

# Sistem Keamanan Bangunan Multi Lokasi Berbasis IoT Menggunakan Siemens LOGO! dan Raspberry Pi

<sup>1</sup>Pricillia Alvina, <sup>2</sup>Handy Wicaksono, <sup>3</sup>Petrus Santoso

Program Studi Teknik Elektro, Universitas Kristen Petra

Jl. Siwalankerto 121-131, Surabaya 60236, Indonesia

<sup>1</sup>pricilliaalvina@gmail.com, <sup>2</sup>handy@petra.ac.id, <sup>3</sup>petrus@petra.ac.id

**Abstract** – A building should have a security system to prevent a thief from entering the building or to avoid fire. Many companies have developed an automatic security system; however it is limited to one building only. This research aims to implement an automatic security system based on internet technology to monitor several building which are remotely separated.

This system consists of two Siemens LOGO! as local controllers and a RaspberryPi 3 as a web server. A gas sensor, a fire sensor, a door sensor, and a fan are wired to LOGO!. Node-RED is used to connect various hardware and software. This system has several features: monitoring, controlling, and alarm notification.

Based on the experiment results, the security system performs well. Notifications are successfully sent to the owner's smartphone with an average time of 4.15 seconds. Application of security system at two separate locations via a local network has time average of 70.7 milliseconds, while its application via internet has time average of 66.85 milli seconds.

**Keywords** — Security system, Siemens LOGO!, Raspberry Pi 3, Node-RED

**Abstrak** – Suatu bangunan perlu memiliki sistem keamanan untuk mencegah adanya pencurian ataupun kebakaran. Banyak pihak telah mengembangkan sistem otomatis untuk tujuan tersebut, namun terbatas pada satu bangunan saja. Penelitian ini bertujuan untuk mewujudkan sistem keamanan otomatis berbasis internet untuk mengawasi beberapa bangunan yang terletak saling berjauhan.

Sistem ini terdiri dari dua buah Siemens LOGO! Sebagai pengendali lokal dan RaspberryPi 3 sebagai web server. Sensor gas, sensor api, PIR sensor, sensor pintu, dan kipas angin dihubungkan ke LOGO!. Node-RED digunakan untuk menghubungkan berbagai komponen hardware dan software. Sistem ini memiliki fitur monitoring, controlling dan alarm notification. Pemilik bangunan dapat menerima notifikasi melalui e-mail, LINE message maupun Telegram message.

Berdasarkan hasil pengujian, sistem keamanan berfungsi dengan baik. Notifikasi berhasil dikirimkan menuju smartphone pemilik dengan waktu rata-rata 4.15 detik. Penerapan sistem keamanan di dua lokasi pada jaringan lokal berhasil dengan waktu rata-rata 70.7 mili detik, sedangkan pada jaringan internet berhasil dengan waktu rata-rata 66.85 mili detik.

**Kata Kunci** — Keamanan bangunan; Siemens LOGO!; Raspberry Pi 3; Node-RED

## I. Pendahuluan

Keamanan suatu bangunan dari gangguan pencuri merupakan salah satu hal yang penting bagi pemilik bangunan. Cara umum yang digunakan untuk menjaga keamanan dari pencuri saat ini adalah menggunakan CCTV. CCTV yang berdiri sendiri memiliki beberapa kelemahan seperti CCTV selalu aktif akan menyebabkan pemborosan pada sisi penyimpanan [1]. CCTV pada umumnya juga tidak memiliki fitur alarm baik berupa *buzzer* pada ruangan tersebut atau berupa notifikasi kepada pemilik.

Selain pencuri, hal lain yang perlu diwaspadai adalah bahaya kebakaran. Kebakaran dapat terjadi kapan saja terutama pada kawasan yang padat penduduk. Kebakaran umumnya disebabkan oleh adanya gas bocor dari dapur atau kompor yang lupa dimatikan [2]. Kejadian ini sering terlambat ditangani karena pemilik atau orang di sekitar bangunan tidak segera mengetahui hal tersebut.

Beberapa pemilik yang memiliki lebih dari satu bangunan akan kesulitan untuk mengawasi bangunannya terus menerus. Pengawasan oleh manusia kurang efektif karena resiko keterlambatan untuk menyadari adanya pencurian atau kebakaran. Dengan adanya sistem keamanan yang siap sedia 24 jam sehari dan *real time* dapat membuat pemilik lebih tenang saat tidak berada di dalam bangunan yang dimilikinya [3]. Untuk memantau bangunan-bangunan yang dimiliki, diperlukan adanya sistem yang dapat mengendalikan lebih dari satu lokasi.

Sistem keamanan lain yang pernah dibuat sebelumnya yaitu sistem *monitoring*, kontrol, dan alarm menggunakan dua buah PLC S7-200 di dua lokasi yang berbeda [4]. Sistem ini hanya dapat dikendalikan dengan program Wonderware InTouch yang relatif mahal. Sistem ini kurang *accessible* karena tidak dapat dikendalikan dengan *gadget* lain seperti *smartphone*. Selain itu, notifikasi pada sistem ini tidak dapat dikirim ke *instant messenger* pada *smartphone*.

Sistem lain yang pernah dibuat sebelumnya yaitu membuat sistem *smart home automation* menggunakan kontroler

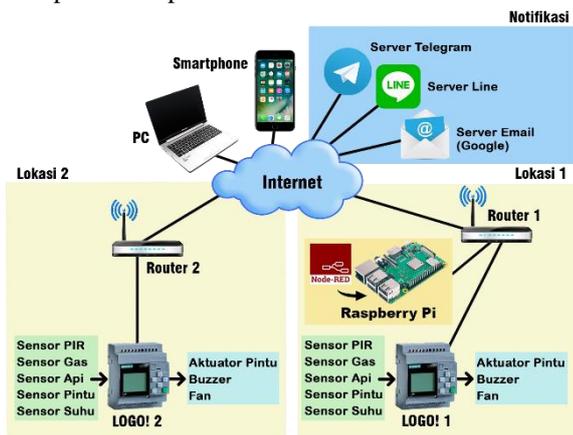
Raspberry Pi 3 Model B dan *cloud* Ubidots sebagai media penyimpanan data [5]. Sistem ini hanya dapat digunakan pada satu lokasi saja karena hanya menggunakan satu kontroler.

## II. Metode Penelitian

Penelitian ini dilakukan dengan melakukan pembuatan desain sistem, desain *software*, dan desain *interface*.

### A. Desain Sistem

Desain sistem keamanan ini bertujuan untuk menjaga keamanan suatu bangunan terhadap adanya pencuri dan terjadinya kebakaran. Sistem ini dapat diterapkan pada rumah, toko, kantor, atau gudang. Sistem ini dapat dikendalikan melalui *gadget* seperti komputer, laptop, atau *smartphone* yang memiliki browser dan koneksi internet. Blok diagram sistem dapat dilihat pada Gambar 1.



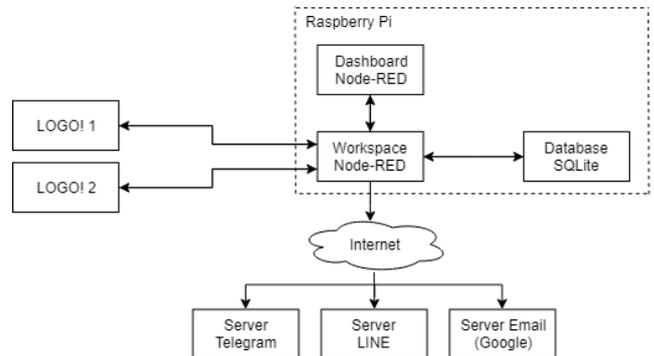
Gambar 1. Blok diagram sistem

Sistem ini dapat mengirimkan notifikasi kepada pemilik bila terjadi sesuatu hal yang tidak diinginkan. Notifikasi yang dikirimkan melalui *e-mail*, Line (*instant messenger*), dan Telegram (*instant messenger*). Setiap notifikasi terkirim sistem juga akan mencatat kapan keadaan yang tidak diinginkan itu terjadi serta jenis sensor yang berkondisi tidak normal. Setiap kali *buzzer* pada lokasi pertama aktif maka *buzzer* pada lokasi kedua juga aktif, begitu pula sebaliknya. Hal ini dilakukan agar ketika pemilik berada di lokasi kedua sedangkan terjadi sesuatu yang tidak diinginkan pada lokasi pertama (atau sebaliknya) maka pemilik dapat langsung mendengar suara *buzzer* yang berbunyi dan segera cek notifikasi.

### B. Desain Software

LOGO! 1 dan LOGO! 2 mengirim status *input*, memori, dan *output* menuju *workspace* Node-RED yang telah terinstall pada Raspberry Pi. Status akan ditampilkan pada *dashboard* Node-RED. Tombol untuk menulis status memori dan *output* terdapat pada *dashboard* Node-RED dan akan dikirimkan ke *workspace* Node-RED lalu diterima oleh LOGO! 1 atau LOGO! 2. *Workspace* Node-RED akan

menambah *query* bila alarm menyala dan disimpan dalam *database* SQLite. Hasil *query* ditampilkan pada *dashboard* Node-RED. Selain menambah *query* bila alarm menyala maka sistem akan mengirim notifikasi kepada pemilik melalui Telegram, LINE, dan *e-mail*. Blok diagram *software components* dapat dilihat pada Gambar 2.



Gambar 2. Blok diagram *software components*

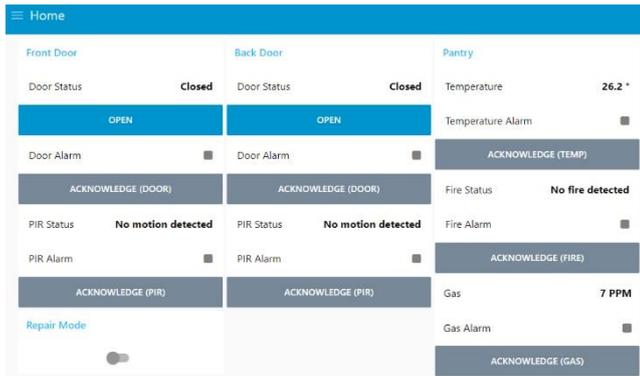
Konfigurasi koneksi jaringan pada Raspberry Pi, LOGO! 1 dan LOGO! 2 dapat dilihat melalui Tabel 1. Raspberry Pi dan LOGO! 1 berada pada lokasi pertama sedangkan LOGO! 2 berada pada lokasi kedua.

Tabel 1. Konfigurasi koneksi Raspberry Pi dan LOGO!

	Raspberry Pi	LOGO! 1	LOGO! 2
IP Address	192.168.0.100	192.168.0.101	192.168.1.101
Subnet Mask	255.255.255.0	255.255.255.0	255.255.255.0
Default Gateway	192.168.0.1	192.168.0.1	192.168.1.1

### C. Desain Interface

*Dashboard* Node-RED merupakan *interface* untuk mengendalikan kedua LOGO!. *Dashboard* Node-RED juga merupakan *front-end* dari Node-RED. Terdapat dua tombol *open* yang digunakan untuk membuka pintu depan dan pintu belakang. Setiap alarm diberi tombol *acknowledge* untuk mematikan *buzzer* yang menyala. Terdapat satu buah *switch* yang digunakan saat salah satu bagian ingin dilakukan perbaikan. Bila *switch repair mode on* maka alarm dan *buzzer* tidak akan menyala. *Dashboard* dapat dilihat pada Gambar 3.



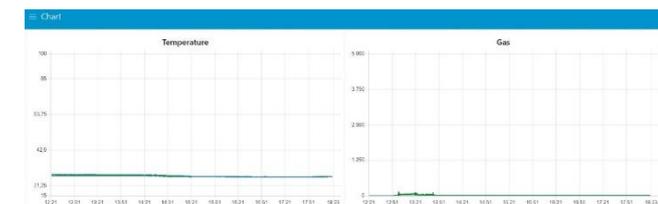
Gambar 3. Dashboard lokasi pertama

Selain pada tab “Home” terdapat tab “Office” yang memiliki tampilan persis dengan tab “Home”. Tab “Logging” yang digunakan untuk menampilkan tabel hari, tanggal, waktu, alarm mana yang menyala, dan lokasi alarm menyala. Dashboard logging dapat dilihat pada Gambar 4.



Gambar 4. Dashboard logging

Tab “Chart” akan menampilkan chart dari sensor suhu dan sensor gas. Sensor suhu lokasi pertama dan lokasi kedua dijadikan dalam satu buah chart. Begitu pula dengan sensor gas, sensor gas lokasi pertama dan kedua dijadikan dalam satu buah chart. Dashboard chart dapat dilihat pada Gambar 5.



Gambar 5. Dashboard charta

### III. Hasil dan Pembahasan

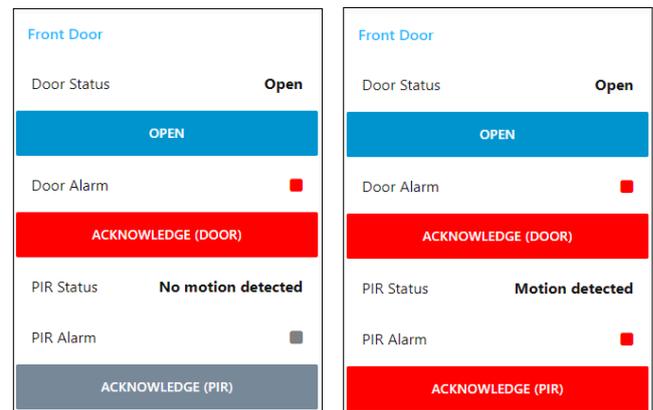
#### A. Pengujian Sistem Keamanan pada Satu Jaringan

Pengujian ini adalah pengujian dimana Raspberry Pi dan LOGO! 1 berada dalam satu jaringan, dapat dilihat pada Tabel 1. Pengujian ini dilakukan untuk mengetahui apakah sensor dan aktuator dapat berjalan sesuai dengan program yang dibuat.

Pengujian ini dilakukan dengan cara membuat keadaan menjadi tidak normal. Pengujian pintu dibuka dilakukan dengan cara aktuator pintu dengan sengaja ditekan masuk kedalam dan pintu dibuka, pengujian ini dapat dilihat pada Gambar 6. Pengujian adanya api dilakukan dengan cara menyalakan korek api di dekat sensor api, pengujian ini dapat

dilihat pada Gambar 8. Pengujian adanya gas dilakukan dengan cara menggunakan kotak lalu melubangi kotak pada satu sisi agar sensor gas dapat masuk dan melubangi kotak pada sisi lainnya untuk memasukkan gas LPG kedalam kotak, pengujian ini dapat dilihat pada Gambar 9. Pengujian terdeteksinya suhu tinggi dilakukan dengan cara menyalakan korek api selama beberapa detik di dekat sensor suhu, pengujian ini dapat dilihat pada Gambar 10. Setelah salah satu keadaan atau lebih menjadi tidak normal lalu melihat apakah *buzzer on*, setelah *buzzer on*, pemilik mengetahuinya, dan pemilik telah menekan tombol *acknowledge* maka *buzzer* akan kembali *off* atau tidak.

Sistem alarm, *buzzer*, dan tombol *acknowledge* mengikuti sistem yang digunakan pada HMI industri dengan beberapa perubahan. Sistem yang digunakan yaitu bila sensor dalam kondisi tidak normal maka *buzzer* akan terus menyala, meskipun kondisi sensor sudah kembali normal *buzzer* tetap menyala hingga pemilik menekan tombol *acknowledge*.

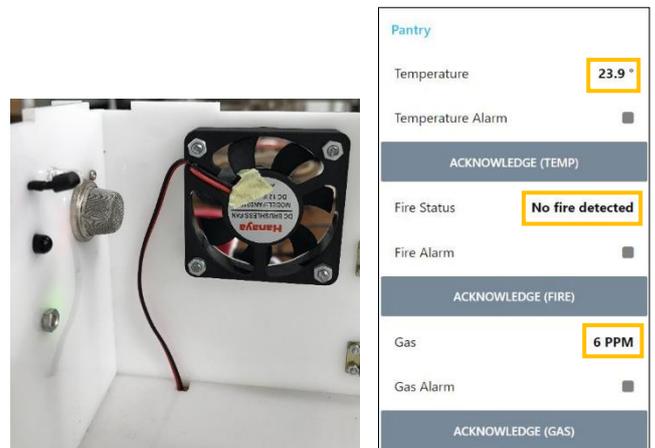


(a) (b)

Gambar 6. Kondisi pintu dibuka paksa

(a) Pintu dibuka paksa tetapi tidak terdeteksi gerakan

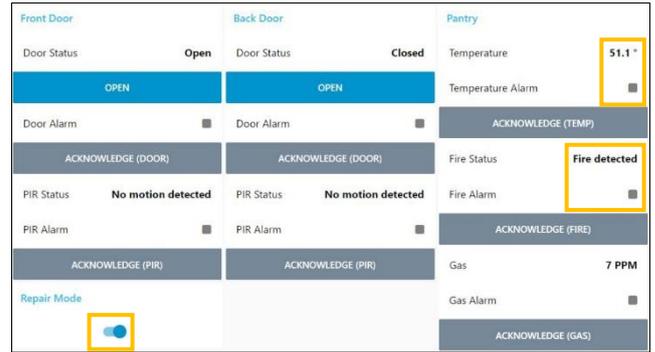
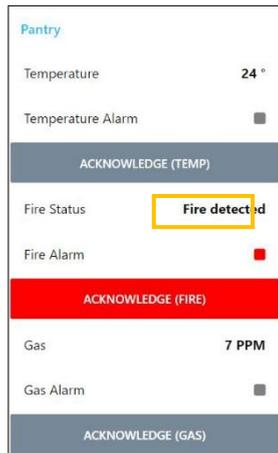
(b) Pintu dibuka paksa dan terdeteksi gerakan



Gambar 7. Kondisi normal dapur



Gambar 8. Kondisi terdeteksi api

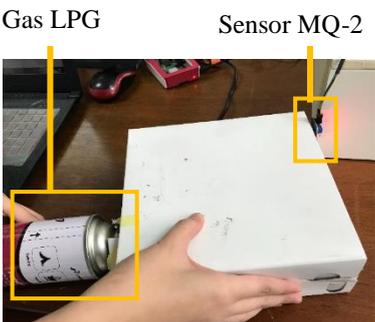


Gambar 11. Mode perbaikan aktif

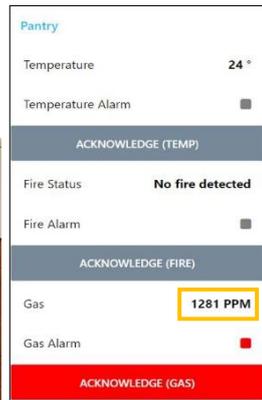
B. Pengujian Sistem Keamanan Antar Jaringan

Pengujian ini adalah pengujian Raspberry Pi dan LOGO! 2 yang berada dalam jaringan yang berbeda, dapat dilihat pada Tabel 1. Pengujian ini dilakukan untuk mengetahui apakah LOGO! 1 dapat mengaktifkan *output* dari LOGO! 2, begitu juga sebaliknya.

Pengujian sistem keamanan antar jaringan dilakukan dengan dengan cara membuat salah satu keadaan menjadi tidak normal pada lokasi pertama. Bila keadaan menjadi tidak normal pada lokasi pertama maka *buzzer* pada lokasi pertama akan aktif, sistem juga akan mengaktifkan *buzzer* pada lokasi kedua, begitu pula sebaliknya. Hasil pengujian dapat dilihat pada Gambar 12.



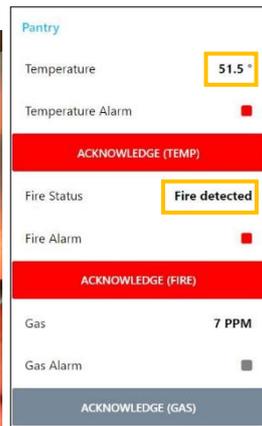
Gambar 9. Kondisi terdeteksi gas



Gambar 12. Kondisi buzzer pada kedua lokasi



Gambar 10. Kondisi terdeteksi suhu tinggi



C. Daftar Terjadinya Alarm

Pengujian ini dilakukan untuk mengetahui apakah *dashboard* Node-RED berhasil menampilkan daftar alarm yang telah terjadi. Daftar alarm ini digunakan untuk mendata seluruh kejadian yang pernah terjadi. Daftar alarm ini disimpan pada *database* SQLite dan ditampilkan menuju *dashboard* Node-RED.

Pengujian ini dilakukan dengan cara membuat kondisi seluruh sensor menjadi tidak normal pada kedua lokasi lalu melihat apakah setiap kondisi sensor-sensor tidak normal tercatat pada *dashboard* Node-RED. Hasil pengujian dapat

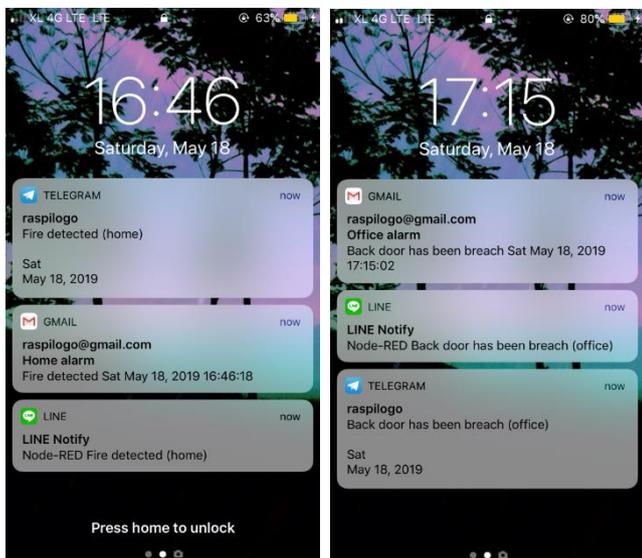
dilihat pada Gambar 13. Seluruh pengujian setiap sensor pada kedua lokasi berhasil tercatat pada *dashboard* Node-RED.

ID	Date	Day	Time	IOStatus	Location
1	May 24, 2019	Fri	11:58:27	Front Door	Home
2	May 24, 2019	Fri	11:58:48	Front Motion	Home
3	May 24, 2019	Fri	11:59:35	Front Door	Office
4	May 24, 2019	Fri	11:59:53	Front Motion	Office
5	May 24, 2019	Fri	12:02:23	Back Door	Home
6	May 24, 2019	Fri	12:02:46	Back Motion	Home
7	May 24, 2019	Fri	12:04:28	Back Door	Office
8	May 24, 2019	Fri	12:04:48	Back Motion	Office
9	May 24, 2019	Fri	12:05:21	Gas	Home
10	May 24, 2019	Fri	12:05:40	Gas	Office
11	May 24, 2019	Fri	12:06:28	Fire	Home
12	May 24, 2019	Fri	12:06:46	Fire	Office
13	May 24, 2019	Fri	12:07:35	Temperature	Home
14	May 24, 2019	Fri	12:08:13	Temperature	Office

Gambar 13. Daftar alarm pada *dashboard* Node-RED

**D. Pengujian Fitur Notifikasi**

Pengujian ini dilakukan untuk mengetahui apakah notifikasi berhasil terkirim setiap kali alarm menyala. Pengujian ini dilakukan dengan cara membuat kondisi sensor menjadi tidak normal lalu melihat apakah setiap kondisi sensor-sensor tidak normal notifikasi berhasil sampai menuju *smartphone*. Jenis notifikasi yang digunakan yaitu Telegram, Line, dan *e-mail* (Google). Hasil dapat dilihat pada Gambar 14.



Gambar 14. Hasil *screenshot* notifikasi pada *smartphone*

Perbandingan waktu *buzzer* aktif hingga notifikasi terkirim menurut waktu pada *e-mail* dapat dilihat pada Tabel 2. Pada Tabel 2 merupakan tabel pada lokasi pertama yang sama dengan lokasi Raspberry Pi diletakkan. Raspberry Pi hanya ada pada lokasi pertama sehingga LOGO! harus mengirimkan datanya menuju Raspberry Pi pada lokasi pertama

Tabel 2. Waktu notifikasi pada lokasi pertama

No	Waktu buzzer aktif	Waktu sampai (email)	Selisih (detik)
1	11:58:27	11:58:30	3
2	11:58:48	11:58:50	2
3	12:02:23	12:02:34	11
4	12:02:46	12:02:50	4
5	12:05:21	12:05:24	3
6	12:06:28	12:06:30	2
7	12:07:35	12:07:40	5
8	12:14:10	12:14:12	2
9	12:14:59	12:15:03	4
10	12:15:41	12:15:44	3
Rata-rata			3.9

Pada Tabel 2 dapat dilihat bahwa waktu tercepat untuk mengirim notifikasi melalui *e-mail* yaitu 2 detik sedangkan waktu terlama yaitu 11 detik. Sedangkan waktu notifikasi tercepat pada lokasi kedua yaitu 2 detik sedangkan waktu terlama yaitu 13 detik dengan rata-rata 4.4 detik. Sekesug rata-rata waktu notifikasi antara lokasi pertama dan lokasi kedua berbeda 0.5 detik. Waktu notifikasi sampai dipengaruhi oleh banyak faktor, beberapa diantaranya yaitu koneksi internet pada Raspberry Pi, koneksi dari server nodemailer, koneksi dari *server* google, dan koneksi internet dari *smartphone*.

**E. Pengujian Waktu Buzzer Aktif**

Pengujian ini dilakukan untuk mengetahui berapa lama waktu yang dibutuhkan untuk mengaktifkan *buzzer* pada lokasi yang berbeda dengan sensor yang mendeteksi kondisi tidak normal. Pengujian ini dilakukan dengan cara membuat kondisi sensor-sensor menjadi tidak normal lalu mencatat waktu ketika sensor tidak normal dan waktu *buzzer* pada lokasi yang berbeda dengan sensor aktif. Waktu yang didapatkan menggunakan waktu pada Raspberry Pi yang ditampilkan melalui *workspace* Node-RED.

Pada pengujian ini dibagi menjadi dua bagian. Bagian pertama yaitu pengujian *buzzer* antar lokasi dalam jaringan lokal dapat dilihat pada Tabel 3. Bagian kedua yaitu pengujian *buzzer* antar lokasi dalam jaringan internet dapat dilihat pada Tabel 4.

Tabel 3. Waktu sensor LOGO! 1 menuju buzzer LOGO! 2 (lokal)

No	Waktu sensor LOGO! 1	Waktu buzzer LOGO! 2	Selisih (mili detik)
1	20:54:45:628	20:54:45:722	94
2	20:54:54:904	20:54:54:978	74
3	20:55:2:472	20:55:2:521	49
4	20:55:6:394	20:55:6:455	61
5	20:55:10:818	20:55:10:887	69
6	20:55:13:940	20:55:14:8	68
7	20:55:17:671	20:55:17:735	64
8	20:55:20:994	20:55:21:51	57
9	20:55:24:912	20:55:24:983	71
10	20:55:29:930	20:55:30:25	95
Rata-rata			70.2

#### IV. Kesimpulan

Kesimpulan yang didapat dari penelitian ini adalah sebagai berikut:

1. Sistem keamanan dapat berjalan sesuai dengan *flowchart* yang dibuat. Sistem keamanan pada dua lokasi berhasil dikendalikan melalui *dashboard* Node-RED pada Raspberry Pi.
2. Sistem alarm berhasil dan notifikasi *e-mail* berhasil terkirim ke *smartphone* dalam waktu rata-rata 3.9 detik dan 4.4 detik. Sedangkan notifikasi Telegram dan Line juga berhasil terkirim ke *smartphone*. Rata-rata waktu yang didapatkan tergolong cukup cepat.
3. Penerapan sistem keamanan di dua lokasi pada jaringan lokal berhasil dengan waktu rata-rata 70.2 dan 71.2 mili detik, sedangkan pada jaringan internet berhasil dengan waktu rata-rata 83.8 dan 49.9 mili detik. Rata-rata waktu yang didapatkan tergolong cukup cepat.

#### V. Daftar Pustaka

- [1] M. Mukhsin, F. and I. Jaenuri, "Rancangan Bangun Prototype Monitoring Keamanan Rumah Berbasis Closed Circuit Television (CCTV) dengan Detektor Gerak," *Widya Teknika*, vol. 22, no. 1, pp. 7-13, March 2014.
- [2] S. S. Dewi, D. Satria, E. Yusibani and D. Sugiyanto, "Sistem Deteksi Kebakaran pada Kasus Kebocoran Gas Berbasis SMS Gateway," *Seminar Nasional II USM*, vol. 1, pp. 106-109, October 2017.
- [3] F. Sirait, "Sistem Monitoring Keamanan Gedung berbasis Raspberry Pi," *Jurnal Teknologi Elektro, Universitas Mercu Buana*, vol. 6, pp. 55-60, January 2015.
- [4] Y. N. Santoso, "Sistem scada berbasis internet untuk model otomasi bangunan," Universitas Kristen Petra, Surabaya, 2012.
- [5] G. M. Limanta, "Pengembangan dan pembuatan sistem home automation memanfaatkan internet of things cloud platform pada Laboratorium Elektronika Universitas Kristen Petra," Universitas Kristen Petra, Surabaya, 2017.

Tabel 4. Waktu sensor LOGO! 1 menuju buzzer LOGO! 2 (internet)

No	Waktu sensor LOGO! 1	Waktu buzzer LOGO! 2	Selisih (mili detik)
1	11:6:45:662	11:6:45:747	85
2	11:6:49:602	11:6:49:673	71
3	11:6:54:939	11:6:55:5	66
4	11:7:8:439	11:7:8:706	267
5	11:7:17:987	11:7:18:51	64
6	11:7:22:614	11:7:22:671	57
7	11:7:27:629	11:7:27:695	66
8	11:7:33:273	11:7:33:337	64
9	11:7:38:703	11:7:38:759	56
10	11:7:44:345	11:7:44:387	42
Rata-rata			83.8

Rata-rata waktu sensor LOGO! 1 menuju buzzer LOGO! 2 dapat dilihat pada Tabel 3 (dalam jaringan lokal) dan Tabel 4 (dalam jaringan internet). Sedangkan rata-rata waktu sensor LOGO! 2 menuju buzzer LOGO! 1 dalam jaringan lokal yaitu 71.2 mili detik, rata-rata waktu sensor LOGO! 2 menuju buzzer LOGO! 1 dalam jaringan internet yaitu 49.9 mili detik.

Dari hasil pengujian dapat dilihat bahwa selisih waktu pada jaringan lokal tidak berbeda jauh. Selisih rata-rata hanya 1 mili detik. Pada Tabel 4 rata-rata selisih waktu mencapai 83.8 mili detik karena pada percobaan keempat terlihat waktu mencapai 267 mili detik, sedangkan pada pengujian sebaliknya rata-rata selisih waktunya hanya 49.9 mili detik. Hal ini dapat terjadi karena pada jaringan internet terdapat data-data lain yang dikirim sehingga meskipun jaringan internet lebih cepat daripada jaringan lokal, jaringan lokal lebih stabil dibandingkan jaringan internet.