

SISTEM MONITORING ENERGI LISTRIK PADA SMART ENERGY METER MENGUNAKAN APLIKASI BLYNK BERBASIS ANDROID

MONITORING SYSTEM ELECTRICAL ENERGY OF SMART ENERGY METER USING ANDROID BASED BLYNK APPLICATION

Vika Laeli Rismawati¹, Hurianti Vidyningtyas², Trasma Yunita³

^{1,2,3}Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom

¹vikalaeli@student.telkomuniversity.ac.id, ²huriantividya@telkomuniversity.ac.id,

³trasmayunita@telkomuniversity.ac.id

Abstrak

Dalam masalah penggunaan listrik, seringkali pengguna menggunakan perangkat listrik dengan daya yang cukup besar. Untuk menghemat penggunaan listrik, harus dilakukan pemanfaatan sumber daya listrik dengan kendali. Oleh karena itu perlu adanya sistem yang dapat membantu untuk mengendalikan sumber daya listrik pada perangkat elektronik yang ada di dalam rumah. Pada Tugas Akhir ini, dirancang sebuah sistem menggunakan aplikasi Blynk yang dapat melakukan *monitoring* perangkat elektronik dengan menampilkan daya dan waktu penggunaan perangkat elektronik, mengaktifkan dan menonaktifkan perangkat elektronik, serta mendapatkan notifikasi melalui *smartphone*. Dari penelitian ini, didapatkan nilai rata-rata *delay* yang terjadi pada sistem *connecting* sebesar 23,448 detik. Nilai rata-rata yang terjadi pada *button*, daya dan waktu. Nilai rata-rata *delay* yang terjadi pada *button on* sebesar 2,604 detik, sedangkan *delay* pada *button off* sebesar 3,464 detik. Untuk nilai rata-rata *delay* yang terjadi saat pengiriman daya pada perangkat 1 sebesar 0,443 detik, perangkat 2 sebesar 0,511 detik, dan perangkat 3 sebesar 0,691 detik. Sedangkan untuk nilai rata-rata *delay* yang terjadi saat pengiriman waktu pada perangkat 1 sebesar 6,7303 detik, perangkat 2 sebesar 6,797 detik, dan perangkat 3 sebesar 6,659 detik. Sistem juga dapat menampilkan notifikasi bila daya mencapai batas maksimal.

Kata kunci : Daya listrik, perangkat elektronik, *smartphone*, aplikasi Blynk.

Abstract

It matters to use electrical, frequently used device electrical with considerable power. For saving use electrical, must doing utilization electrical power source with controlling. Therefore, there is a necessary system for helping to control and monitoring electrical power sources in the home. In this Final Project, a system uses the Blynk application that can monitor electronic devices by displaying power at the load used, displaying usage time, activating and deactivating electronic devices, and getting notification via smartphone. From this research, smart energy meter devices can be controlled by the Blynk application. Obtained the average value of delay that occurs on the button, power, and time. The average delay value that occurs on button on is 2.604 seconds, while the average value of delay that occurs on button off is 3.464 seconds. The average delay value when sending power to device 1 is 0.443 seconds, device 2 is 0.511 seconds, and device 3 is 0.691 seconds. As for the average value of delay that occurs when sending time on device 1 is 6.7303 seconds, device 2 is 6.797 seconds, and device 3 is 6.659 seconds. The system can also display notification when the power reaches the maximum limit.

Keywords: Power electrical, electronic devices, *smartphone*, Blynk application.

1. Pendahuluan

Konsumsi penggunaan listrik yang dipakai di rumah menjadi hal yang penting untuk diperhatikan, agar tidak terjadi pemborosan energi listrik. Pemilik rumah yang lupa mematikan peralatan listrik dapat menyebabkan penggunaan konsumsi daya sangat besar dan yang paling bahaya adalah terjadinya korsleting listrik. Untuk menanggapi hal tersebut, dibutuhkan sebuah inovasi teknologi untuk mengendalikan dan memantau keadaan listrik yang ada didalam rumah[1].

Pengukuran energi pada perangkat listrik rumah tangga sudah banyak diteliti, salah satunya adalah penelitian[2] yang membahas mengenai sebuah alat untuk membaca tegangan dan arus *direct current* dari baterai dan mengontrol penggunaan sumber daya listrik dari baterai. Hasil dari penelitian tersebut adalah sistem yang memiliki kemampuan untuk mengaktifkan dan menonaktifkan proses penggunaan baterai (*discharging*) atau pengisian (*charging*), *switch* sumber listrik dan pembacaan tegangan dan arus beban yang terintegrasi dengan penelitian lain berbasis jaringan ke *user*[2]. Pada penelitian tersebut tidak dibahas mengenai pembacaan daya beban pada penggunaan perangkat listrik, dan komunikasi data yang terintegrasi jaringan dari sumber atau ke *user*[2].

Dari permasalahan tersebut, maka penulis melakukan penelitian dengan merancang sistem *monitoring* untuk *smart energy meter* berbasis IoT yang dapat membaca daya beban dan dapat mengontrol serta memantau pada penggunaan perangkat listrik melalui *smartphone* yang terintegrasi ke jaringan. Diharapkan dengan adanya alat ini dapat mempermudah dalam mengontrol dan melakukan *monitoring* pada perangkat elektronik yang ada dirumah, sehingga tidak perlu mengontrol secara langsung, hanya dengan mengontrol dan melakukan *monitoring* melalui *smartphone* saja dengan menggunakan aplikasi Blynk. Pentingnya memiliki aplikasi ini, agar kita mudah memonitoring penggunaan listrik jika daya perangkat melebihi batas maksimum yang ditentukan. Hal tersebut dapat meminimalisir terjadinya korsleting listrik, dan penggunaan konsumsi daya listrik yang berlebihan.

2. Dasar Teori dan Perancangan

2.1 Internet of Things (IoT)

Konsep dari Internet of Things (IoT) sebenarnya cukup sederhana yaitu, "*Anything that can be connected, will be connected.*" Ketika sesuatu dapat terhubung ke internet, maka sesuatu tersebut dapat mengirim informasi atau menerima informasi, atau keduanya.

Dalam penerapannya, IoT dapat mengidentifikasi, menemukan, melacak, memantau objek dan memicu *event* secara otomatis dan *real-time*. IoT terdiri dari perangkat pintar berkemampuan web yang menggunakan prosesor tertanam, sensor, dan *hardware* untuk mengumpulkan, mengirim, dan bertindak sesuai data yang diperoleh dari lingkungan mereka. Perangkat IoT berbagi data sensor yang dikumpulkan dengan menghubungkan ke IoT *gateway* atau perangkat lainnya tempat data dikirim ke *cloud* untuk dianalisis[3].

2.2 Smart Energy Meter

Smart energy meter merupakan salah satu contoh *smart devices*. *Smart devices* merupakan perangkat yang dapat dikontrol dan berkoordinasi dengan baik, setelah menerima sinyal dari perangkat maupun dari user. *Smart devices* dapat memfasilitasi pengurangan atau pergeseran konsumsi energi secara keseluruhan dengan membuat rata-rata energi yang dikonsumsi secara teratur dan tepat waktu[1]. *Smart energy meter* bertujuan untuk memberikan informasi energi yang dikonsumsi oleh perangkat listrik, untuk melakukan strategi tarif berdasarkan waktu pemakaian dan kualitas daya[4].

2.3 ESP8266-01

ESP8266-01 atau ESP01 adalah modul Wi-Fi yang dapat mengakses mikrokontroler ke jaringan Wi-Fi. Modul ini berfungsi untuk sebagai titik akses/membuat hotspot dan sebagai stasiun Wi-Fi sehingga dapat mengambil data dengan mudah untuk pengaplikasian Internet of Things[5]. Modul ini juga dapat diprogram dengan Arduino IDE. Modul ESP8266 hanya dapat bekerja dengan tegangan 3.3V. ESP8266-01 terdiri dari 8 pin, yaitu Ground, Tx, CH-EN, RST, VCC, Rx, GPIO-0, dan GPIO-2[6]. Bentuk fisik dari ESP8266-01 dapat dilihat pada **Gambar 2.3**.



Gambar 1. Modul Wi-Fi ESP8266-01

2.4 Smartphone

Smartphone adalah ponsel cerdas yang fungsinya lebih dari sekadar mengirim teks atau pesan dan telepon[7]. *Smartphone* dapat menjelajahi internet dan melakukan program *software* seperti halnya komputer[9]. Layaknya komputer, *smartphone* mempunyai *Operating System* (OS). Beberapa OS *smartphone* yang populer adalah BlackBerry (dibuat oleh Research In Motion), iOS (dibuat oleh Apple), Android (dibuat oleh Google), dan Windows Phone (dibuat oleh Microsoft)[8]

2.5 Android

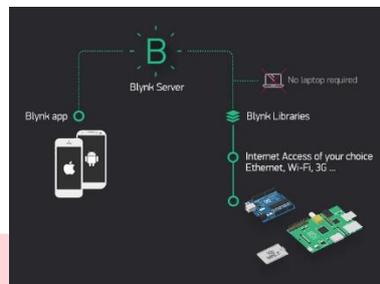
Android adalah sistem operasi seluler yang dikembangkan oleh sebuah perusahaan yang bernama Android Inc, yang merupakan perusahaan pembuat perangkat lunak untuk ponsel yang kemudian dibeli oleh perusahaan Google Inc[9]. Sistem Android bersifat *open source*, yang berarti *source code* tersedia untuk pabrikan dengan Apache License 2.0, yaitu pemegang hak cipta memberikan hak untuk mempelajari, mengubah, dan mendistribusikan perangkat lunak kepada siapa pun dan untuk tujuan apa pun[9] atau menyediakan *platform* terbuka bagi para pengembang untuk membuat aplikasi sendiri[10].

2.6 Aplikasi Blynk

Aplikasi Blynk merupakan sebuah *platform* yang digunakan untuk mengendalikan modul Arduino, ESP8266, Raspberry Pi, dan modul lainnya melalui internet[11]. Layanan server ini memiliki lingkungan *mobile user* baik Android maupun iOS yang mendukung berbagai macam *hardware* yang dapat digunakan untuk *project Internet of Things*. Aplikasi Blynk dapat diunduh melalui Google Play dan Apple Store.

Blynk diciptakan dengan tujuan untuk kontrol dan *monitoring hardware* secara jarak jauh menggunakan komunikasi data internet. Ada tiga komponen utama pada *platform* Blynk, antara lain[11]:

1. Blynk App memungkinkan untuk membuat *interface* pada *project* yang dibuat dengan menggunakan berbagai *widget* yang sudah disediakan[19].
 2. Blynk Server bertanggung jawab untuk komunikasi antara *smartphone* dan *hardware*[11].
- Blynk Libraries untuk membantu pengembangan *code*. Blynk *library* tersedia pada banyak *platform* perangkat keras sehingga dapat memudahkan para pengembang IoT dengan fleksibilitas *hardware* yang didukung oleh Blynk[11]

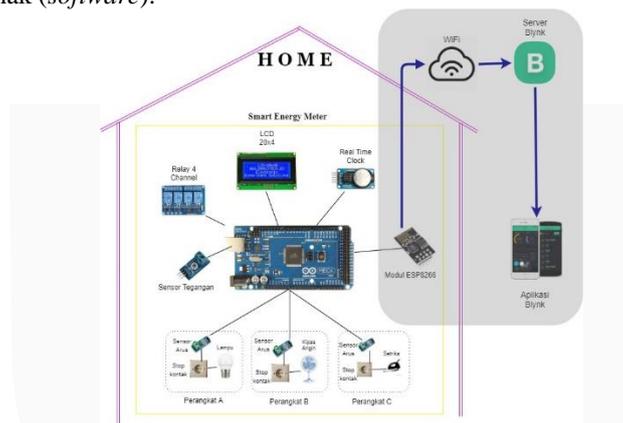


Gambar 2. Blynk cloud

3. Model Sistem dan Perancangan

3.1 Desain Sistem

Pada perancangan sistem ini, terdiri dari 2 bagian, yaitu perancangan perangkat keras (*hardware*) dan perancangan perangkat lunak (*software*).

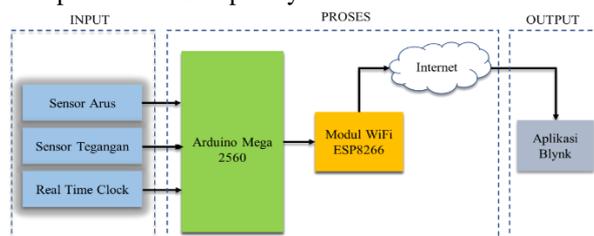


Gambar 3. Desain sistem keseluruhan

Gambar 3. merupakan desain sistem secara keseluruhan. Berdasarkan ilustrasi tersebut, terdapat dua bagian yaitu, bagian pertama adalah perancangan *smart energy meter*. Sistem ini menggunakan sensor arus dan sensor tegangan untuk mendeteksi arus dan tegangan yang dipasang pada sebuah perangkat elektronik di dalam rumah. Bagian kedua adalah perancangan sistem *monitoring* aplikasi Blynk. Sistem ini menggunakan modul WiFi ESP8266-01 sebagai penghubung agar data dari *smart energy meter* dapat ditampilkan ke aplikasi Blynk.

3.1.1 Diagram Blok Sistem

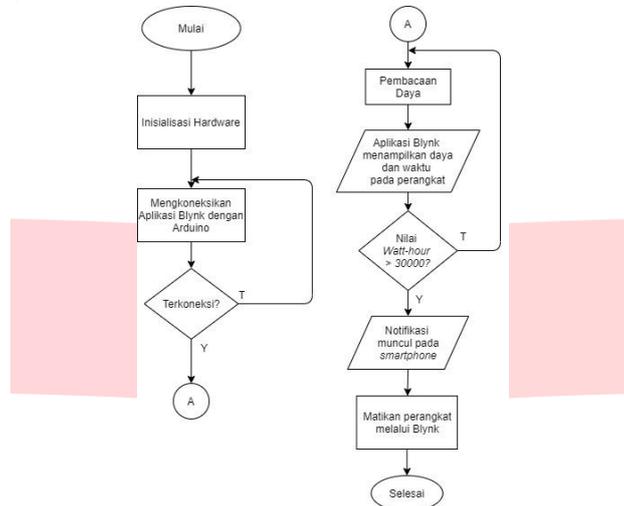
Sistem *input* terdiri dari sensor arus, sensor tegangan, dan RTC yang dihubungkan ke mikrokontroler. Mikrokontroler tersebut berupa Arduino Mega 2560 sebagai pusat kontrol yang memproses dan mengolah perintah dari input. Modul WiFi ESP8266 sebagai penghubung antara mikrokontroler dan aplikasi Blynk untuk menyimpan dan mengambil data dari sensor. *Output* dari sistem ini adalah dapat mengaktifkan dan menonaktifkan *smart energy meter*, serta menampilkan data berupa daya dan waktu.



Gambar 4. Diagram blok sistem

3.1.2 Diagram Alir

Sistem ini menggunakan aplikasi Blynk sebagai *platform* untuk mengendalikan Arduino. Hal pertama yang harus dilakukan adalah inialisasi *hardware* atau alat *smart energy meter* terlebih dahulu. Kemudian menghubungkan aplikasi Blynk dan Arduino yang terkoneksi dengan WiFi. Jika sudah terkoneksi, maka sistem akan membaca daya pada masing-masing perangkat. Daya dan waktu pada masing-masing perangkat akan ditampilkan oleh aplikasi Blynk. Jika total nilai daya pada 3 perangkat melebihi 1,5 kWh, maka notifikasi akan muncul pada layar *smartphone* yang memberi informasi bahwa daya yang digunakan pada perangkat sudah melebihi batas. Jika daya melebihi batas, maka langkah selanjutnya adalah mematikan ketiga perangkat tersebut melalui *smart energy meter*.

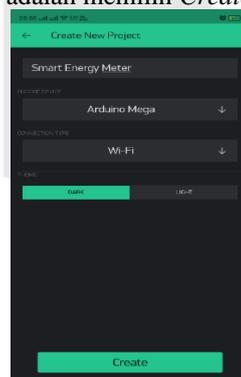


Gambar 5. Flowchart sistem monitoring

3.2 Perancangan Aplikasi Blynk

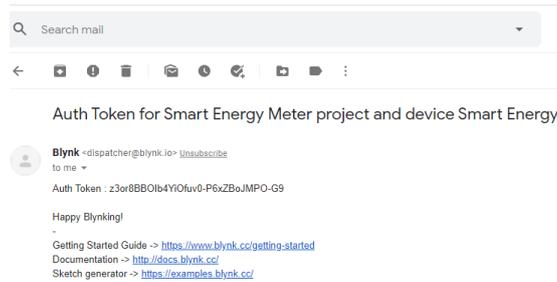
Aplikasi untuk mengontrol *smart energy meter* adalah aplikasi Blynk. Aplikasi Blynk dapat diunduh di Play Store. Langkah-langkah untuk melakukan perancangan aplikasi untuk melakukan *monitoring smart energy meter* adalah sebagai berikut:

1. *Login*. Pada saat pertama kali membuka aplikasi Blynk, tampilan yang muncul pertama kali adalah tampilan untuk *login*. Bagi pengguna yang belum mempunyai akun Blynk dapat mendaftarkan akunnya dengan cara memilih “*Create New Account*”. Sedangkan bagi pengguna yang sudah memiliki akun dapat langsung login ke akun blynk atau juga bisa login dengan menggunakan akun *Facebook*.
2. *New Project*. Pengguna yang sudah berhasil *login* ke aplikasi Blynk dapat membuat proyek baru dengan cara memilih “*New Project*”. Langkah selanjutnya adalah memberi nama *project*, kemudian memilih *hardware* yang digunakan – Arduino Mega – dan memilih *connection type* yang sudah digunakan – Wi-fi – jika semua langkah sudah diatur, langkah selanjutnya adalah memilih *Create*.



Gambar 6. Tampilan *Create New Project*

4. Setelah memilih *Create*, aplikasi akan mengirimkan *Auth Token* ke *email user*. *Auth Token* merupakan kode yang digunakan untuk menghubungkan perangkat dengan aplikasi Blynk.



Gambar 7. Tampilan *auth token* pada *e-mail*

5. Atur tampilan aplikasi Blynk dengan cara memilih *widget* sesuai dengan yang dibutuhkan. *Widget* yang dibutuhkan untuk perancangan sistem *monitoring smart energy meter* adalah *gauge display*, *button*, *labeled value*, dan *notification*.



Gambar 8. Tampilan sistem *monitoring*

4. Pengujian dan Analisis

4.1 Pengujian Delay

Pada **Tabel 4.1** ditunjukkan beberapa kategori *delay* menurut TIPHON[12]. *Delay* dengan kategori sangat bagus, jika *delay* sebesar <150 ms. *Delay* dengan kategori bagus jika *delay* sebesar 150-300 ms. *Delay* dengan kategori bagus jika *delay* sebesar 300-450 ms. *Delay* dengan kategori bagus jika *delay* sebesar >450 ms.

Tabel 1. Parameter *Delay*

Kategori <i>Delay</i>	Besar <i>Delay</i> (ms)
Sangat Bagus	<150 ms
Bagus	150 – 300 ms
Sedang	300 - 450 ms
Jelek	>450 ms

4.1.1 Pengujian *Delay* Sistem *Connecting*

Pengujian ini bertujuan untuk mengetahui *delay* yang dibutuhkan aplikasi Blynk dengan *smart energy meter* ketika mengkoneksikan jaringan internet. Pengujian ini dilakukan dengan cara mengukur waktu *delay* yang dibutuhkan ketika mengkoneksikan aplikasi Blynk dan *smart energy meter* dengan menggunakan jaringan internet.

Tabel 2. Data pengujian *delay* sistem *connecting*

Data ke-	<i>Delay</i> (detik)	Data ke-	<i>Delay</i> (detik)
1	22,07	16	21,96
2	24,54	17	22,09
3	24,38	18	23,6
4	25,24	19	23,05
5	23,09	20	22,67
6	23,06	21	23,56
7	23,37	22	24,99
8	24,28	23	23,5
9	22,23	24	21,75
10	22,44	25	27,75
11	23,27	26	23,17
12	22,99	27	22,44

Data ke-	Delay (detik)	Data ke-	Delay (detik)
13	25,23	28	23,72
14	23,08	29	22,29
15	22,63	30	25,01
Nilai Rata-rata Delay		23,448	

Delay yang terjadi pada sistem *connecting* sebesar 23,448 detik. Hal ini terjadi karena pengujian dilakukan pada jam sibuk *work from home* yaitu jam 10.00 WIB sampai dengan jam 12.00 WIB. Akibatnya koneksi provider Telkomsel mengalami penurunan dan *delay* yang dibutuhkan sangat besar.

4.1.2 Pengujian Delay Button

4.1.2.1 Pengujian Delay Button On

Pengujian ini dilakukan untuk menghitung *delay* yang dibutuhkan dengan cara menekan *button on* untuk menyalakan alat *smart energy meter* melalui aplikasi Blynk, kemudian dihitung waktu yang dibutuhkan relay untuk memutuskan arus listrik pada *smart energy meter*.

Tabel 3. Data pengujian *delay button on*

Data ke-	Delay (detik)	Data ke-	Delay (detik)
1	1,17	16	1,75
2	2,62	17	2,04
3	1,03	18	2,9
4	0,69	19	5,47
5	1,43	20	4,55
6	0,5	21	3,93
7	0,72	22	1,47
8	3,87	23	4,35
9	1,53	24	1,12
10	5,03	25	5,36
11	3,74	26	3,15
12	5,64	27	0,75
13	3,05	28	3,18
14	1,07	29	4,55
15	1,46	30	4,19
Nilai Rata-rata Delay		2,604	

Pengujian ini dilakukan pada tanggal 3 juli sampai dengan 5 juli 2020 pada pukul 11.45 WIB sampai dengan 15.00 WIB. Berdasarkan data tersebut, da engan pukul 13.00 WIB. Faktor yang mempengaruhi *delay* antara lain adalah kecepatan koneksi provider yang digunakan pada jam tersebut mengalami penurunan. Dan Relay pada *smart energy meter* juga terdapat *delay* untuk merespon aksi dari Arduino IDE yaitu sebesar 0,5 detik.

4.1.2.2 Pengujian Delay Button Off

Pengujian ini dilakukan dengan cara menghitung *delay* yang terjadi untuk menonaktifkan *smart energy meter* dengan menekan *button off* pada aplikasi Blynk.

Tabel 4. Data pengujian *delay button off*

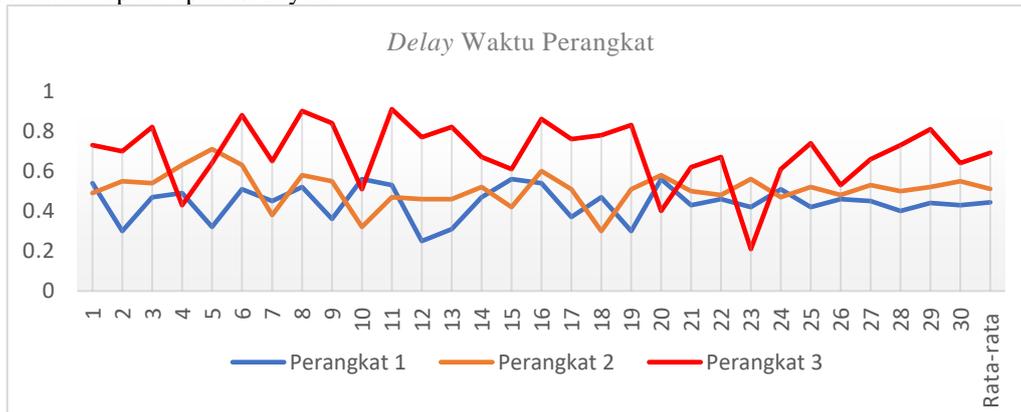
Data ke-	Delay (detik)	Data ke-	Delay (detik)
1	3,38	16	3,5
2	3,62	17	4,89
3	4,69	18	3,71
4	4,12	19	2,68
5	5,06	20	5,47
6	2,81	21	5,49
7	1,6	22	2,83
8	3,37	23	5,12
9	3,46	24	0,87
10	2,47	25	0,9
11	1,8	26	3,67
12	5,11	27	2,51
13	5,39	28	2,37
14	3,46	29	2,22

Data ke-	Delay (detik)	Data ke-	Delay (detik)
15	3,9	30	3,45
Nilai Rata-rata Delay		3,464	

Pengujian ini dilakukan pada tanggal 3 juli sampai dengan 5 juli 2020 pada pukul 20.00 WIB sampai dengan 22.00 WIB. Faktor yang mempengaruhi delay antara lain adalah jam tersebut merupakan jam sibuk ketika malam hari. Akibatnya, kecepatan koneksi provider yang digunakan pada jam tersebut mengalami penurunan. Dan Relay pada smart energy meter juga terdapat delay untuk merespon aksi dari Arduino IDE sebesar 0,5 detik.

4.1.3 Pengujian Delay Daya Perangkat

Pengujian ini dilakukan dengan cara menghubungkan kipas angin, *hairdryer*, dan setrika ke stopkontak pada *smart energy meter*, kemudian dihitung *delay* yang terjadi ketika informasi daya pada masing-masing perangkat muncul pada aplikasi Blynk.

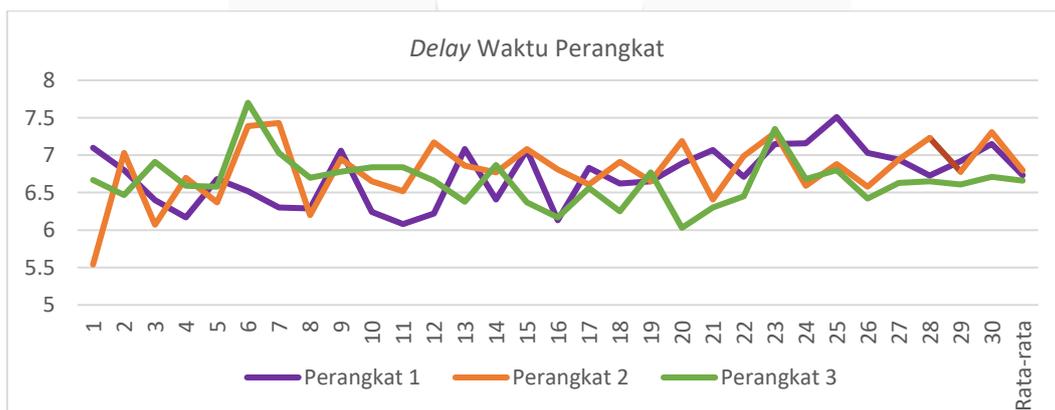


Gambar 9. Hasil pengujian *delay* daya perangkat

Delay yang terjadi pada saat pengiriman daya pada perangkat 1 sebesar 0,443 detik, perangkat 2 sebesar 0,511 detik, dan perangkat 3 sebesar 0,691 detik. *Delay* ini tidak sebesar pengujian sistem connecting. Faktor yang mempengaruhinya adalah ketika melakukan pengujian ketika jaringan 4G Telkomsel tidak berada pada jam sibuk.

4.1.4 Pengujian Delay Waktu Perangkat

Pengujian ini dilakukan dengan cara menghubungkan ketiga perangkat yaitu kipas angin, *hairdryer*, dan adaptor ke stopkontak pada *smart energy meter*, kemudian dihitung *delay* yang terjadi ketika melakukan pengiriman waktu ke aplikasi Blynk.

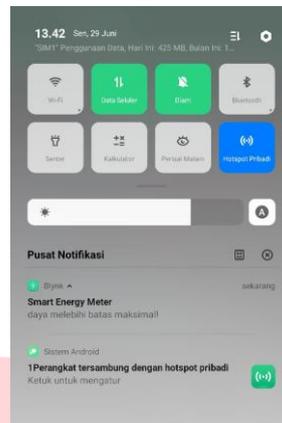


Gambar 10. Hasil pengujian *delay* waktu

Delay yang terjadi saat melakukan pengiriman waktu pada perangkat 1 sebesar 6,7303 detik, perangkat 2 sebesar 6,797 detik, dan perangkat 3 sebesar 6,659 detik. Pengujian ini dilakukan ketika jam sibuk yaitu pukul 21.00 sampai dengan 23.00 WIB. Sehingga saat melakukan pengiriman waktu yang ditampilkan ke aplikasi Blynk terdapat *delay* yang terjadi dengan rata-rata sebesar 6 detik.

4.2 Pengujian Notifikasi

Sistem ini dapat menampilkan notifikasi pada *smartphone*. Notifikasi ini memberikan pemberitahuan atau pesan yang ditampilkan aplikasi Blynk sebagai pengingat bahwa daya perangkat elektronik sudah melebihi batas maksimal. Notifikasi akan muncul ketika daya melebihi batas maksimal yang telah ditentukan.



Gambar 11. Tampilan notifikasi Blynk pada *smartphone*

3. Kesimpulan

Berdasarkan penelitian yang telah dilakukan, dapat diambil beberapa kesimpulan yaitu, alat *smart energy meter* mampu dikendalikan oleh aplikasi Blynk menggunakan jaringan internet. Kemudian dari hasil pengujian *delay*, *delay* yang terjadi pada sistem *connecting* sebesar 23,448 detik. *Delay* yang terjadi pada *button* saat mengaktifkan *smart energy meter* sebesar 2,604 detik. Sedangkan, *delay* saat menonaktifkan *smart energy meter* sebesar 3,464 detik. *Delay* yang terjadi pada saat pengiriman daya pada perangkat 1 sebesar 0,443 detik, perangkat 2 sebesar 0,511 detik, dan perangkat 3 sebesar 0,691 detik. *Delay* yang terjadi saat melakukan pengiriman waktu pada perangkat 1 sebesar 6,7303 detik, perangkat 2 sebesar 6,797 detik, dan perangkat 3 sebesar 6,659 detik. Sistem juga dapat menampilkan notifikasi ketika daya melebihi batas maksimal.

Daftar Pustaka:

- [1] R. Kulkarni, S. Hallur, and P. Patavardhan, "SMART COMPONENTS FOR A SMART ENERGY METER Sudhakar," no. November, 2017.
- [2] A. Maulana, E. Suhartono, T. Yunita, F. T. Elektro, and U. Telkom, "SISTEM PENGUKURAN ENERGI LISTRIK PADA SMART ENERGY METER UNTUK APLIKASI SMART HOUSE YANG MENGGUNAKAN ROOFTOP PHOTOVOLTAIC ELECTRICAL ENERGY MEASUREMENT SYSTEM AT SMART ENERGY METER."
- [3] J. Anthony, "Komponen Arsitektur Android Komponen Arsitektur Android," 2015. [Online]. Available: <http://www.insinyoer.com/komponen-arsitektur-android/>. [Accessed: 21-Oct-2019].
- [4] A. B. Muljono, I. M. A. Nrartha, I. M. Ginarsa, and I. M. B. Suksmadana, "Rancang Bangun Smart Energy Meter Berbasis UNO dan Raspberry Pi," *J. Rekayasa Elektr.*, vol. 14, no. 1, pp. 9–18, 2018.
- [5] Component101, "ESP8266 - WiFi Module," 2018. [Online]. Available: <https://components101.com/wireless/esp8266-pinout-configuration-features-datasheet>. [Accessed: 06-Apr-2020].
- [6] A.-T. Team, "ESP-01 01 WiFi Module Version 1.0," pp. 1–19, 2015.
- [7] Computer Hope, "Smartphone," 2020. [Online]. Available: <https://www.computerhope.com/jargon/s/smartphone.htm>. [Accessed: 28-Mar-2020].
- [8] A. Sharma, R. Kumar, and V. Mansotra, "Proposed Stemming Algorithm for Hindi Information Retrieval," *Int. J. Innov. Res. Comput. Commun. Eng. (An ISO Certif. Organ.)*, vol. 3297, no. 6, pp. 11449–11455, 2016.
- [9] Lenovo, "What is Smartphone?," 2020. [Online]. Available: <https://www.lenovo.com/us/en/faqs/pc-life-faqs/what-is-a-smartphone/>. [Accessed: 28-Mar-2020].
- [10] R. Rudi, I. Dinata, and R. Kurniawan, "Rancang Bangun Prototype Sistem Smart Parking Berbasis Arduino Dan Pemantauan Melalui Smartphone," *J. ECOTIPE*, vol. 4, no. 2, pp. 14–20, 2017.
- [11] Blynk, "How Blynk Works." [Online]. Available: <http://docs.blynk.cc/>. [Accessed: 29-Oct-2019].
- [12] R. Wulandari, "ANALISIS QoS (QUALITY OF SERVICE) PADA JARINGAN INTERNET (STUDI KASUS : UPT LOKA UJI TEKNIK PENAMBANGAN JAMPANG KULON – LIPI)," *J. Tek. Inform. dan Sist. Inf.*, vol. 2, no. 2, pp. 162–172, 2016.