

IMPLEMENTASI SISTEM PENGENDALI RUMAH PINTAR MENGUNAKAN LARAVEL

Laila Khoirunnisa Nurul Imani¹, Nanda Alicia², Fahmizal³, Unan Yusmaniar Oktiawati⁴

^{1,2,3,4}Departmen Teknik Elektro dan Informatika, Sekolah Vokasi, Universitas Gadjah Mada
laila99@mail.ugm.ac.id¹, nandaalicia@mail.ugm.ac.id², fahmizal@ugm.ac.id³, unan_yusmaniar@ugm.ac.id⁴

Abstract – *The control of household electronic equipment can now be done remotely using a website, or better known as the smart home controller application to reduce waste of electricity usage. The electronic equipment that is controlled changes the lights and sockets. This electronic equipment controller website is made with the Laravel framework which uses the PHP 5.8 programming language which is closed, therefore a connection is made with the REST API that uses the Javascript programming language to bridge the communication between electronic equipment and the website. Delivery of data between the website and components requires a server and broker to broker data paths. Thus MQTT is needed as a means of exchanging data on the hardware side so it requires a gateway in the form of Node Js to connect PHP with MQTT. Test results are done by testing the speed of the lights using various internet connection media. The test results showed the fastest internet connection media, namely using a laptop connected to a WiFi internet network with an average speed of 0.691 seconds. The smart home controller website named Pantaw can be accessed using the UGM hotspot WiFi network.*

Keywords: *Website, smarthome, MQTT, Laravel, Node Js.*

Intisari – Pengendali peralatan elektronik rumah tangga kini dapat dilakukan dari jarak jauh menggunakan *website*, atau lebih dikenal dengan istilah aplikasi pengendali rumah pintar guna mengurangi pemborosan pemakaian listrik. Adapun peralatan elektronik yang dikendalikan berupa lampu dan stopkontak. *Website* pengendali peralatan elektronik ini dibuat dengan *framework* Laravel yang menggunakan bahasa pemrograman PHP 5.8 yang bersifat tertutup, dimana kode PHP dijalankan di sisi server. Oleh karena itu, dibuat koneksi dengan REST API yang menggunakan bahasa pemrograman Javascript untuk menjembatani komunikasi antara peralatan elektronik dengan *website*. Pengiriman data antara *website* dan komponen memerlukan sebuah server dan *broker* untuk menjadi perantara jalur data. Dengan demikian MQTT diperlukan sebagai sarana pertukaran data di sisi *hardware* sehingga memerlukan sebuah *gateway* berupa Node Js untuk menghubungkan PHP dengan MQTT. Pengujian *website* dilakukan dengan menguji kecepatan nyala lampu menggunakan berbagai media koneksi internet. Hasil pengujian menunjukkan media koneksi internet yang tercepat yaitu dengan menggunakan laptop yang terkoneksi ke jaringan internet WiFi dengan kecepatan rata-rata 0.691 detik. *Website* pengendali rumah pintar yang diberi nama *Pantaw* ini dapat diakses menggunakan jaringan WiFi UGM *hotspot*.

Kata kunci: *Website, rumah pintar, MQTT, Laravel, Node Js.*

I. PENDAHULUAN

Dewasa ini seluruh sistem penunjang kebutuhan manusia dituntut untuk memiliki mobilitas yang semakin tinggi. Kebutuhan sistem penunjang dengan mobilitas yang tinggi tersebut digunakan untuk mempermudah kehidupan manusia sehingga manusia berlomba-lomba menciptakan inovasi-inovasi baru, terutama dalam bidang teknologi. *Internet of things* (IoT) di masa depan akan mengubah objek dunia nyata menjadi objek virtual yang cerdas[1]. Hal ini berarti setiap orang dan atau sesuatu memiliki *locatable*, *addressable*, dan dapat dibaca di internet[2]. Konsep IoT bertujuan membuat internet dapat lebih dimanfaatkan secara mendalam[3]. IoT dapat dianggap sebagai komponen infrastruktur jaringan global dari berbagai perangkat yang terhubung yang mengandalkan teknologi sensorik, komunikasi, jaringan, dan pemrosesan informasi[4]. IoT mengacu pada interkoneksi jaringan benda sehari-hari, yang dilengkapi dengan kecerdasan[5]. IoT tidak memiliki standar identifikasi[6]. Berkat manajemen daya yang semakin efisien, komunikasi *broadband*, memori yang andal, dan kemajuan dalam teknologi mikroprosesor, telah dimungkinkan untuk mendigitalkan fungsi dan kapabilitas kunci dari produk pada zaman industri[7]. Internet saat ini tidak hanya diakses untuk kebutuhan komunikasi, namun bisa juga digunakan sebagai sarana pengendali jarak jauh seperti pada peralatan elektronik di kantor maupun di rumah. Konsep tersebut lebih dikenal dengan *smarthome* atau rumah pintar.

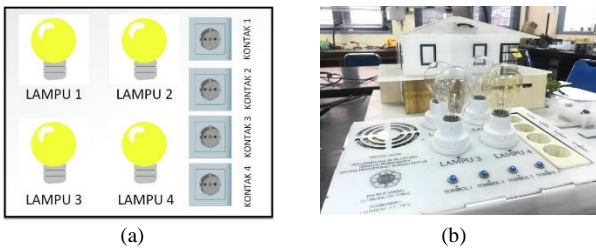
Konsep *smarthome* memainkan peran penting dalam perencanaan model perumahan untuk masa depan[8]. Banyak orang berpendapat bahwa dunia itu sendiri akan dapat dilengkapi indra dan aktuasi dan menciptakan apa yang disebut sebagai dunia yang cerdas[9]. Konsep rumah pintar diharapkan dapat memberikan kenyamanan, keamanan serta efisien bagi pengguna. Konsep ini memudahkan seseorang untuk mengendalikan peralatan elektronik yang ada di rumahnya dari jarak jauh[10]. Kontrol peralatan rumah seperti pendingin udara, pendingin, mesin cuci, dan lain-lain akan memungkinkan pengelolaan rumah dan energi yang lebih baik[11]. Peralatan elektronik yang dikendalikan pada penelitian ini berupa lampu dan stopkontak. Lampu merupakan alat penerang di tempat yang gelap maupun pada malam hari. Lampu sendiri sering digunakan oleh masyarakat sehingga seringkali penggunaannya berlebihan dan menyebabkan pemborosan penggunaan listrik. Salah satu solusi untuk permasalahan tersebut adalah pemilik rumah dapat mengendalikan pemakaian peralatan listrik dari jarak jauh melalui web. Pengendalian berbasis web diharapkan mampu memudahkan pemilik rumah karena dapat dilakukan baik melalui ponsel maupun perangkat lainnya yang terhubung dengan internet dimana saja dan kapan saja. Selain itu dengan akses pengendalian yang mudah diharapkan pengguna dapat memakai peralatannya secara bijak dan untuk menghemat pemakaian listrik. Keseluruhan sistem yang dikembangkan dalam penelitian "Implementasi Sistem Pengendali Rumah Pintar

Menggunakan Laravel" diharapkan dapat memudahkan pengguna untuk mengendalikan peralatan rumah tangga lebih efisien.

II. METODE

2.1 Gambaran Umum Alat

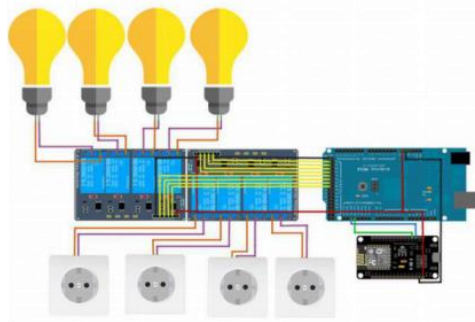
Implementasi *website* berbasis Laravel adalah sebagai alat kontrol elektronik berupa empat buah lampu dan juga empat buah stopkontak yang digunakan untuk berbagai macam alat elektronik seperti kipas angin, lampu, dan lain-lain. Gambar 1.a menunjukkan gambaran umum bentuk *hardware* yang dikontrol menggunakan web berbasis Laravel dan Gambar 1.b menunjukkan bentuk *hardware* sesungguhnya.



Gambar 1. (a) Gambaran rancangan *hardware* (b) Bentuk asli *hardware*

2.2 Skema Perancangan Elektronis

Desain elektronis dibuat dengan menghubungkan empat komponen utama yaitu modul *relay*, Arduino Mega, pada *ethernet shield protocol* (ESP), lampu, dan stopkontak. *Pin* pada Arduino Mega yang digunakan adalah *pin* digital 53, 51, 49, 47 yang terhubung ke *relay* untuk mengatur lampu dan *pin* 45, 43, 41, 39 yang terhubung ke *relay* untuk mengatur stopkontak. *Pin* 'Tx' ESP terhubung ke *pin* 'Rx' pada Arduino Mega, sedangkan *pin* 'Rx' pada ESP terhubung ke *pin* 'Tx' pada Arduino Mega. Berikut detail perancangan elektronis yang terlihat pada Gambar 2.



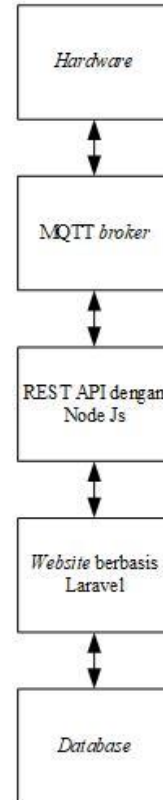
Keterangan warna kabel:

- VCC
- GND
- Lampu/stopkontak ke COM *relay*
- Lampu/stopkontak ke NO *relay*
- Input *relay* ke *pin* digital Arduino Mega (39, 41, 43, 45, 47, 49, 51, dan 53)
- *Pin* Tx Arduino Mega ke *pin* Rx ESP
- *Pin* Rx Arduino Mega ke *pin* Tx ESP

Gambar 2. Skema perancangan elektronis

2.3 Sistem Komunikasi Data

Proses pertukaran data dimulai dari *hardware* yang terhubung ke MQTT *broker*, kemudian baru data dikirim ke *website* berbasis Laravel. Laravel menggunakan bahasa pemrograman PHP yang sifatnya tertutup atau kodenya dijalankan di sisi server[12], sehingga dibuat sebuah *representational state transfer* (REST API) untuk menghubungkan MQTT *broker* dengan *website*. Data yang sudah sampai di web, kemudian dimasukkan ke dalam *database*. Sistem alur komunikasi data ini juga dapat berjalan secara dua arah atau bolak-balik.



Gambar 3. Sistem komunikasi data

2.4 Perancangan Komunikasi MQTT

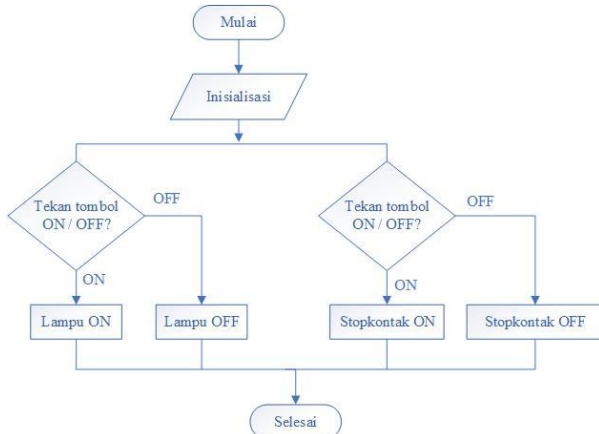
Seperti pada Gambar 3, sebagai media pertukaran data digunakan sebuah MQTT. Pada MQTT struktur pengiriman data menggunakan topik untuk tempat data. Pada tiap-tiap topik akan menjadi acuan tiap data dikirimkan. Setiap data akan ditempatkan atau dikirimkan pada topik-topik yang berbeda. Berikut merupakan pembagian topik yang dijelaskan pada Tabel 1.

Tabel 1 Pembagian topik pada MQTT

Item	Topik	Pesan
Lampu 1	LED/1	0 = OFF ; 1 = ON
Lampu 2	LED/2	0 = OFF ; 1 = ON
Lampu 3	LED/3	0 = OFF ; 1 = ON
Lampu 4	LED/4	0 = OFF ; 1 = ON
Kontak 1	KONTAK/1	0 = OFF ; 1 = ON
Kontak 2	KONTAK/2	0 = OFF ; 1 = ON
Kontak 3	KONTAK/3	0 = OFF ; 1 = ON
Kontak 4	KONTAK/4	0 = OFF ; 1 = ON

2.5 Alur Diagram Aplikasi

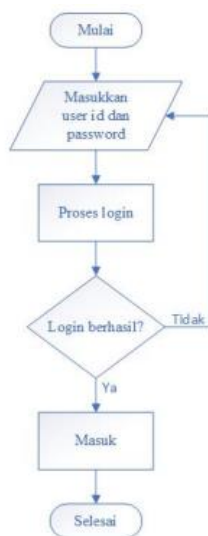
Gambar 4 menjelaskan alur diagram aplikasi dimulai saat pengguna melakukan inisialisasi untuk menyalakan atau mematikan peralatan elektronik. Terdapat delapan tombol yang terdiri dari empat tombol untuk lampu dan empat tombol untuk stopkontak. Pengguna dapat memilih tombol mana yang akan difungsikan. Jika pengguna menekan tombol *on* atau *off*, maka *website* akan mengirim perintah ke Arduino Mega dan akan dilanjutkan menuju alat yang dituju, baik itu lampu maupun stopkontak. Adapun ketika pengguna melakukan *logout*, maka *website* akan diarahkan menuju ke halaman utama.



Gambar 4. Flowchart aplikasi rumah pintar

2.6 Alur Diagram Sistem Login

Gambar 5 menjelaskan alur diagram sistem login. Pengguna memulai dengan mengakses situs *website* lalu masuk ke halaman *login* dan memasukkan *user id* dan *password* yang telah didapat. Jika *user id* dan *password* yang dimasukkan benar, maka pengguna akan diarahkan menuju *dashboard* utama, namun jika gagal maka secara otomatis pengguna akan diarahkan lagi ke halaman *login* untuk memasukkan kembali *user id* dan *password* yang benar.

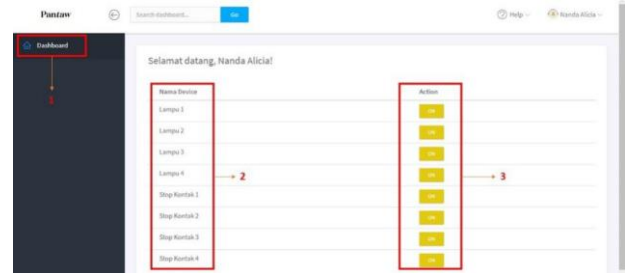


Gambar 5. Flowchart sistem login

2.7 Perancangan Interface

Perancangan *interface website* meliputi perancangan pada halaman-halaman yang terdapat pada *website*, diantaranya yaitu halaman utama, halaman *login*, dan

halaman *dashboard*. Halaman utama ialah halaman yang tertampil ketika pengguna mengakses ke alamat *website*. Halaman *login* ialah halaman untuk pengguna masuk ke dalam akunnya masing-masing untuk dapat mengakses sistem pengendalian pada rumah pintar. Halaman *dashboard* yaitu halaman yang memuat tombol untuk mengendalikan peralatan elektronik yang ada di rumah pengguna. Terdapat delapan tombol atau saklar pada halaman ini yang terdiri dari empat saklar lampu dan empat saklar stopkontak. Gambar 6 menunjukkan tampilan halaman *dashboard* yang terdapat (1) *sidebar* menu pada *dashboard*, (2) nama peralatan yang dimiliki pengguna, dan (3) tombol *switch on/off*.



Gambar 6. Interface halaman utama

Halaman ini hanya dapat diakses oleh pengguna yang sudah terdaftar, sehingga data masing-masing pengguna sangat terjamin keamanannya.

III. HASIL DAN PEMBAHASAN

Website pada penelitian ini berperan sebagai *remote* untuk mengendalikan peralatan elektronik di rumah tangga. Adapun bahasa pemrograman yang digunakan pada Laravel ialah *hypertext preprocessor* (PHP). Sementara bahasa pemrograman PHP itu sendiri bersifat tertutup atau dijalankan di sisi server[12], maka dari itu dibuatlah koneksi dengan REST API untuk menjembatani komunikasi antara peralatan elektronik dengan *website*. REST API menggunakan bahasa pemrograman berupa Javasript. Pengiriman data antara *website* dan komponen memerlukan sebuah server dan *broker* untuk menjadi perantara jalur data. Server yang digunakan ialah server Departemen Teknik Elektro dan Informatika Sekolah Vokasi UGM dengan alamat IP 10.33.109.37. Adapun *broker* yang digunakan sebagai media pertukaran data ialah MQTT yang berfungsi mengirim dan menerima pesan antara *hardware* dengan *website*.

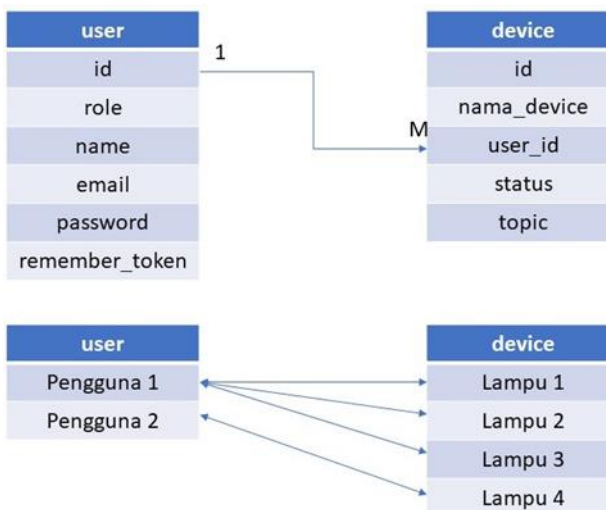
3.1 Implementasi PHP pada Perancangan Antarmuka

Website sistem pengendali rumah pintar ini diberi nama *Pantaw*. Pemilihan *platform website* sebagai basis aplikasi ini dimaksudkan agar pengguna bisa mengaksesnya melalui *smartphone* maupun *desktop*. Tampilan antarmuka *website* pada Laravel terdapat dalam folder “*views/*” yang digunakan untuk menyimpan file-file PHP untuk keperluan tampilan (*frontend*) aplikasi. Setelah pengguna *login* dengan akun *Pantaw*, pengguna akan terarahkan menuju halaman *dashboard* yang dipanggil oleh *routing* lalu diarahkan menuju ke alamat “*/dashboard*”. *Routing* digunakan untuk menangani *request* yang diberikan ke dalam *website*. Setiap ada *request* ke alamat tertentu, *routing* akan mengeksekusi sebelum akhirnya menampilkan respon. *Route* “*/dashboard*” hanya dapat diakses oleh

pengguna yang memiliki akun *Pantaw*. Hal itu dikarenakan pada *route-route* tertentu telah dipasang *middleware* yang mana dalam konteks Laravel merupakan *class* khusus yang berperan sebagai penengah antara *request* tertentu untuk menuju ke *controller*. Dalam hal ini penggunaan *middleware* bertujuan untuk memverifikasi status *login* setiap pengguna untuk menuju *route* “/dashboard”.

3.2 Implementasi PHP pada Perancangan Tabel Basis Data

Konsep pada *website* pengendali rumah pintar ini adalah pengguna dapat mengendalikan peralatan elektronik sesuai dengan yang dimilikinya. Dengan demikian, digunakan dua unsur pengendalian yaitu pengguna dan peralatan yang dikontrol, sehingga dibuat dua tabel pada basis data untuk menyimpan informasi tersebut, yaitu tabel *users* dan *devices*. Maka dari itu perlu dibuat relasi antara kedua tabel tersebut agar bisa terkoneksi satu sama lain berupa relasi *one to many* yang merupakan relasi dimana suatu data pada tabel dapat memiliki banyak data dari tabel lain dan begitu pula banyak data dari tabel dimiliki oleh satu data dari sebuah tabel [13]. Dalam hal ini, terdapat pada hubungan antara tabel *user* dan tabel *device* pada basis data. Gambar 7 menunjukkan skema relasi *one to many* antara tabel *device* dan *user* pada basis data.



Gambar 7. *One to many relationship*

3.3 Implementasi PHP pada Sistem Kendali secara Keseluruhan

Halaman *dashboard* terdiri dari data-data perangkat yang dimiliki oleh pengguna. Perangkat elektronik itu sendiri terdiri dari lampu dan stopkontak yang masing-masing berjumlah empat buah. Terdapat *pushbutton* untuk menyalakan atau mematikan perangkat. Ketika pengguna menekan *pushbutton*, ia akan mengirim *request* ke *route state* pada file “web.php”. Jika *pushbutton* dalam kondisi *ON* ditekan maka ia akan *request* ke *route* dengan *state* ‘off’ dan jika *pushbutton* dalam kondisi *OFF* ditekan maka ia akan *request* ke *route* dengan *state* ‘on’. *Route state* akan mengarahkan menuju *DeviceController.php* dimana pada *controller* tersebut terdapat fungsi *state on* dan *off*. Selain kondisi perangkat tersimpan ke basis data, kondisi tersebut juga akan terpublikasikan ke MQTT *broker* lewat perantara *DashboardController.php* yang mengirim alamat *endpoint* menuju *gateway* untuk kemudian dapat terhubung ke *hardware*. Hasil pengujian ini dapat pula dilihat pada [14].

3.4 Perancangan MQTT

Penggunaan MQTT dimaksudkan untuk perantara pengiriman data. MQTT dipilih karena dengan penggunaannya sebagai media perantara pengiriman data, menjadikan *hardware* menjadi hemat daya. Pembuatan *website* ini ditujukan untuk keperluan mengontrol peralatan elektronik yang ada di dalam beberapa ruangan. Pengontrolan dari sisi *hardware* dilakukan dengan menggunakan Arduino Mega. Selain dengan Arduino Mega, pada sisi *hardware* juga terpasang ESP yang berfungsi sebagai penghubung *hardware* ke jaringan internet. Pengujian MQTT dilakukan dengan menggunakan *extensions* MQTT Lens.

Gambar 8. Detail pengaturan pada MQTT Lens

3.5 Pembuatan Gateway dengan Node Js

REST API merupakan jembatan yang menghubungkan *website* berbasis Laravel dengan MQTT. REST API digunakan karena pada Laravel bahasa yang digunakan adalah bahasa pemrograman PHP, sehingga sifatnya adalah tertutup. Oleh karena itu untuk menghubungkan *website* dengan hal eksternal lainnya digunakanlah REST API. Berikut merupakan program REST API dari sistem ini:

Program REST API tersebut dijalankan pada sisi server, sehingga tidak perlu dilakukan *running* program setiap akan menggunakannya. Pada REST API dibuat *endpoint* untuk mengirim setiap data yang akan dilewatkan di ke *hardware* dari sisi *website*. Nantinya REST API akan meneruskan topik beserta *message* atau pesan ke alamat yang sesuai dari Laravel menuju MQTT.

```

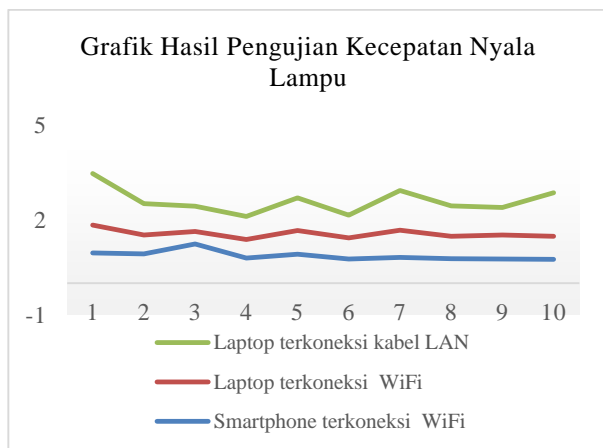
const express = require('express')
const mqtt = require('mqtt')
const client = mqtt.connect('mqtt://10.33.109.37')
const app = express()
app.use(express.json())
app.get('/', (req, res) => {
  res.send('m-gateway')
})
app.post('/endpoint', (req, res) => {
  var topic = req.body.topic
  var message = req.body.message
  client.publish(topic, message)
  res.send(topic + " " + message)}
app.listen(3000, () => {
  console.log('m-gateway running on port 3000')
})
  
```

3.6 Pengujian Koneksi PHP ke MQTT

Jaringan internet digunakan sebagai media pengiriman data melalui server dari Laravel ke MQTT dan *hardware*. Kecepatan kontrol nyala lampu dengan melalui internet diuji dengan menggunakan tiga cara yang berbeda. Berikut hasil dari pengujian kecepatan nyala lampu ditunjukkan pada Tabel 2.

Tabel 2 Pengujian kecepatan nyala lampu

Pengujian ke-n	Smartphone	Laptop	Laptop
	terkoneksi WiFi	terkoneksi WiFi	terkoneksi kabel LAN
	Waktu (detik)		
1	0.95	0.88	1.62
2	0.92	0.60	0.98
3	1.23	0.40	0.79
4	0.79	0.58	0.73
5	0.91	0.75	1.03
6	0.76	0.66	0.72
7	0.81	0.86	1.25
8	0.77	0.70	0.96
9	0.76	0.76	0.86
10	0.75	0.72	1.38
Rata-rata	0.8565	0.691	1.032



Gambar 9. Grafik hasil pengujian kecepatan nyala lampu

Berdasarkan hasil pengujian, dapat diketahui bahwa respon tercepat untuk menyalakan lampu adalah dengan menggunakan laptop yang terhubung ke jaringan internet melalui WiFi.

IV. KESIMPULAN

Berdasarkan hasil pembahasan mengenai "Implementasi Sistem Pengendali Rumah Pintar Menggunakan Laravel" telah dilakukan pengujian kecepatan nyala lampu menggunakan media koneksi internet yang berbeda-beda. Pengujian kecepatan nyala lampu yang tercepat yaitu dengan menggunakan laptop yang terkoneksi ke jaringan internet WiFi dengan kecepatan rata-rata 0.691 detik. Sementara pengujian dengan *smartphone* terkoneksi ke jaringan internet WiFi memiliki kecepatan rata-rata 0.8565 detik dan pengujian dengan laptop yang terkoneksi melalui kabel LAN memiliki kecepatan rata-rata 1.032 detik. *Relasi one to many* antara perangkat dengan *user* menyebabkan keterbatasan akses untuk mengendalikan perangkat elektronik karena satu perangkat hanya bisa dimiliki atau dikendalikan oleh satu pengguna.

DAFTAR PUSTAKA

- [1] S. Madakam, R. Ramaswamy, and S. Tripathi, "Internet of Things (IoT): A Literature Review," *J. Comput. Commun.*, vol. 03, no. 05, pp. 164–173, 2015.
- [2] J. Zheng, D. Simplot-Ryl, C. Bisdikian, and H. T. Mouftah, "The internet of things," *IEEE Commun. Mag.*, vol. 49, no. 11, pp. 30–31, 2011.
- [3] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of things for smart cities," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, 2014.
- [4] L. Da Xu, W. He, and S. Li, "Internet of things in industries: A survey," *IEEE Trans. Ind. Informatics*, vol. 10, no. 4, pp. 2233–2243, 2014.
- [5] O. León, J. Hernández-Serrano, and M. Soriano, "Securing cognitive radio networks," *Int. J. Commun. Syst.*, vol. 23, no. 5, pp. 633–652, 2010.
- [6] L. Tan and N. Wang, "Future Internet: The Internet of Things," *ICACTE 2010 - 2010 3rd Int. Conf. Adv. Comput. Theory Eng. Proc.*, vol. 5, pp. 376–380, 2010.
- [7] F. Wortmann and K. Flüchter, "Internet of Things: Technology and Value Added," *Bus. Inf. Syst. Eng.*, vol. 57, no. 3, pp. 221–224, 2015.
- [8] L. Jiang, D. Y. Liu, and B. Yang, "Smart home research," *Proc. 2004 Int. Conf. Mach. Learn. Cybern.*, vol. 2, no. August, pp. 659–663, 2004.
- [9] J. A. Stankovic, "Research directions for the internet of things," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 3–9, 2014.
- [10] C. Hasiholan, R. Primananda, and K. Amron, "Implementasi Konsep Internet of Things pada Sistem Monitoring Banjir menggunakan Protokol MQTT," *J. Pengemb. Teknol. Inf. dan Ilmu Komput. Univ. Brawijaya*, vol. 2, no. 12, pp. 6128–6135, 2018.
- [11] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Futur. Gener. Comput. Syst.*, vol. 29, no. 7, pp. 1645–1660, 2013.
- [12] A. Aulbach, A. Zmievski, and E. Schmid, "PHP Manual," pp. 1–1520, 2003.
- [13] G. Triyono, "Pertimbangan Melakukan Denormalisasi Pada Model Basis Data Relasi," *J. Telemat. MKOM*, vol. 3, no. 2, pp. 19–25, 2011.
- [14] Fahmizal (2020) Video Implementasi Sistem Pengendali Rumah Pintar menggunakan Website berbasis Laravel. [Online] <https://youtu.be/bHXBHCZdfGg>, tanggal akses: 29-Jan2020.