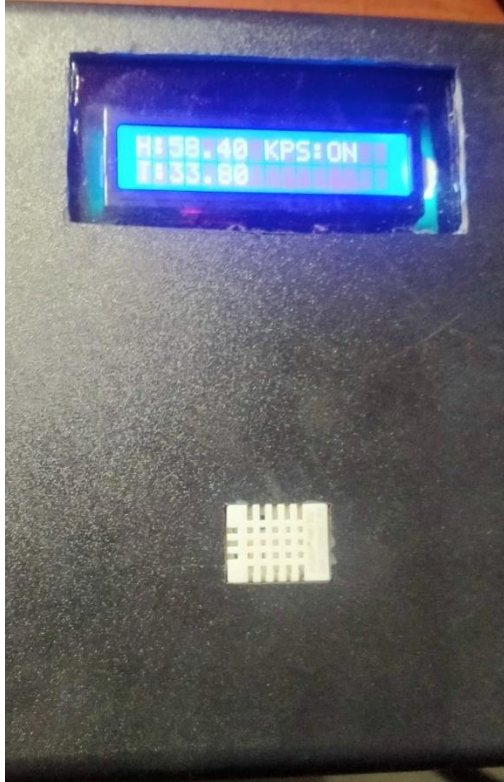


## DAFTAR PUSTAKA

- Arief Guritno, Dwi jatmiko Nugro, Rakhmad Yatim, 2014. Implementasi Sensor SHT11 Untuk Pengkondisian Suhu dan Kelembaban Relatif Berbantuan Mikrokontroler. Seminar Nasional Sains dan Teknologi 2014 ISSN: 2407 – 1846.
- Faudin, Agus. 2017. Apa Itu Modul ESP8266 Beserta Penjelasannya di <https://nyebarilmu.com> (diakses 15 Januari 2019).
- Haryanto, Budi, Nanang Simail dan Eko Joni Pristiano. 2018. Sistem Monitoring Suhu dan Kelembaban Secara Nirkabel Pada Budidaya Hidroponik. Jurnal. Teknik Elektro Universitas Islam Negeri Sunan Gunung Djati.
- Kho, Dickson. 2019. Prinsip Kerja DC Power Supply. [Internet] di <https://teknikelektronika.com> (diakses 15 Januari 2019).
- Lamudi. 2014. Pengertian Exhaust Fan dan Cara Memilihnya. [Internet] di <https://lamudi.co.id> (diakses 15 Januari 2019)
- Mahasari, Susanti. 2017. Rancang Bangun Internet Of Things (IOT) Node Sensor Cuaca Dengan Menggunakan Website, Politeknik Negeri Sriwijaya.
- Rachmat Munggaran Zuansah. (2015). Rancang bangun kontrol suhu dan kelembaban pada sistem distribusi kubikel 20 kV. Indonesia: Jurusan Fisika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jendral Ahmad Yani
- Saleh, Muhammad. 2017. Rancang Bangun Sistem Keamanan Rumah Menggunakan Relay. Jurnal. Universitas Suryadarma
- SPLN 52-3 (1983). Pengoprasian Kubikel 20 kV (Pengertian dan Fungsi Kubikel). Jakarta
- Tim Fakultas Teknologi Industri. 2019. Pedoman Tugas Akhir. Surabaya: Fakultas Teknologi Industri Universitas PGRI Adi Buana Surabaya
- Wicaksana, Irsandi Satria, Firdaus Iman Ubaidillah, Yeni Prasetio Hadi, Sandi Tyas Wahyu, dan Istiadi “Perancangan Sistem Monitoring Temperatur Gudang Berbasis Internet Of Things (IOT)” Konferensi. Teknik Elektro Universitas Widyagama Malang, 2018

## LAMPIRAN 1

### PRODUK



## LAMPIRAN 2

### PROGRAM ARDUINO

```
#include <Wire.h>
#include "DHT.h"
#include <WiFi.h>
#include "ThingSpeak.h"
#include "secrets.h"
#include<stdlib.h>
#include<LiquidCrystal.h>
const int rs = 19, en = 23, d4 = 4, d5 = 16, d6 = 17, d7 = 18;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

////////////////////////////////////////RTC////////////////////////////////////////
const int I2C_ADDRESS = 0x68;

const char* days[] = {"Minggu", "Senin", "Selasa", "Rabu", "Kamis", "Jumat",
"Sabtu"};
const char* months[] = {"Januari", "Februari", "Maret", "April", "Mei", "Juni",
"Juli", "Augustus", "September", "October", "November", "Desember"};

byte second = 0;
byte minute = 0;
byte hour = 0;
byte weekday = 0;
byte monthday = 0;
byte month = 0;
byte year = 0;

////////////////////////////////////////ESP & DHT////////////////////////////////////////
char ssid[] = SECRET_SSID; // your network SSID (name)
char pass[] = SECRET_PASS; // your network password
int keyIndex = 0; // your network key Index number (needed only for
WEP)
WiFiClient client;

unsigned long myChannelNumber = SECRET_CH_ID;
const char * myWriteAPIKey = SECRET_WRITE_APIKEY;
String myStatus = "";

#define DHTPIN 15 // Digital pin connected to the DHT sensor
// #define DHTTYPE DHT11 // DHT 11
#define DHTTYPE DHT22 // DHT 22 (AM2302), AM2321
// #define DHTTYPE DHT21 // DHT 21 (AM2301)

DHT dht(DHTPIN, DHTTYPE);
```

```

int maxHum = 60;
int minHum = 50;

#define fan 33

void setup(){
  Wire.begin(25, 26);
  Serial.begin(9600);
  Serial.println(F("DHTxx test!"));
  WiFi.mode(WIFI_STA);
  ThingSpeak.begin(client); // Initialize ThingSpeak
  dht.begin();
  lcd.begin(16,2);
  pinMode(fan, OUTPUT); // kipas
  //digitalWrite(fan, LOW);
  //mengaturWaktu();
  lcd.setCursor(0,0);
  lcd.print("**TUGAS AKHIR**");
  lcd.setCursor(0,1);
  lcd.print("**TEKNIK ELEKTRO**");
  delay(3000);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("* MOH RIFQI A *");
  lcd.setCursor(0,1);
  lcd.print("*NIM: 163600049*");
  delay(3000);
  lcd.clear();
}

void loop(){
  printTime();
  jadwal();
  SUHU();
  delay(500);
}

void THINGSPEAK(){
  // Wait a few seconds between measurements.
  //delay(2000);
  if(WiFi.status() != WL_CONNECTED){
    Serial.print("Attempting to connect to SSID: ");
    Serial.println(SECRET_SSID);
    while(WiFi.status() != WL_CONNECTED){
      WiFi.begin(ssid, pass); // Connect to WPA/WPA2 network. Change this line
if using open or WEP network
      Serial.print(".");
      delay(5000);
    }
  }
}

```

```

Serial.println("\nConnected.");
lcd.setCursor(8,1);
lcd.print("Connect");
}

// Reading temperature or humidity takes about 250 milliseconds!
// Sensor readings may also be up to 2 seconds 'old' (its a very slow sensor)
float h = dht.readHumidity();
// Read temperature as Celsius (the default)
float t = dht.readTemperature();
// Read temperature as Fahrenheit (isFahrenheit = true)
float f = dht.readTemperature(true);
ThingSpeak.setField(2, h);
ThingSpeak.setField(3, t);
// Check if any reads failed and exit early (to try again).
if (isnan(h) || isnan(t) || isnan(f)) {
  Serial.println(F("Failed to read from DHT sensor!"));
  return;
}

// Compute heat index in Fahrenheit (the default)
float hif = dht.computeHeatIndex(f, h);
// Compute heat index in Celsius (isFahreheit = false)
float hic = dht.computeHeatIndex(t, h, false);

Serial.print(F("Humidity: "));
lcd.setCursor(0,0);
lcd.print("H:");
lcd.print(h);
Serial.print(h);
Serial.print(F("% Temperature: "));
Serial.print(t);
lcd.setCursor(0,1);
lcd.print("T:");
lcd.print(t);
Serial.println(F("°C "));
/*Serial.print(f);
Serial.print(F("°F Heat index: "));
Serial.print(hic);
Serial.print(F("°C "));
Serial.print(hif);
Serial.println(F("°F"));*/

// set the status
ThingSpeak.setStatus(myStatus);

// write to the ThingSpeak channel
int x = ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);

```

```

if(x == 200){
  Serial.println("Channel update successful.");
  lcd.setCursor(8,1);
  lcd.print("success");
}
else{
  Serial.println("Problem updating channel. HTTP error code " + String(x));
  lcd.setCursor(8,1);
  lcd.print("error ");
}

//delay(20000); // Wait 20 seconds to update the channel again
}

```

```

////////////////////////RTC////////////////////////////////////////

```

```

void printTime() {
  char buffer[3];
  const char* AMPM = 0;
  membacaWaktu();
  Serial.print(days[weekday-1]);
  Serial.print(" ");
  Serial.print(monthday);
  Serial.print(" ");
  Serial.print(months[month-1]);
  Serial.print(" 20");
  Serial.print(year);
  Serial.print(" ");
  if (hour > 12) {
    hour -= 12;
    AMPM = " PM";
  }
  else AMPM = " AM";
  Serial.print(hour);
  Serial.print(":");
  sprintf(buffer, "%02d", minute);
  Serial.print(buffer);
  Serial.print(":");
  sprintf(buffer, "%02d", second);
  Serial.print(buffer);
  Serial.println(AMPM);
}

void membacaWaktu() {

```

```

  Wire.beginTransmission(I2C_ADDRESS);

```

```

Wire.write(byte(0));
Wire.endTransmission();
Wire.requestFrom(I2C_ADDRESS, 7);
second = bcdToDec(Wire.read());
minute = bcdToDec(Wire.read());
hour = bcdToDec(Wire.read());
weekday = bcdToDec(Wire.read());
monthday = bcdToDec(Wire.read());
month = bcdToDec(Wire.read());
year = bcdToDec(Wire.read());

}
void mengaturWaktu() {

Wire.beginTransmission(I2C_ADDRESS);
Wire.write(byte(0));
Wire.write(decToBcd(10)); // second
Wire.write(decToBcd(48)); // minute
Wire.write(decToBcd(4)); // hour
Wire.write(decToBcd(6)); // weekday
Wire.write(decToBcd(15)); // date
Wire.write(decToBcd(4)); // month
Wire.write(decToBcd(21)); // year
Wire.write(byte(0));
Wire.endTransmission();

}
byte decToBcd(byte val) {
return ((val/10*16) + (val% 10));
}
byte bcdToDec(byte val) {
return ((val/16*10) + (val% 16));
}

////////////////////////////////////
void jadwal() {
mulai:
membacaWaktu();
if(minute == 4 & second == 59){
THINGSPEAK();
}
if(minute == 9 & second == 59){
THINGSPEAK();
}
if(minute == 14 & second == 59){
THINGSPEAK();
}
if(minute == 19 & second == 59){

```

```

THINGSPEAK();
}
if(minute == 24 & second == 59){
THINGSPEAK();
}
if(minute == 29 & second == 59){
THINGSPEAK();
}
if(minute == 34 & second == 59){
THINGSPEAK();
}
if(minute == 39 & second == 59){
THINGSPEAK();
}
if(minute == 44 & second == 59){
THINGSPEAK();
}
if(minute == 49 & second == 59){
THINGSPEAK();
}
if(minute == 54 & second == 59){
THINGSPEAK();
}
if(minute == 59 & second == 59){
THINGSPEAK();
}
}
}

//////////////////////////////////DHT//////////////////////////////////
void SUHU(){
  float h = dht.readHumidity();
  // Read temperature as Celsius (the default)
  float t = dht.readTemperature();
  // Read temperature as Fahrenheit (isFahrenheit = true)
  float f = dht.readTemperature(true);

  /*if (isnan(h) || isnan(t) || isnan(f)) {
    Serial.println(F("Failed to read from DHT sensor!"));
    return;*/

  // Compute heat index in Fahrenheit (the default)
  float hif = dht.computeHeatIndex(f, h);
  // Compute heat index in Celsius (isFahreheit = false)
  float hic = dht.computeHeatIndex(t, h, false);

  Serial.print(F("Humidity: "));
  Serial.print(h);

```



```
lcd.setCursor(0,0);
lcd.print("H:");
lcd.print(h);
Serial.print(F("% Temperature: "));
Serial.print(t);
lcd.setCursor(0,1);
lcd.print("T:");
lcd.print(t);
Serial.println(F("°C "));

if(h > maxHum) {
  lcd.setCursor(8,0);
  lcd.print("KPS:ON ");
  digitalWrite(fan, HIGH);
}
if(h < minHum) {
  lcd.setCursor(8,0);
  lcd.print("KPS:OFF ");
  digitalWrite(fan, LOW);
}
}
```



# UNIVERSITAS PGRI ADI BUANA SURABAYA

## FAKULTAS TEKNIK

Program Studi : Teknik Lingkungan – Perencanaan Wilayah Kota  
Teknik Industri – Teknik Elektro - PVKK

KAMPUS II: Jl. Dukuh Menanggal XII/4 ☎ (031) 8281181 Surabaya 60234

Website : [www.ft.unipasby.ac.id](http://www.ft.unipasby.ac.id) E-mail : [ft@unipasby.ac.id](mailto:ft@unipasby.ac.id)

### BERITA ACARA UJIAN SKRIPSI

Pada

Hari, tanggal : Rabu, 30 Juni 2021  
Jam : 09.45 - 10.30  
Tempat : di lakukan secara daring

Telah dilaksanakan Ujian Skripsi:

Nama Mahasiswa : Moh. Rifqi Afifudin  
NIM : 162600049  
Program Studi : TEKNIK ELEKTRO  
Judul : Alat kontrol Monitoring suhu dan kelembaban  
pada sistem distribusi Tenaga Listrik  
kubikel 20 kv berbasis website  
Bidang Keahlian :  
Tanda Tangan :

Saran-saran perbaikan :

- 1) Latar belakang lebih general lagi
- 2) Perbaiki font dan tata cara penulisan
- 3) Ruang lingkup belum ada
- 4) Kesimpulan harus menjawab dari rumusan masalah

#### Tim Penguji

Nama ( Tanda tangan )

1. Dwi. Hastuti, S. Kom, MT.

2. Winarno, FB., IR, M. Eng.

\*) Jangka waktu perbaikan Skripsi dua minggu setelah ujian.

Apabila waktu tersebut tidak dipenuhi, maka nilai Ujian Skripsi dianggap batal dan mahasiswa yang bersangkutan diwajibkan mengulang Ujian lisan



# UNIVERSITAS PGRI ADI BUANA SURABAYA

## FAKULTAS TEKNIK

Program Studi : Teknik Lingkungan – Perencanaan Wilayah Kota  
Teknik Industri – Teknik Elektro - PVKK

KAMPUS II: Jl. Dukuh Menanggal XII/4 ☎ (031) 8281181 Surabaya 60234

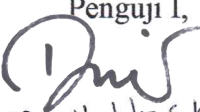
Website : [www.ft.unipasby.ac.id](http://www.ft.unipasby.ac.id) E-mail : [ft@unipasby.ac.id](mailto:ft@unipasby.ac.id)

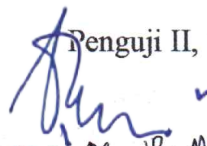
### FORM REVISI SKRIPSI

Nama Mahasiswa : Moh. Rizki Afifudin  
NIM : 163600049  
Fakultas / Progdi : Fakultas Teknik / Teknik Elektro  
Judul Skripsi : Abt kontrol Monitoring suhu dan kelembaban pada sistem distribusi Tenaga Listrik kubikel 20kv berbasis website  
Ujian Tanggal : 30 - Juni - 2021

No Bab.	Tanggal	Materi Konsultasi	Keterangan Catatan	Tanda Tangan Penguji
I	01 - Juli - 2021	Pendahuluan	Acc	(Dwi)
II	01 - Juli - 2021	kajian Pustaka	Acc	(Dwi)
III	02 - Juli - 2021	Metodologi Penelitian	Acc	(Dwi)
IV	02 - Juli - 2021	Metode dan Pembahasan	Acc	(Dwi) ✓
V	02 - Juli - 2021	kesimpulan dan Saran	Acc	(Dwi) ✓

Disetujui Dosen Penguji  
Pada Tanggal, ....02.juli.2021.....

Penguji I,  
  
(Dwi Hastuti, S.kom, MT.)

Penguji II,  
  
(Winarno PB., IR., M.Eng)

- Penyelesaian Revisi paling lambat 2 minggu dari pelaksanaan Ujian Skripsi.
  - Pengetikan, penjilidan, penandatanganan Skripsi dan mengumpulkan Skripsi paling lambat 2 minggu dari revisi.
- Apabila sampai batas waktu tersebut ( point 1,a dan b ) mahasiswa belum menyelesaikan revisi dan tanda tangan, maka **Ujian dinyatakan Gugur.**
- Foto copy Form Revisi diserahkan ke Program Studi.
  - Skripsi yang sudah direvisi diserahkan ke Fakultas tiga eksemplar untuk dijilid.