

PERANCANGAN WEB MONITORING TEGANGAN UJUNG FEEDER BERBASIS ARDUINO

M. Adam Syifaul Huda ^{*}), Abdul Syakur, dan Yosua Alvin Adi Soetrisno

Departemen Teknik Elektro, Fakultas Teknik, Universitas Diponegoro
Jl. Prof. Sudharto, SH, Kampus UNDIP Tembalang, Semarang 50275, Indonesia

^{*}E-mail: madamsyifaulhuda@gmail.com

Abstrak

Hingga saat ini pengukuran tegangan ujung Feeder dilakukan petugas PLN secara manual dengan data di waktu tertentu saja, sehingga pengukuran tidak berjalan dengan efektif karena memerlukan banyak waktu dan biaya yang tidak murah. Pada penelitian ini dibuat suatu sistem untuk mempermudah proses pengukuran dan pengawasan tegangan pada ujung feeder menggunakan konsep berbasis IoT, yang dapat menampilkan data secara real-time sehingga memudahkan petugas PLN dalam mengawasi tegangan yang ada di ujung feeder dan dapat secara terus menerus menilai kualitas tenaga listrik yang disalurkan. Prototype alat ukur tegangan ujung feeder berbasis Web dengan output berupa tegangan pada ujung feeder yang terhubung dengan modul ESP8266-01, data dari modul ESP8266-01 tersebut akan dikirimkan ke ThingSpeak dan ditampilkan di Web yang nantinya dapat diakses melalui internet.

Kata kunci : IoT, Web, ESP8266-01

Abstract

Until now, PLN officers have performed the feeder tip voltage measurement manually with data at a certain time, so the measurement does not run effectively because it requires a lot of time and costs are not cheap. In this study, a system was created to simplify the process of measuring and monitoring the voltage at the end of the feeder using an IOT-based concept, which can display real-time data so that it makes it easier for PLN officers to monitor the voltage at the end of the feeder and can continuously assess the quality of electric power, which is funneled. The prototype of a Web-based feeder tip voltage measuring device with an output in the form of a voltage at the end of the feeder connected to the ESP8266-01 module, data from the ESP8266-01 module will be sent to ThingSpeak and displayed on the Web which can later be accessed via the internet.

Keywords: IoT, Web, ESP8266-01

1. Pendahuluan

Internet of Things adalah suatu konsep yang bertujuan untuk memanfaatkan teknologi internet yang terus berkembang agar dapat diimplementasikan ke dalam benda fisik sehingga manusia dapat berinteraksi langsung dengan benda tersebut seperti mengirim data dan melakukan kendali jarak jauh secara real-time.[1]

Untuk menjaga agar pelayanan pada pelanggan makin prima maka dibutuhkan monitoring secara real time agar fluktuasi tegangan dapat diketahui. Selama ini di PT.PLN (Persero) melakukan pengukuran beban secara manual per jangka waktu tertentu, sehingga rawan akan tidak terpantaunya saat terjadi tegangan jatuh. Hal ini bisa diakibatkan karena tidak termonitornya kualitas tegangan pada ujung feeder secara real time.[2]

Pemantauan kualitas tegangan masih dilakukan secara manual atau tidak secara real time, hasil pemantauan tidak

terekam setiap saat sehingga hasil fluktuasi tegangan pada ujung feeder tidak terlihat. Hal ini melatar belakangi penulis untuk membuat suatu perangkat atau alat yang mampu menginformasikan kondisi tegangan ujung feeder secara real time, dan pengamat hanya memantau kondisi tegangan ujung feeder dari komputer secara real time. Adapun tujuan penelitian ini adalah agar dapat mengidentifikasi besar tegangan jatuh terkait pengukuran tegangan ujung feeder dan agar dapat merancang alat monitoring tegangan ujung feeder secara real time. [3]

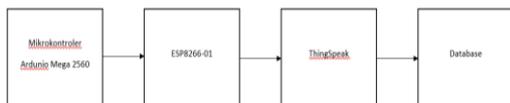
Penelitian kali ini akan merancang sebuah purwarupa sistem monitoring dan komunikasi data berbasis web pada tegangan ujung. Pembacaan dan pengiriman data sensor ke cloud database serta pengolahan data dari cloud database akan dilakukan oleh mikrokontroler. Komunikasi antara mikrokontroler dengan website serta cloud database dilakukan melalui koneksi internet dengan jaringan Wi-Fi. [4]

Rancangan alat ini dibuat untuk diaplikasikan untuk user guna memonitoring tegangan pada ujung *feeder* tanpa harus ke lokasi, pada Sistem diharapkan dapat membantu meminimalkan biaya dari pengukuran tegangan jatuh. Prinsip kerjanya yaitu Tegangan masuk dibaca oleh sensor tegangan AC lalu data diolah di Arduino Mega 2560, setelah itu dikirimkan ke modul ESP8266-01 yang berfungsi sebagai adaptor WiFi akses internet nirkabel. Setelah itu data pembacaan dikirimkan ke *ThingSpeak* sebagai database lalu akan ditampilkan di *Website* secara *real time*. [5] Dari hal-hal yang telah dijelaskan, maka penulis telah memilih judul “**PERANCANGAN WEB MONITORING TEGANGAN UJUNG FEEDER BERBASIS ARDUINO**”.

2. Metode

2.1. Perancangan Monitoring Data dari Mikrokontroler Arduino Mega 2560 berbasis web

Perancangan sistem monitoring data berbasis web memungkinkan adanya pengendalian dan pemantauan tegangan jatuh pada ujung *feeder* secara *online*, dengan memanfaatkan halaman *web* sehingga proses monitoring pada ujung *feeder* dapat dilakukan secara langsung dan fleksibel dari manapun selama terhubung dengan koneksi internet. Pemantauan keadaan ujung *feeder* dilakukan dengan menggunakan 1 sensor yang terpasang pada mikrokontroler Arduino Mega 2560, sedangkan pengiriman data ke *database* dilakukan dengan menggunakan mikrokontroler ESP8266-01 sebagai modul komunikasi data. Mikrokontroler ESP8266-01 juga berperan untuk menghubungkan ke internet lalu diteruskan ke server. [6] Perancangan sistem keseluruhan ditunjukkan pada Gambar 1.



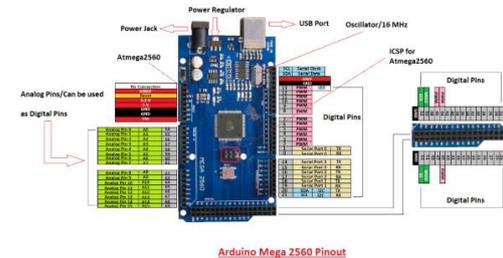
Gambar 1. Diagram Sistem Monitoring Data dari Mikrokontroler Arduino 2560 berbasis web

Gambar 1 menunjukkan Sistem Monitoring Gardu Distribusi Jaringan Tegangan Rendah dengan Arduino Mega 2560 berbasis *Website*. Dari ujung *feeder* data dibaca oleh sensor tegangan, kemudian data tersebut diolah oleh arduino untuk di rekap menjadi satu, kemudian modul ESP8266-01 mencari koneksi internet, jika sudah terkoneksi dengan internet maka data tersebut akan dikirimkan menuju *web server Thingspeak* dan nantinya data yg ada pada *ThingSpeak* akan ditampilkan di *Website*.

2.2 Perancangan Hardware

Mikrokontroler yang digunakan pada perancangan sistem komunikasi data pada tugas akhir ini terdiri dari dua

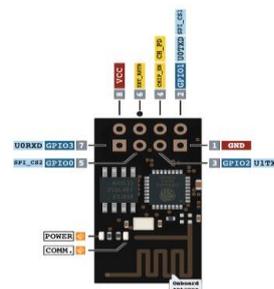
mikrokontroler, yaitu Arduino Mega 2560 dan ESP8266-01. Fungsi dari mikrokontroler Arduino Mega disini adalah sebagai pengolah data pada tugas akhir ini. [7] Bentuk fisik dan *pinout* dari Arduino Mega ditunjukkan pada Gambar 2.



Gambar 2 Pinout Arduino Mega 2560

Mikrokontroler ini menggunakan *chip* Atmel ATmega 328 yang beroperasi pada *clock* 16MHz dengan tegangan operasi 5 volt dengan tegangan masukan yang direkomendasikan sebesar 7-12 volt. Arduino Mega 2560 memiliki 14 pin digital dengan 6 pin diantaranya mampu berfungsi sebagai PWM *output*, selain itu, Arduino Mega 2560 juga memiliki 8 pin analog, arus DC untuk tiap pin I/O dibatasi sebesar 40mA. *Flash Memory* yang dimiliki Arduino Mega 2560 yang digunakan disini sebesar 32KB dengan SRAM 2KB dan 1 KB EEPROM. [8]

Mikrokontroler ESP8266-01 yang digunakan adalah tipe *development board* ESP8266-01 V1.0 yang mengintegrasikan *chip* ESP8266-01 dengan komponen pendukung seperti *voltage regulator* dan *USB to TTL converter* seperti CH340G. *Pinout* dari ESP8266-01 [9] dapat dilihat pada Gambar 3.



Gambar 3 Konfigurasi Pinout ESP8266-01

Tabel 1. Wiring antara Arduino Mega 2560 dengan ESP8266-01

Pin Arduino Mega 2560	Pin ESP8266-01
GND	GND
3.3 V	VCC
3.3 V	CH_PD
RX2	RX
TX2	TX

Development board ESP8266-01 menggunakan tegangan kerja sebesar 3.3V, sebab itu, maka dibutuhkan 3.3v – 5v *bidirectional level shifter* untuk menjembatani komunikasi antara ESP8266-01 dengan Arduino Mega 2560. *Wiring* antara Arduino Mega 2560 dengan ESP8266-01 melalui *bidirectional level shifter*, yang akan dijelaskan dengan [10] Tabel 1.

Sambungan GND pada tabel *wiring* diatas berfungsi sebagai *common ground* antar mikrokontroler sehingga tingkat logika (*logic level*) yang digunakan berada pada tingkatan yang sama antara ESP8266-01 dengan Arduino Mega 2560. Sambungan antara pin 3.3V dari Arduino Mega 2560 ke pin VCC dan CH_PD pada ESP8266 digunakan untuk mengaktifkan transistor, Hubungan ke 3 yaitu dari pin RX2 Arduino Mega 2560 ke pin RX yang terhubung ke ESP8266-01, kemudian untuk pin TX2 dari Arduino Mega 2560 terhubung pada pin TX yang terhubung pada ESP8266-01.

2.3. Perancangan Software Arduino Mega 2560

Perancangan perangkat lunak untuk mikrokontroler Arduino Mega 2560 mencakup inialisasi *library* serta pengiriman data ke mikrokontroler ESP8266-01. *Library* yang digunakan dalam *software* untuk mengirim data ke ESP8266-01 adalah *library SoftwareSerial*. *Pengiriman Data ke ESP8266-01 merupakan data pemicu atau trigger untuk pembacaan sensor pada ESP8266-01, data trigger ini akan dikirimkan ketika hasil pembacaan RTC pada Arduino Mega 2560 sesuai dengan waktu pengiriman data yang diatur melalui Thingspeak*. [11]

2.4. Perancangan Software ESP8266-01

Perancangan perangkat lunak atau software untuk modul ESP8266-01 meliputi pembangunan senarai komunikasi dua arah yaitu proses pengiriman data dari mikrokontroler Arduino Mega 2560 ke cloud database serta proses penerimaan data dari cloud database ke mikrokontroler. Pembangunan software dilakukan dengan Bahasa C dengan compiler Arduino IDE.

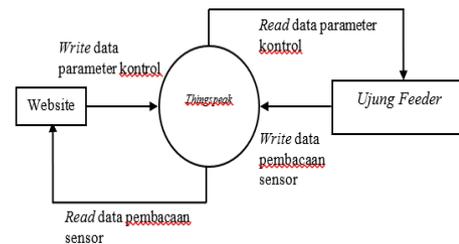
Perancangan perangkat lunak untuk modul ESP8266-01 mencakup inialisasi *library* dan *variable*, serta senarai pengiriman dan pembacaan data dari Arduino Mega 2560.

- Senarai inialisasi *library* dan variabel.
- Inialisasi *library*.
- Deklarasi variabel.

Senarai pembacaan sensor digunakan untuk membaca hasil pengukuran sensor yang terhubung pada mikrokontroler Arduino Mega 2560, dimana data pembacaan sensor tersebut akan digunakan untuk pengawasan atau pemantauan melalui website. [12]

2.5. Perancangan Sistem Database

Perancangan sistem cloud database terdiri dari dua bagian yaitu pembuatan channel dalam Thingspeak dan perancangan script PHP yang akan membantu pemantauan data dari mikrokontroler. Pada penelitian ini digunakan jasa web hosting gratis dengan menggunakan 000webhost, sedangkan domain yang digunakan untuk mengakses website nya adalah teganganjatuh.000webhostapp.com.



Gambar 4. Diagram konteks aliran data pada sistem database

Berdasarkan Gambar 4, sistem database yang dirancang menghubungkan antara Website dengan perangkat Arduino Mega 2560. Dari diagram konteks aliran data tersebut, dapat dilihat bahwa baik dari website maupun perangkat Arduino Mega mengakses database sebagai perantara pengiriman data. [13]

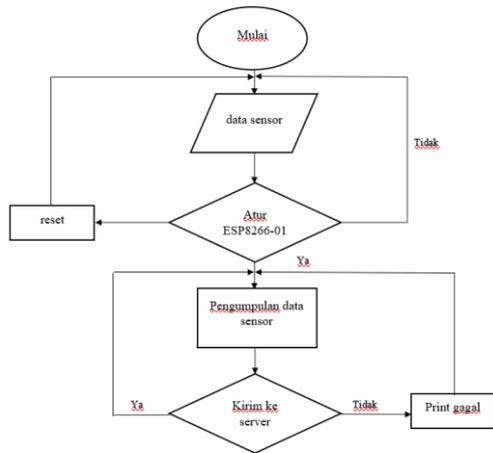
Data dari perangkat Arduino Mega dalam hal ini adalah data pembacaan sensor dikirim ke database kemudian ditampilkan dalam Website sebagai sistem monitoring.

Perancangan Script HTML atau program PHP yang dirancang pada penelitian ini digunakan untuk mengambil data dari cloud database, dan menampilkannya untuk memonitoring tegangan dari ujung feeder. Sebagai bahan informasi untuk user dimana nantinya jika ada kesalahan teknis seperti listrik padam, maka akan ada pemberitahuan pada website berupa animasi yang menunjukkan bahwa sedang terjadi mati listrik. [14]

Script PHP untuk melakukan read data dari cloud database berfungsi untuk menyediakan data dari cloud database ke dalam bentuk chart dan integer sebagai informasi untuk user. [15]

3.5 Perancangan Script Tampilan Website

Perancangan Script untuk tampilan ini menggunakan template dari bootstrap dimana akan membantu dalam pembuatan website agar terlihat dinamis atau tidak kaku. [16]



Gambar 5. Diagram alir pengiriman data menuju Website

Website yang di desain terdiri dari tiga yaitu: sistem notifikasi, sistem data berbentuk grafik dan data real time.

3. Hasil dan Analisis

3.1. Pengujian Komunikasi Data Arduino Mega 2560 dengan ESP8266-01

Pengujian komunikasi data antara Arduino Mega 2560 dengan ESP8266-01 pada penelitian ini terdiri dari satu bagian saja yaitu pengiriman data sensor dari Arduino Mega 2560 dan ditampilkan melalui serial monitor ESP8266-01. Gambar 10 menunjukkan data yang dikirim ke Arduino Mega 2560 dapat dilihat melalui serial monitor ESP8266-01.

```

Voltage: 1026.09831. at command => AT+CIPMUX=1 OK
832. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OK
833. at command => AT+CIPSEND=0,55 OK
834. at command => AT+CIPCLOSE=0 OK
Voltage: 1047.36835. at command => AT+CIPMUX=1 OK
836. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OK
837. at command => AT+CIPSEND=0,55 OK
838. at command => AT+CIPCLOSE=0 OK
  
```

Gambar 6. Tampilan serial monitor pengiriman data dari Arduino Mega 2560 ke ESP8266-01

Pada gambar 10 diatas menunjukkan format data yang diterima ESP8266-01 dari sensor tegangan, data tersebut merupakan hasil komunikasi data serial yang dilakukan oleh Arduino mega 2560 dengan ESP8266-01, ditunjukkan dengan kata "OK" yang artinya koneksi berhasil dilakukan dan kabel/jumper terhubung dengan baik dan benar, kemudian data tersebut nantinya akan dikirim menuju ThingSpeak oleh modul ESP8266-01.

Pengujian komunikasi data antara Arduino Mega 2560 dengan ESP8266-01 pengiriman data sensor dari Arduino Mega 2560 dan ditampilkan melalui serial monitor ESP8266-01. Gambar 11 menunjukkan data yang dikirim

ke Arduino Mega 2560 dapat dilihat melalui serial monitor ESP8266-01.

```

COM5
0. at command => AT Fail
0. at command => AT+CWMODE=1 Fail
0. at command => AT+CWJAP="ihik","98989898" Fail
Voltage: 151.830. at command => AT+CIPMUX=1 Fail
0. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 Fail
0. at command => AT+CIPSEND=0,54
  
```

Gambar 11. Tampilan serial monitor pengiriman data dari Arduino Mega 2560 ke ESP8266-01

Pada gambar 11 diatas menunjukkan format data yang diterima ESP8266-01 dari sensor tegangan, data tersebut merupakan hasil komunikasi data serial yang dilakukan oleh Arduino mega 2560 dengan ESP8266-01, ditunjukkan dengan kata "Fail" yang artinya koneksi gagal dilakukan ini dikarenakan kabel/jumper tidak terhubung dengan baik dan benar, kemudian data tersebut nantinya akan dikirim menuju ThingSpeak oleh modul ESP8266-01.

3.2. Pengujian Koneksi Modul ESP8266-01 dengan Internet

Pengujian dilakukan dengan mencoba membuat koneksi melalui MiFi sebagai hotspot yang sesuai dengan pengaturan pada program yang telah dibuat untuk mikrokontroler ESP8266-01 dengan menggunakan Arduino IDE. Proses pembangunan koneksi ke internet dapat dipantau melalui serial monitor Arduino IDE yang akan menampilkan informasi koneksi internet apabila sambungan berhasil dibangun. Berikut Gambar 12 menunjukkan respon serial monitor ketika koneksi berhasil dibuat.

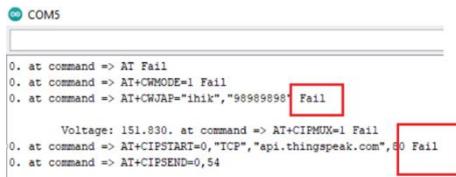
```

COM5
and => AT+CIPSEND=0,54 OK
688. at command => AT+CIPCLOSE=0 OK
0. at command => AT OK
1. at command => AT+CWMODE=1 OK
2. at command => AT+CWJAP="ihik","98989898" OK
  
```

Gambar 12. Tampilan serial monitor ketika koneksi berhasil tersambung

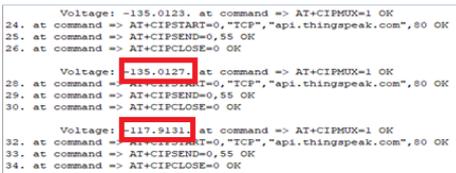
Pada gambar 12 menunjukkan bahwa koneksi yang dibuat oleh Mifi sebagai hotspot berhasil terhubung dengan modul ESP8266-01. Respon Serial Monitor yang menunjukkan koneksi berhasil dilakukan yaitu dengan kata "OK".

Respon yang menunjukkan gagalnya proses pembuatan koneksi ke internet sehingga perangkat tidak dapat mengirim dan menerima data melalui internet. Apabila koneksi gagal dibuat, maka tampilan pada serial monitor akan menampilkan proses percobaan pembuatan koneksi, yang ditampilkan seperti pada Gambar 13berikut.



Gambar 13. Tampilan serial monitor ketika pembangunan koneksi tidak tersambung

Pada gambar 13 menunjukkan bahwa koneksi yang dibuat oleh Mifi gagal terhubung dengan modul ESP8266-01. Respon Serial Monitor yang menunjukkan koneksi berhasil dilakukan yaitu dengan kata “Fail”, hal ini bisa karena jaringan internet sedang tidak stabil atau buruk. Respon yang menunjukkan koneksi dalam keadaan tidak stabil antara modul ESP8266-01 ke internet sehingga perangkat tidak dapat mengirim dan menerima data melalui internet. Apabila koneksi dalam keadaan tidak stabil, maka tampilan pada serial monitor seperti pada Gambar 14.



Gambar 14. Tampilan serial monitor ketika koneksi terputus saat pembacaan sedang dilakukan

Pada gambar 14 Respon yang menunjukkan koneksi pada saat tidak stabil, dan tegangan yang terbaca akan menjadi minus. Hal ini terjadi karena koneksi jaringan internet yang buruk.

3.3. Pengujian Komunikasi Data ESP8266-01 dengan ThingSpeak

Pengiriman data dari ESP8266-01 dilakukan setelah port komunikasi serial dari ESP8266-01 menerima data. Data hasil pembacaan akan dikirimkan ke ThingSpeak dan akan memberikan tampilan seperti pada Gambar 15.

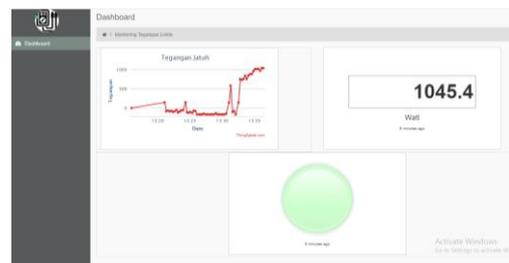


Gambar 15. Tampilan ThingSpeak ketika ESP8266-01 berhasil menambahkan data

Pada gambar 15 merupakan hasil pembacaan tegangan dari sensor tegangan yang di teruskan ke ThingSpeak oleh modul ESP8266-01. Data tersebut masuk ke dalam database Thingspeak yang di sebut chanel kemudian terpecah lagi menjadi beberapa field yang menunjukkan data grafik, digital dan indikator lampu.

3.4. Pengujian Komunikasi Data ThingSpeak dengan Website

Pengujian komunikasi data dari ThingSpeak akan ditampilkan di website. Hasil pembacaan akan diolah menggunakan script PHP dan akan ditampilkan pada alamat *teganganjatuh.000webhostapp.com* data yang diolah di tunjukkan seperti pada Gambar 16.



Gambar 16. Tampilan halaman website ketika script berhasil memanggil data dari ThingSpeak

Pada Gambar 16 menunjukkan bahwa data dari ThingSpeak berhasil diambil dan ditampilkan ke dalam website untuk proses monitoring. Tampilan Website yang seperti gambar diatas sudah mencakup informasi yang dibutuhkan dalam tugas akhir ini. Sehingga hanya memerlukan satu halaman saja.

3.5. Pengujian Waktu Penerimaan data

Pengujian ini dilakukan untuk mengetahui cepat lambatnya respon pengiriman data dari ESP8266-01 ke ThingSpeak dengan membandingkan waktu ketika ESP8266-01 mengirim data dengan waktu ketika ThingSpeak menerima data. Data waktu dari hasil pengujian ditunjukkan pada Tabel; 2

Tabel 2. Pengujian Waktu komunikasi data ESP8266-01 dengan ThingSpeak

Pengujian	Tegangan	Mengirim Data Melalui ESP8266-01	Waktu Data Diterima Oleh ThingSpeak	Delay
1	3836	06:40:37	06:40:52	15 detik
2	3781	06:40:52	06:41:07	15 detik
3	3912	06:41:07	06:41:22	15 detik
4	3922	06:41:22	06:41:38	16 detik
5	3804	06:41:38	06:41:53	15 detik
6	3913	06:41:53	06:42:08	15 detik
7	4191	06:42:08	06:42:23	15 detik
8	4641	06:42:23	06:42:38	15 detik
9	4699	06:42:38	06:42:54	16 detik
10	4717	06:42:54	06:43:10	16 detik

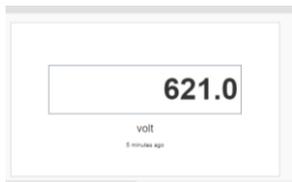
Pada pengujian waktu penerimaan data didapatkan bahwa perubahan data pada *server* telah sesuai dengan serial monitor dengan *delay* 15 – 16 detik, dikarenakan dalam proses mengkoneksikannya ke *ThingSpeak* membutuhkan waktu dan dalam pembungkusan data menjadi satu dalam satu paket data untuk dikirimkan lewat modul ESP8266-01 membutuhkan waktu beberapa detik.

3.6. Pengujian Monitoring Data

Pengujian ini dilakukan untuk mengetahui bahwa sistem monitoring dapat berjalan dengan benar. Di buktikan dengan tampilan yang menunjukkan hasil sesuai dengan referensinya.

```
Voltage: 622.0531. at command => AT+CIPMUX=1 OK
32. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OK
33. at command => AT+CIPSEND=0,54 OK
34. at command => AT+CIPCLOSE=0 OK
```

(a)



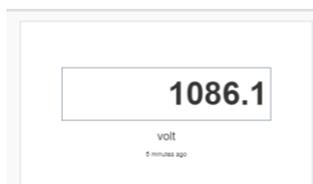
(b)

Gambar 17. Perbandingan tegangan pada 0.6 KV a) Pada serial Monitor, b) Pada Website

Pada gambar 17 menunjukkan bahwa terdapat perbedaan tegangan antara serial monitor dan yang di tampilkan di website sebesar 1,05 volt pada referensi tegangan 0,6 KV.

```
Voltage: 1083.121656. at command => AT+CIPMUX=1 OK
1657. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OK
1658. at command => AT+CIPSEND=0,55 OK
1659. at command => AT+CIPCLOSE=0 OK
```

(a)



(b)

Gambar 18. Perbandingan tegangan pada 1 KV a) Pada serial Monitor, b) Pada Website

Pada gambar 18 menunjukkan bahwa terdapat perbedaan tegangan antara serial monitor dan yang di tampilkan di website sebesar 2,9 volt pada referensi tegangan 1 KV.

```
Voltage: 1577.522044. at command => AT+CIPMUX=1 OK
2045. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OK
2046. at command => AT+CIPSEND=0,55 OK
2047. at command => AT+CIPCLOSE=0 OK
```

(a)



(b)

Gambar 19. Perbandingan tegangan pada 1.5 KV a) Pada serial Monitor, b) Pada Website

Pada gambar 19 menunjukkan bahwa terdapat perbedaan tegangan antara serial monitor dan yang di tampilkan di website sebesar 2,3 volt pada referensi tegangan 1,5 KV.

```
Voltage: 1890.15371. at command => AT+CIPMUX=1 OK
372. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OK
373. at command => AT+CIPSEND=0,55 OK
374. at command => AT+CIPCLOSE=0 OK
```

(a)



(b)

Gambar 20. Perbandingan tegangan pada 1,8 KV a) Pada serial Monitor, b) Pada Website

Pada gambar 20 menunjukkan bahwa terdapat perbedaan tegangan antara serial monitor dan yang di tampilkan di website sebesar 0,1 volt pada referensi tegangan 1,8 KV.

```
Voltage: 2011.682272. at command => AT+CIPMUX=1 OK
2273. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OK
2274. at command => AT+CIPSEND=0,55 OK
2275. at command => AT+CIPCLOSE=0 OK
```

(a)

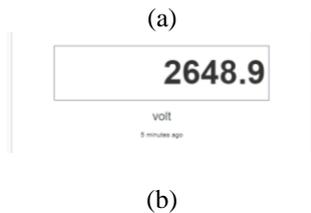


(b)

Gambar 21. Perbandingan tegangan pada 2 KV a) Pada serial Monitor, b) Pada Website

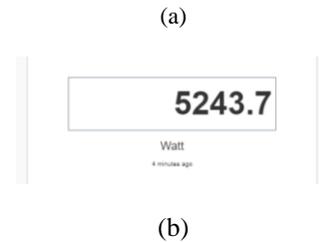
Pada gambar 21 menunjukkan bahwa terdapat perbedaan tegangan antara serial monitor dan yang di tampilkan di website sebesar 7,9 volt pada referensi tegangan 2 KV.

```
Voltage: 2639.9115. at command => AT+CIPMUX=1 OK
16. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OK
17. at command => AT+CIPSEND=0,55 OK
18. at command => AT+CIPCLOSE=0 OK
```



Pada gambar 24 menunjukkan bahwa terdapat perbedaan tegangan antara serial monitor dan yang di tampilkan di website sebesar 9,4 volt pada referensi tegangan 4,6 KV.

```
Voltage: 5285.511042. at command => AT+CIPMUX=1 OK
1043. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OK
1044. at command => AT+CIPSEND=0,55 OK
1045. at command => AT+CIPCLOSE=0 OK
```



Gambar 22. Perbandingan tegangan pada 2,6 KV a) Pada serial Monitor, b) Pada Website

Gambar 25. Perbandingan tegangan pada 5,2 KV s) Pada serial Monitor, b) Pada Website

Pada gambar 22 menunjukkan bahwa terdapat perbedaan tegangan antara serial monitor dan yang di tampilkan di website sebesar 9 volt pada referensi tegangan 2,6 KV.

Pada gambar 25 menunjukkan bahwa terdapat perbedaan tegangan antara serial monitor dan yang di tampilkan di website sebesar 41,8 volt pada referensi tegangan 5,2 KV.

```
Voltage: 3887.79534. at command => AT+CIPMUX=1 OK
535. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OK
536. at command => AT+CIPSEND=0,55 OK
537. at command => AT+CIPCLOSE=0 OK
```



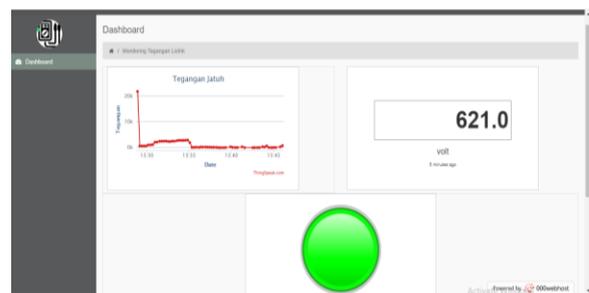
3.7. Pengujian Sistem Indikator

Sistem Indikator yang dirancang berfungsi untuk mengidentifikasi ketika terjadi tegangan jatuh. Indikator ini menunjukkan pada lampu indikator warna hijau yang akan menyala terang ketika tegangan berada dibawah 1 KV yang berarti ini teridentifikasi tegangan jatuh, sedangkan lampu indikator warna hijau akan redup ketika terjadi tegangan berada diatas 1 KV. Gambar 4.16 menunjukkan ketika terjadi tegangan jatuh.

Gambar 23. Perbandingan tegangan pada 3,9 KV a) Pada serial Monitor, b) Pada Website

Pada gambar 23 menunjukkan bahwa terdapat perbedaan tegangan antara serial monitor dan yang di tampilkan di website sebesar 25,2 volt pada referensi tegangan 3,9 KV.

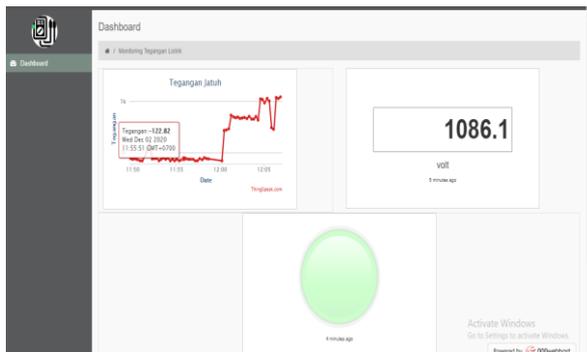
```
Voltage: 4632.54858. at command => AT+CIPMUX=1 OK
859. at command => AT+CIPSTART=0,"TCP","api.thingspeak.com",80 OK
860. at command => AT+CIPSEND=0,55 OK
861. at command => AT+CIPCLOSE=0 OK
```



Gambar 26. Tampilan Ketika 0,6 KV

Pada gambar 26 menunjukkan tampilan grafik, tampilan digital dan tampilan Indikator ketika tegangan berada pada 0,6 KV, dengan ini tampilan halaman website di atas teridentifikasi tegangan jatuh dan lampu warna hijau menyala terang,

Gambar 24. Perbandingan tegangan pada 4,6 KV a) Pada serial Monitor, b) Pada Website



Gambar 27 Tampilan Ketika 1 KV

Pada gambar 27 menunjukkan tampilan grafik, tampilan digital dan tampilan Indikator ketika tegangan berada pada 1 KV. Tampilan Indikator tegangan di atas 1KV menunjukkan lampu warna hijau akan meredup.

4. Kesimpulan

Berdasarkan hasil pengujian dan analisis Penelitian ini dapat disimpulkan bahwa sistem pengiriman data Arduino Mega 2560 menuju modul ESP8266-01 telah berhasil dirancang yang ditunjukkan oleh terbacanya data tegangan dari sensor oleh serial monitor. Sistem komunikasi antara ESP8266-01 dengan *ThingSpeak* yang telah dibuat akan menunjukkan balasan "OK" pada serial monitor apabila telah berhasil terhubung, apabila komunikasi gagal terhubung maka akan menunjukkan balasan "FAIL" sedangkan apabila koneksi yang telah terhubung dan kemudian terputus secara tiba-tiba maka akan menunjukkan pembacaan nilai bernilai negatif. Pengujian respon waktu antara ESP8266-01 dengan *ThingSpeak* terdapat delay antara 15-16 detik. Pengujian sistem monitoring telah berhasil dilakukan melalui *website* dari hasil pembacaan *ThingSpeak* yang ditunjukkan dengan samanya nilai antara serial monitor dengan data digital pada *website*. Pengujian sistem indikator, sistem akan menunjukkan lampu indikator berwarna hijau menyala terang ketika tegangan yang terbaca di bawah 1 KV, sebaliknya jika indikator lampu hijau menyala redup maka tegangan yang terbaca menunjukkan di atas 1KV.

Referensi

[1]. Khan, Minhaj Ahmad, and Khaled Salah. "IoT security: Review, blockchain solutions, and open challenges." *Future Generation Computer Systems* 82 (2018): 395-411.

[2]. Listrik, Kualitas Tenaga. "Evaluasi Keandalan sistem Distribusi Tenaga Listrik Berdasarkan SAIDI dan SAIFI." (2011)..

[3]. Tanjung, Abrar. "Rekonfigurasi Sistem Distribusi 20 KV Gardu Induk Teluk Lembu dan PLTMG Langgam Power Untuk Mengurangi Rugi Daya dan Drop Tegangan." *Jurnal Sains dan Teknologi Industri* 11.2 (2015): 160-166.

[4]. Shrader, Theodore Jack London, et al. "Web server account manager plug-in for monitoring resources." U.S. Patent No. 6,026,440. 15 Feb. 2000.

[5]. Schwartz, Marco. *Internet of Things with ESP8266*. Packt Publishing Ltd, 2016.

[6]. Sulistyowati, Riny, and Dedi Dwi Febiantoro. "Perancangan prototype sistem kontrol dan monitoring pembatas daya listrik berbasis mikrokontroler." *Jurnal IPTEK* 16.1 (2012).

[7]. Roihan, Ahmad, Angga Permana, and Desy Mila. "Monitoring kebocoran gas menggunakan mikrokontroler arduino uno dan esp8266 berbasis internet of things." *ICIT (Innovative Creative and Information Technology)* 2.2 (2016): 170-183.

[8]. Ramadhan, Ade Surya, and L. Budi Handoko. "Rancang bangun sistem keamanan rumah berbasis arduino mega 2560." *Techno. com* 15.2 (2016): 117-124.

[9]. Shobrina, Upik Jamil, Rakhmadhany Primananda, and Rizal Maulana. "Analisis Kinerja Pengiriman Data Modul Transceiver NRF24I01, Xbee dan Wifi ESP8266 Pada Wireless Sensor Network." *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer e-ISSN 2548* (2018): 964X.

[10]. Sachio, Steven, Agustinus Noertjahyana, and Resmana Lim. "Prototype penggunaan IoT untuk monitoring level pada penampung air berbasis ESP8266." *Jurnal Infra* 5.2 (2017): 37-42.

[11]. Maureira, Marcello A. Gómez, Daan Oldenhof, and Livia Teernstra. "ThingSpeak—an API and Web Service for the Internet of Things." *World Wide Web* (2011).

[12]. Limantara, Arthur Daniel, Yosef Cahyo Setianto Purnomo, and Sri Wiwoho Mudjanarko. "Pemodelan Sistem Pelacakan LOT Parkir Kosong Berbasis Sensor Ultrasonic Dan Internet Of Things (IOT) Pada Lahan Parkir Diluar Jalan." *Prosiding Semnastek* (2017).

[13]. Yulita, Resi Risma, and Juli Sardi. "Perancangan Database Pengukuran Detak Jantung menggunakan Web Thingspeak." *JTEV (Jurnal Teknik Elektro dan Vokasional)* 6.2 (2020): 449-459.

[14]. Fuady, Wildan, and Lilis Diana. "Analisis dan Perancangan Aplikasi Administrasi Keuangan STMIK æAMIKBANDUNGæ Berbasis Web dengan Menggunakan HTML Bootstrap." *KOPERTIP: Jurnal Ilmiah Manajemen Informatika dan Komputer* 1.03 (2017): 113-116.

[15]. Budi, Tri Agung Setya. *LKP: Monitoring Suhu dan Kelembapan Ruang Server PT. Sier Surabaya Berbasis Web*. Diss. Institut Bisnis dan Informatika Stikom Surabaya, 2018.