`

DAFTAR PUSTAKA

1. Risqiwati, Diah, Purnomo, Mauridhi Hery, dan Priyadi, Ardyono. 2012. "Metode Pendeteksian Lokasi Gangguan Pada Jaring Tiang Distribusi 20 KV". Java Journal Electrical And Electronics Engineering. Vol 10. Untoro

2. Roostaee, S., Thomas, M. s, Mehfuz, S., 2017. Experimental studies on impedance based fault location for long transmission lines. Springer Open 14. https://doi.org/DOI 10.1186/s41601-017-0048-y

3. https://embeddednesia.com/v1/tutorial-esp32-program pertama-dengan-arduino-ide/ dan https://www.ardutech.com/mengenal-esp32-development-kit-untuk-iot-internet-of-things/, diakses tanggal 10 Desember 2020

4. Android Developer Guide : http://developer.android.com(28 Juni 2020, jam 12.00)

5. Android TM Programing Tutorial, Mark L.Murphy, CommonsWare Press, http://www.bmkg.go.id/ (28 Juni 2020, jam 12.30)

6. Al Qoyyim, T.A., Penangsang, O., Aryani, N.K., 2017. Penentuan Lokasi Gangguan Hubung Singkat pada Jaringan Distribusi 20 kV Penyulang Tegalsari Surabaya dengan Metode Impedansi Berbasis GIS (Geographic Information System). J. Tek. ITS vol.6, No.1, 6.

7. Azzahra Rahmah, 2019, https://rumus.co.id/hukum-kirchoff/ , diakses tanggal 10 desember 2020.

8. **Ardutech,** 2020, https://www.ardutech.com/mengenal-esp32-development-kit-untuk-iot-internet-of-things/, diakses tanggal 10 Desember 2020

9. Unisa , 2020 *, https://psti.unisayogya.ac.id/2020/01/06/apa-itu-mit-app-inventor/* , diakses tanggal 10 Desember 2020

10. Karyana , 2013 . "Pedoman dan Petunjuk Sistem Proteksi Transmisi dan Gardu Induk Jawa Bali"

11. Rifqi Dwiputra Parawangsa ,2017. "Rancang Bangun Prototype Smarthome menggunakan telegram mesegger berbasis arduino".

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
Website : www.ft.unipasby.ac.id E-mail : ft@unipasby.ac.id

BERITA ACARA BIMBINGAN SKRIPSI

Form Skripsi-03

Nama	: JAWI KUSWANTO
NIM	: 193609015
Program Studi	: TEKNIK ELEKTRO
Pembimbing	: SAGITA ROCHMAN, ST. M.Si
Periode Bimbingan	: Gasal/Genap Tahun 2020 / 2021
Judul Skripsi	APLIKASI DAN SISTEM MONITORING PENENTUAN LOKASI TITIK GANGGUAN SALURAN TRANSMISI SUTT BERBASIS *ANDROID*

KEGIATAN KONSULTASI / BIMBINGAN

No	Tanggal	Materi pembimbingan	Keterangan	Paraf
1.	11 Desember 2020	Revisi Latar belatang	Acc	
2	25 Desember 2020	Revisi Rumusan Masalah	Acc	
3	28 Desember 2020	Bab II penambahan pustaka/teori	Acc	
4	8 Januari 2021	Bab III Revisi Desain produk	Acc	
5	11 Januari 2021	metode pengambilan data	Acc	
6	15 Januari 2021	Bab IV Revisi analisis data	Acc	
7.		Penyajian data	Acc	
8	18 - Januari 2021	Bab V Revisi kesimpulan/saran	Acc	

Dinyatakan selesai tanggal : 21 Januari 2021

Surabaya, 21 Januari 2021

Mengetahui,
Ketua Program Studi

AKHMAD FAUZI, S.Si, M.Si

Pembimbing,

SAGITA ROCHMAN, ST.M.Si

Mahasiswa,

JAWI KUSWANTO

UNIVERSITAS PGRI ADI BUANA SURABAYA
FAKULTAS TEKNOLOGI INDUSTRI
Program Studi : Teknik Industri – Teknik Elektro
KAMPUS II: Jl. Dukuh Menanggal XII/4 ☎ (031) 8281181

FORM REVISI PROPOSAL TUGAS AKHIR

Nama Mahasiswa : Jawi Kuswanto
NIM : 193609015
Fakultas / Progdi : Fakultas Teknik / T. Elektro
Judul Proposal Tugas Akhir : Aplikasi dan Sistem Monitoring Pemeliharaan Tank Lokasi Gangguan Saluran Transmisi berbasis Android

Ujian Tanggal :

No Bab.	Tanggal	Materi Konsultasi	Keterangan Catatan	Tanda Tangan Penguji
I	02 Feb 2021	- Latar belakang. Penelitian	-	
II	05. Feb 2021	- Ruang Lingkup	-	
III	08 Feb 2021	- Hasil uji ditambah	-	
IV	10 Feb 2021	- Referensi penelitian	-	
V	10 Feb 2021	- Tata Cara Penulisan	-	

Disetujui Dosen Penguji
Pada Tanggal,.....

Penguji I,

(Drs. Budi. Prijo. S)

Penguji II,

(Rarasaing Diptya. W.)

1. a. Penyelesaian Revisi paling lambat 2 minggu dari pelaksanaan Ujian Proposal Tugas Akhir.
   b. Pengetikan, penjilidan, penandatangani Proposal Tugas Akhir dan mengumpulkan Proposal Tugas Akhir paling lambat 2 minggu dari revisi.
2. Apabila sampai batas waktu tersebut (point 1,a dan b) mahasiswa belum menyelesaikan revisi dan tanda tangan, maka **Ujian dinyatakan Gugur**.
3. a. Foto copy Form Revisi diserahkan ke Program Studi
   b. Proposal Tugas Akhir yang sudah direvisi diserahkan ke Program Studi 2 (Dua) eksemplar yang sudah dijilid softcover. (warna cover sesuai masing-masing program studi).

`

LAMPIRAN

Source Code Alat

```
// Inisiasi Awal


#define BLYNK_PRINT Serial
#include <ESP32WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <PZEM004T.h>
#include <Wire.h>
//#include <LCD.h> // D4 SDA & D3 SCL //
#include <LiquidCrystal_I2C.h>
//inisiasi LCD DISPLAY
LiquidCrystal_I2C lcd(0x27, 16, 2);
//LiquidCrystal_I2C lcd(0x27 ,2,1,0,4,5,6,7,3, POSITIVE);
//inisiasi PZEM-004T
PZEM004T pzem(12,14);//nodemcu berarti D6 dan D5 sebagai RX, TX
IPAddress ip (192,168,1,1);
//Inisiasi variabel pengukuran
float voltage_blynk=0;
float current_blynk=0;
float power_blynk=0;
float energy_blynk=0;
float faktordaya_blynk=0;
89// auth yang tertera pada apps BLYNK
char auth[] = "tlQr3t2YtBDNyfgbB0EDL9ESpGwR1UDq";
//SSID jaringan wifi yang digunakan
```

`

```
//Password jaringan tersebut
char ssid[] = "PASSION";
char pass[] = "Unipa1";
BlynkTimer timer;
unsigned long lastMillis=0;
//inisiasi variabel cosphi
/*int pin = 15;
float pulsewidth=0;
float pf=0;
float phase=0;
float rads = 57.29577951;
float degree = 360;
float freq = 50;
void powerfactor ()
{
pulsewidth = pulseIn (pin,HIGH);
phase =( degree * freq * pulsewidth / 1000000);
pf = cos(phase * 3.1415 / 180 );
//lcd.setCursor (0,0);
// lcd.print(phase);
90// lcd.setCursor(0,1);
//lcd.print (pf);
faktordaya_blynk=pf;
Serial.print ("pulse="); Serial.print(pulsewidth);
Serial.print("Cosphi= "); Serial.print(pf);
Serial.print("Sudut= "); Serial.print(phase);
```

```
`


//Serial.print("Time="); Serial.print(lastMillis );
delay (500);
}
*/
void mySensorDataSend()
{ float p = (pzem.power(ip)*0.94);
float v = pzem.voltage(ip);
float i = pzem.current(ip);
/// Read meter PZEM
{
if(v < 0.0) v=0.0;
{ voltage_blynk =v; } //V
Serial.print("V= "); Serial.print(v); Serial.print(" Volt ");
lcd.setCursor(0,0);
lcd.print("V= ");
lcd.setCursor(2,0);
91 lcd.print(v);
}
{
if(i < 0.0 ) i=0.0;
{ current_blynk=i; }
Serial.print("I= "); Serial.print(i); Serial.print(" Ampere ");
lcd.setCursor(9,0);
lcd.print("I= ");
lcd.setCursor(11,0);
lcd.print(i);
```

```
`


}
{
if(p < 0.0) p=0.0;
{power_blynk=(p); } //kW
Serial.print("P= "); Serial.print(p); Serial.print(" Watt ");
lcd.setCursor(9,1);
lcd.print("P= ");
lcd.setCursor(11,1);
lcd.print(p);
}
92 {
float pf = (p/(v*i));
/*((pzem.power(ip))/((pzem.voltage(ip))*(pzem.current(ip))));*/
if ( pf > 1.0 )
{ ( pf = 1.0); faktordaya_blynk=pf; }
else if ( pf < 0.0 )
{( pf = 0.0); faktordaya_blynk=pf; }
else
{
//pf=((pzem.power(ip))/((pzem.voltage(ip))*(pzem.current(ip))));
pf = (p/(v*i));
}
lcd.setCursor(0,1);
lcd.print("Pf=");
lcd.setCursor(3,1);
lcd.print(pf);
```

`

Serial.print("PF= "); Serial.print(pf); Serial.print(" ");

faktordaya_blynk=pf;

}

{

float e = pzem.energy(ip);

if(e < 0.0) e=0.0;

{ energy_blynk =e; } ///kWh

Serial.print("E= "); Serial.println(e);

// lcd.setCursor(0,1);

// lcd.print("E=");

93 // lcd.setCursor(2,1);

// lcd.print(e);

}

//delay(8000);

// lcd.print(" \n");

//Blynk.virtualWrite(V5,millis()/1000);

//Publish data every 10 seconds (10000 milliseconds). Change this value to publish at

a different interval.

if (millis() - lastMillis > 10000) {

lastMillis = millis();

//Serial.print("Time Elapsed Total = ");

//Serial.println(lastMillis);

Blynk.virtualWrite(V1, voltage_blynk);

Blynk.virtualWrite(V2, current_blynk);

Blynk.virtualWrite(V3, power_blynk);

```
`


Blynk.virtualWrite(V4, energy_blynk );

Blynk.virtualWrite(V5, lastMillis );

Blynk.virtualWrite(V6, faktordaya_blynk);

}

}

void setup(){

// Debug console

Serial.begin(9600);

pzem.setAddress(ip);

9495

Blynk.begin(auth, ssid, pass, "blynk-cloud.com",8442);

timer.setInterval(10000L, mySensorDataSend);

Wire.begin(2,0);

lcd.init(); // initializing the LCD

lcd.backlight(); // Enable or Turn On the backlight

}

void loop(){

// powerfactor();

Blynk.run();

timer.run(); // Initiates BlynkTimer

}
```

`

LAMPIRAN 3 :

NODEMCU LOLIN V3

Lolin NodeMCU V3 is an open source IoT platform. It uses the Lua scripting language. The eLua project is the basis of board, and built on the ESP8266 SDK 1.4. NodeMCU uses many open source projects, such as lua-cjson, and spiffs. The NodeMCU runs on the ESP8266 Wi-Fi SoC, and hardware which based on the ESP-12 module.

The Lolin NodeMCU V3 board ads USB/UART converter chip as well as decoupled LDO power supply. Also the board adds 2 miniature push buttons. The most important feature is that it breaks out all ESP8266 pins to board headers. The board headers are breadboard compatible 2.54 mm pitch headers.

The Lolin NodeMCU board uses the CH340G USB/UART converter chip. You will need to download and install the proper driver to get going with the development. You can find the drivers here:

NodeMCU CH340/CH340G Driver Download page (if not automatically recognised): Click Here

For MAC users please check this link .