DAFTAR PUSTAKA

Endrayanto, Royyannuur Kurniawan (2019). Advanced Encryption Standard (AES) pada Modul Internet of Things (IoT). Universitas Brawijaya

Eugene.C ; Hanapi,Gunawan. Drs.Ir.(1993). Mesin dan Rangkaian Listrik. Edisi Keenam. ITB. Bandung

Fifadhlillah, J. (2004T). Prototype kWh Meter Dengan Transmisi Nirkabel Berbasis Arduino Uno. Universitas Negeri Yogyakarta.

Herandi, Galla. dkk (2019). *Monitoring Biaya Dan Pengukuran Konsumsi Daya Listrik Berbasis Arduino Mega2560 Menggunakan Web*. Universitas Negeri Yogyakarta.

H. D. Septama, "*Smart Wirehouse*: Sistem Pemantauan Dan Kontrol Otomatis Suhu Serta Kelembaban Gudang," *Seminar Nasional Inovasi, Teknologi, dan Aplikasi (SeNTiA)*, p. 1, 2018.

Junaidi, dkk (2017). *Analisis Kapasitas Dan Kebutuhan Daya Listrik Untuk Menghemat Penggunaan Energi Listrik Di Fakultas Teknik Universitas Tanjungpura.* Pontianak: Universitas Tanjungpura,

Nurul Hidayati Lusita Dewi, dkk (2019). Prototype Smart Home Dengan Modul Nodemcu ESP8266 Berbasis Internet Of Things (IOT). Universitas Islam Majapahit

Rianto, Y. (2020) ‘Mendekteksi Gerakan Kamera Menggunakan Wemos D1 R1 Berbasis IOT’, (100), pp. 1–28.

Wicaksono, Muhammad Fajar (2017). *Implementasi Modul Wifi Nodemcu Esp8266 Untuk Smart Home.* Bandung: Universitas Komputer.

LAMPIRAN

Lampiran 1 Pengujian Alat Elektronik

Lampiran 2 Source Code Arduino IDE

```
#include <PZEM004Tv30.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>
#include <WiFiClient.h>

const char* ssid = "tamtamii";
const char* password = "tamtamiii";

//Your Domain name with URL path or IP address with path
String serverGetData = "http://192.168.244.147/listrikrumahkost/api/insert.php";

// the following variables are unsigned longs because the time, measured in
// milliseconds, will quickly become a bigger number than can be stored in an int.
unsigned long lastTime = 0;

unsigned long timerDelay = 850;

LiquidCrystal_I2C lcd(0x27, 16, 2);

PZEM004Tv30 pzem(D11 , D12); // Software Serial pin 11 (RX) & 12 (TX)

float total;
float kwh;

void setup() {
  Serial.begin(115200);
  lcd.begin();
  lcd.backlight();
  WiFi.begin(ssid, password);
  Serial.println("Connecting");
  while(WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
    lcd.setCursor(0,0);
    lcd.print("Connecting.....");
  }
  Serial.println("");
  Serial.print("Connected to WiFi network with IP Address:
  "); Serial.println(WiFi.localIP());

  Serial.println("Timer set to 1 seconds (timerDelay variable), it will take 5
seconds before publishing the first reading.");
```

```
}

void loop() {

  // Send an HTTP POST request depending on timerDelay
  if ((millis() - lastTime) > timerDelay) {
    //Check WiFi connection status
    if(WiFi.status()== WL_CONNECTED){
      WiFiClient  client;
      HTTPClient http;

      int voltage = pzem.voltage();
       Serial.print("Voltage: ");
       Serial.print(voltage);
       Serial.println(" V");
       lcd.setCursor(0,0);
       lcd.print(voltage);
       lcd.print(".V       ");

   float current = pzem.current();
       Serial.print("Current: ");
       Serial.print(current);
       Serial.println(" A");
//       lcd.setCursor(8,0);
//       lcd.print(current);
//       lcd.print(".Amp");


     float power = pzem.power();
         Serial.print("Power: ");
         Serial.print(power);
         Serial.println(" W");
         total = (power / 3600) + total;
         kwh = total / 1000;
         Serial.print(kwh, 7);
         Serial.println(" kWh");
         int watt = power;
         lcd.setCursor(0,1);
         lcd.print(watt );
         lcd.print(".W ");
         lcd.setCursor(9,1);
         lcd.print(kwh,7);
         lcd.print(" .kwh");


     Serial.println();
```

```
    String serverPath = serverGetData + "?id=1&data="+ kwh; // id Pengontrak

    // Your Domain name with URL path or IP address with path
    http.begin(client, serverPath.c_str());
    http.addHeader("Content-Type", "text/html");
    // Send HTTP GET request
    int httpResponseCode = http.GET();

    if (httpResponseCode>0) {
      Serial.println("HTTP Response code: ");
      lcd.setCursor(12,0);
      lcd.print("Send");
      //Serial.println(httpResponseCode);
      String payload = http.getString();
      //Serial.println(payload);
    }
    else {
      Serial.println("Error code: ");
      lcd.setCursor(12,0);
      lcd.print("Fail");
      //Serial.println(httpResponseCode);
    }
    // Free resources
    http.end();
   }
   else {
    Serial.println("WiFi Disconnected");
   }
   lastTime = millis();
  }

}
```