlarme();
    estadoAnterior = alarme;
    arquivo = SPIFFS.open("/anterior", "w");
    arquivo.write(estadoAnterior);
    arquivo.close();
    lcd.print(0, 1, "    via RFID    ");
    timerLeituraRFID = millis() + 2000; // Intervalo para próxima leitura
do RFID
}
else {
    lcd.clear();
    lcd.print(0, 0, "Cartão Inválido!");
    timerLeituraRFID = millis() + 2000; // Intervalo para próxima leitura
}
}

void mudarAlarme () {
    lcd.clear();
    if (alarme == false) {
        Blynk.virtualWrite(V0, HIGH);
        lcd.print(0, 0, "Alarme Desligado");
        disparo = false;
        arquivo = SPIFFS.open("/disparo", "w");
        arquivo.write(0);
        arquivo.close();
        ativo = false;
        arquivo = SPIFFS.open("/ativo", "w");
        arquivo.write(0);
        arquivo.close();
        timer.disable(timerLED);
        timer.disable(timerBUZZER);
        timer.disable(timerATIVAR);
        digitalWrite(BUZZERpin, LOW);
        digitalWrite(LEDpin, LOW);
    }
    else if (alarme == true && ativo == false) {
        Blynk.virtualWrite(V0, LOW);
        lcd.print(0, 0, " Alarme Ligado ");
        timer.enable(timerLED);
        timer.restartTimer(timerATIVAR);
        timer.enable(timerATIVAR);
    }
    else if (ativo == true && disparo == false) {
        Blynk.virtualWrite(V0, LOW);
        lcd.print(0, 0, " Alarme Ligado ");
    }
}

```

```

        digitalWrite(LEDpin, HIGH);
    }
    else if (disparo == true) {
        lcd.print(0, 0, "    Movimento    ");
        lcd.print(0, 1, "    detectado    ");
        timer.enable(timerLED);
        digitalWrite(BUZZERpin, HIGH);
    }
}

void mudarRele () {
    if (estadoRele == HIGH) {
        Blynk.virtualWrite(V2, LOW);
    }
    else {
        Blynk.virtualWrite(V2, HIGH);
    }
}

```