

Lampiran

1. Kode Arduino UNO Sensor IR

```
#include <DHT.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

#define DHTPIN 2
#define DHTTYPE DHT11

#define IR_IN 6
#define IR_OUT 7
#define RELAY_PIN 3

DHT dht(DHTPIN, DHTTYPE);
LiquidCrystal_I2C lcd(0x27, 16, 2);

// ===== VARIABEL =====
int jumlahOrang = 0;

unsigned long triggerTime = 0;
const unsigned long detectTimeout = 2000;

unsigned long waktuMasuk = 0;
unsigned long waktuKeluar = 0;
unsigned long waktuKipas = 0;

bool kipasON = false;

// ===== STATE =====
enum State {
    IDLE,
    MASUK_DETECT,
    KELUAR_DETECT
};

State state = IDLE;

void setup() {

    Serial.begin(9600);

    pinMode(IR_IN, INPUT_PULLUP);
    pinMode(IR_OUT, INPUT_PULLUP);
    pinMode(RELAY_PIN, OUTPUT);
```

```

digitalWrite(RELAY_PIN, LOW);

dht.begin();

lcd.begin();
lcd.backlight();

lcd.setCursor(0,0);
lcd.print("Smart Classroom");
lcd.setCursor(0,1);
lcd.print("Sistem Aktif");
delay(2000);
lcd.clear();

Serial.println("==== SISTEM AKTIF ====");
}

void loop() {

    bool inDetect = (digitalRead(IR_IN) == LOW);
    bool outDetect = (digitalRead(IR_OUT) == LOW);

    // ===== STATE MACHINE =====
    switch (state) {

        case IDLE:

            if (inDetect) {

                waktuMasuk = millis();
                Serial.println("Sensor IN aktif");

                state = MASUK_DETECT;
                triggerTime = millis();
            }

            else if (outDetect) {

                waktuKeluar = millis();
                Serial.println("Sensor OUT aktif");

                state = KELUAR_DETECT;
                triggerTime = millis();
            }

            break;

        case MASUK_DETECT:

```

```

        if (outDetect) {

            unsigned long delayDeteksi = millis() -
waktuMasuk;

            jumlahOrang++;

            Serial.print("Orang MASUK | Delay: ");
            Serial.print(delayDeteksi);
            Serial.print(" ms | Total: ");
            Serial.println(jumlahOrang);

            state = IDLE;
            delay(300);
        }

        else if (millis() - triggerTime > detectTimeout)
{
            state = IDLE;
        }

        break;

        case KELUAR_DETECT:

            if (inDetect) {

                unsigned long delayDeteksi = millis() -
waktuKeluar;

                if (jumlahOrang > 0) jumlahOrang--;

                Serial.print("Orang KELUAR | Delay: ");
                Serial.print(delayDeteksi);
                Serial.print(" ms | Total: ");
                Serial.println(jumlahOrang);

                state = IDLE;
                delay(300);
            }

            else if (millis() - triggerTime > detectTimeout)
{
                state = IDLE;
            }

            break;
        }

```

```

// ===== BACA SUHU =====
float suhu = dht.readTemperature();

if (isnan(suhu)) return;

// ===== LOGIKA KIPAS =====
if (suhu >= 29 && jumlahOrang >= 2) {

    if (!kipasON) {

        waktuKipas = millis();

        Serial.print("KIPAS MENYALA | Suhu: ");
        Serial.print(suhu);
        Serial.print("C | Orang: ");
        Serial.println(jumlahOrang);

    }

    digitalWrite(RELAY_PIN, HIGH);
    kipasON = true;
}

else {

    if (kipasON) {
        Serial.println("KIPAS MATI");
    }

    digitalWrite(RELAY_PIN, LOW);
    kipasON = false;
}

// ===== LCD =====

lcd.setCursor(0,0);

if (jumlahOrang > 0) {

    lcd.print("Digunakan (");
    lcd.print(jumlahOrang);
    lcd.print(" org)");
    lcd.print(" ");

}

else {

    lcd.print("Ruangan Kosong ");

```

```

}

lcd.setCursor(0,1);

lcd.print((int)suhu);
lcd.print((char)223);
lcd.print("C ");

lcd.print("Kipas:");

if (kipasON) lcd.print("ON ");
else lcd.print("OFF");

delay(100);
}

```

2. Kode NodeMCU ESP8266

```

#include <ESP8266WiFi.h>
#include <FirebaseESP8266.h>

#define WIFI_SSID "DELETE.COM 4G"
#define WIFI_PASSWORD "75770212"

#define FIREBASE_HOST "https://realtime-database-ldr-
default-rtdb.firebaseio.com/"
#define FIREBASE_AUTH "AIzaSyATafW4tkhdoraJ-
IjhE6HuPIFE7gs2nYA"

FirebaseData firebaseData;

void setup() {
  Serial.begin(9600);

  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("Connecting WiFi");

  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
  }

  Serial.println("Connected");

  Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
  Firebase.reconnectWiFi(true);
}

```

```
void loop() {  
  
    if (Serial.available()) {  
  
        String data = Serial.readStringUntil('\n');  
  
        int jumlah = data.substring(0,  
data.indexOf(',')).toInt();  
        int suhu = data.substring(data.indexOf(',') +  
1).toInt();  
  
        Firebase.setInt(firebaseData,  
"/smartclassroom/jumlah_mahasiswa", jumlah);  
        Firebase.setInt(firebaseData,  
"/smartclassroom/suhu_ruangan", suhu);  
  
    }  
  
}
```