Lampiran 1

Tabel Spesifikasi Sensor Tegangan (ZMPT101B)

No	Nama	Keterangan
1	Jangkauan frekuensi	50 ~ 60 Hz
2	Rasio	1000 : 1000
3	Derajat Sudut Error	<20 (50 Ω)
4	Tegangan Operasional	20 Vac ~ 400 Vac
5	Arus Operasional	0 ~ 3mA
6	Suhu operasionl	-40C -130 C
7	Linearitas	≤0.2% (20% dot ~ 120% dot)
8	Kelas Akurasi	0,2
9	DC reistance pada 20 C	110 Ω
10	Isolasi	Epoxy
11	Isolasi Resistansi	>100 M Ω
12	Vcc	5 Vdc
13	PinOut	Analog Sinyal

Lampiran 2

Program C++ “ Rancang Bangun Sistem Monitoring dan Kontrol Tegangan Listrik Pada Pompa Irigasi Sawah Berbasis Arduino” dengan Arduino IDE

```
#define BLYNK_TEMPLATE_ID "TMPL3OSu2vEz"
#define BLYNK_DEVICE_NAME "skripsi2"
#define BLYNK_AUTH_TOKEN "lVn83HNsHQtWrgpoCpCgrld37inFcrfr"
#define BLYNK_PRINT Serial
#include "EmonLib.h"
#include <ESP8266_Lib.h>
#include <BlynkSimpleShieldEsp8266.h>
char auth[] = BLYNK_AUTH_TOKEN;
char ssid[] = "BoltWifi";
char pass[] = "87b761q8";
SoftwareSerial EspSerial(2, 3); // RX, TX
// Your ESP8266 baud rate:
#define ESP8266_BAUD 9600
int count = 0;
EnergyMonitor emon1;
ESP8266 wifi(&EspSerial);
const int ledPin = 7;
const int btnPin = 6;
const int relay = 5;
BlynkTimer timer;
void checkPhysicalButton();
int ledState = LOW;
int btnState = HIGH;
int restate = HIGH;
```

```
// Every time we connect to the cloud...
BLYNK_CONNECTED() {
  // Request the latest state from the server
  Blynk.syncVirtual(V1);
  // Alternatively, you could override server state using:
  //Blynk.virtualWrite(V1, ledState);
}
// When App button is pushed - switch the state
BLYNK_WRITE(V1) {
  ledState = param.asInt();
  digitalWrite(ledPin, ledState);
  digitalWrite(relay, !ledState);
}
void checkPhysicalButton()
{
  if (digitalRead(btnPin) == LOW) {
    // btnState is used to avoid sequential toggles
    if (btnState != LOW) {
      // Toggle LED state
      ledState = !ledState;
      digitalWrite(ledPin, ledState);
            digitalWrite(relay, !ledState);
      // Update Button Widget
      Blynk.virtualWrite(V1, ledState);
    }
    btnState = LOW;
  }
  else {
    btnState = HIGH;
  }
}
void checkVolt()
{
 emon1.calcVI(20,2000);
  double supplyVoltage  = emon1.Vrms;
if ( (supplyVoltage > 200)&&(supplyVoltage < 240) ) {
count = 0;
}
else {
count++;
}
    if (count >= 10) {
      // Toggle LED state
      ledState = LOW;
      digitalWrite(ledPin, ledState);
      digitalWrite(relay, HIGH);
      // Update Button Widget
      Blynk.virtualWrite(V1, ledState);
    }
Serial.println (count);
     Blynk.virtualWrite(V0, supplyVoltage);
}
void setup()
{
  Serial.begin(9600);
analogReference(EXTERNAL);
  emon1.voltage(1, 234.26, 1.7);
  emon1.current(0, 35);
  // Set ESP8266 baud rate
  EspSerial.begin(ESP8266_BAUD);
  delay(10);
  //Blynk.begin(auth, wifi, ssid, pass);
  // You can also specify server:
  Blynk.begin(auth, wifi, ssid, pass, "blynk.cloud", 80);
  //Blynk.begin(auth, wifi, ssid, pass, IPAddress(192,168,1,100), 8080);
  pinMode(ledPin, OUTPUT);
  pinMode(relay, OUTPUT);
  pinMode(btnPin, INPUT_PULLUP);
  digitalWrite(ledPin, ledState);
  digitalWrite(relay, !ledState);
  // Setup a function to be called every 100 ms
  timer.setInterval(100L, checkPhysicalButton);
    timer.setInterval(1000L, checkVolt);
}
void loop()
{
  Blynk.run();
  timer.run();
}
```