

SISTEM KEAMANAN DENGAN KONTROL RFID MENGGUNAKAN E-KTP DAN INTERNET OF THINGS (IoT)

SECURITY SYSTEM WITH RFID CONTROL USING E-KTP AND INTERNET OF THINGS (IoT)

Andi Ainun Najib¹, Prof. Dr. Ir. Rendy Munadi M.T.², Dr. Nyoman Bogi Aditya K, ST.MSEE.³

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

¹andiainunnajib59@gmail.com , ²rendymunadi@telkomuniversity.co.id ,

³aditya@telkomuniversity.ac.id

Abstrak

Keamanan adalah hal yang perlu diperhatikan untuk menciptakan rasa kenyamanan ketika berada di rumah atau ketika pemilik rumah meninggalkan rumah untuk bekerja atau berpergian. Penggunaan kunci konvensional masih rentan terhadap pembobolan. Maka dibutuhkan sistem keamanan yang lebih efisien dan praktis. Untuk mengatasi hal tersebut, maka dibutuhkan *Internet of Things (IoT)*. *Internet of Things (IoT)* adalah teknologi yang menggunakan internet sebagai media penghubung tanpa perlu mengontrol benda secara langsung.

Pada Tugas Akhir ini, akan diusulkan penggunaan *Internet of Things (IoT)* dengan menggunakan sensor *Radio Frequency Identification (RFID)*, dan *buzzer* sebagai penanda, solenoid sebagai pengunci pintu, E-KTP sebagai RFID tag dan juga aplikasi yang dibuat dengan Android Studio yang terhubung dengan *NodeMCU V3 ESP8266* dan *firebase* sebagai pengganti kunci konvensional. Apabila terdapat gerakan disekitar rumah maka sistem akan memberikan notifikasi melalui aplikasi yang terhubung dengan sensor PIR.

Dari hasil pengujian dan analisis menunjukkan bahwa sensor PIR dapat mendeteksi gerakan di sekitar alat dan RFID dapat mendeteksi E-KTP dengan jarak maksimum 4 cm. Penggunaan penghalang menunjukkan bahwa E-KTP sangat dipengaruhi oleh induksi elektromagnetika, semakin tebal bahan penghalang semakin berkurang jarak pendeteksian. Pendeteksian E-KTP berpengaruh terhadap sudut yang digunakan, semakin besar sudut pendeteksian semakin dekat jarak deteksi E-KTP. Pada aplikasi, untuk fitur *lock* dan *open* mendapatkan nilai rata-rata *throughput* 18366,667 b/s dengan indeks sangat bagus, *packet loss* bernilai 0,03% dengan indeks sangat bagus, *delay* bernilai 65,268 ms dengan indeks sangat bagus, dan *jitter* bernilai 65,028 ms dengan indeks bagus. Untuk fitur *alert* pengirim notifikasi menghasilkan nilai rata-rata *throughput* 18066,667 b/s, *packet loss* dengan indeks sangat bagus, bernilai 0,09% , *delay* dengan indeks sangat bagus, bernilai 67,235 ms, dan *jitter* dengan indeks bagus, bernilai 67,561 ms.

Kata kunci : *Internet of Things, Solenoid, Radio Frequency Identification, Android Studio, NodeMCU V3 ESP8266, E-KTP, Sensor PIR, Buzzer, Relay.*

Abstract

Safety is a matter that must be considered to create a sense of comfort when at home or when the homeowner leaves the house to work or travel. The use of conventional keys is still vulnerable to breaking. So we need a security system that is more efficient and practical. To overcome this, an Internet of Things (IoT) is needed. Internet of Things (IoT) is a technology that uses the internet as a connecting medium without the need to control objects directly.

In this Final Project, the proposed use of Internet of Things (IoT) using Radio Frequency Identification (RFID) sensors, solenoids as door locks, E-KTP as RFID tags and also applications made with Android Studio are connected to NodeMCU V3 ESP8266 as storage data and connect with firebase instead of conventional keys. If a motion occurs the system will provide notification via the application connecting with PIR sensor.

The results of testing and analysis show that the PIR sensor can detect movement around the device and RFID can detect E-KTP with a maximum distance of 4 cm. The use of a barrier shows that E-KTP is strongly influenced by electromagnetic induction, the thicker the barrier the less the detection distance. E-KTP detection affects the angle used, the greater the detection angle

the closer the E-KTP detection distance. In the application, the lock and open features get an average throughput value of 18366,667 b/s with a very good index, packet loss is worth 0,03% with a very good index, delay is worth 65,2686 ms with a very good index, and jitter is worth 65,028 ms with a good index. For the alert feature the sender of the notification produces an average value of throughput 18066,667 b/s, packet loss with a very good index, worth 0,09%, delay with a very good index, is 67,235 ms, and jitter with a good index, is worth 67,561 ms.

Keywords: *Internet of Things, Solenoid, Radio Frequency Identification, Android Studio, NodeMCU V3 ESP8266, E-KTP, Sensor PIR, Buzzer,relay.*

1. Pendahuluan

Keamanan rumah adalah salah satu hak seseorang untuk mempertahankan barang berharga dan data pribadi, dan juga berhak mendapatkan kenyamanan di dalam rumah. Kejahatan terhadap hak milik tanpa menggunakan kekerasan dalam hal ini adalah pencurian dan pembobolan sering terjadi. Dikutip dari Badan Pusat Statistik, kejadian Kejahatan Pencurian Tanpa Penggunaan Kekerasan merupakan jenis kejahatan yang paling banyak dalam segi jumlah setiap tahunnya[1].

Pada penelitian Tugas Akhir membuat suatu sistem keamanan *Smart Door Lock* menggunakan sensor RFID dan sensor PIR pada pengaman pintu yang akan terhubung menggunakan E-KTP. *Prototype* yang dibuat pada Tugas Akhir ini dapat dikontrol menggunakan aplikasi sehingga memudahkan pemilik rumah dalam meningkatkan keamanan rumah. Penggunaan E-KTP sebagai RFID tag karena, Fungsi E-KTP untuk sekarang ini hanya sebagai indetitas saja[2].

Dalam penelitian ini, dilakukan implementasi dari alat pengaman pintu menggunakan sensor RFID dan sensor PIR yang terhubung dengan *NodeMCU V3 ESP8266* dan aplikasi android. Fungsi dari sensor RFID sebagai pendeteksi ID dari pemilik rumah, sensor PIR sebagai sensor pendeteksi gerakan dan juga sebagai fungsi alarm yang akan terhubung dengan aplikasi android. Aplikasi android berfungsi untuk mengontrol pengamanan pintu secara *online* dan juga menerima data dari sensor PIR jika terdeteksi gerakan di dalam rumah, *NodeMCU V3 ESP8266* untuk mengirim log data yang akan diteruskan ke *firebase*. Fungsi dari *firebase* sebagai tempat penyimpanan data secara *real-time*.

2. Dasar Teori

2.1 Internet of Things

Internet of Things atau dikenal juga dengan singkatan IoT, merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus. Adapun kemampuan seperti berbagi data, *remote control*, dan sebagainya, termasuk juga pada benda di dunia nyata. Contohnya bahan pangan, elektronik, koleksi, peralatan apa saja, termasuk benda hidup yang semuanya tersambung ke jaringan lokal dan global melalui sensor yang tertanam dan selalu aktif[3].

2.5 Parameter uji

Parameter uji adalah bagian dari penelitian yang nilai dari hasil pengujian berdasarkan hasil dari performasi dari *Prototype* yang dibuat. Adapun parameter uji hasil performasi dalam Tugas Akhir ini :

2.5.1 Throughput

Throughput merupakan jumlah total kedatangan paket yang sukses yang diamati pada destination selama interval waktu tertentu dibagi oleh durasi interval waktu tersebut[4].

Tabel 2.1. Kategori *Throughput*[4].

Kategori <i>Throughput</i>	Besar <i>Throughput</i> (bps)	indeks
Sangat Bagus	100	4
Bagus	75	3
Sedang	50	2
Buruk	< 25	1

Nilai throughput dapat dicari dengan persamaan :

$$Throughput = \frac{\text{Data diterima}}{\text{Lama waktu pengamatan}} \quad (2.1)$$

2.5.2 Packet loss

Packet loss adalah suatu parameter yang memberikan suatu kondisi yang menunjukkan jumlah packet yang hilang[3].

Tabel 2.2. Kategori *Packet Loss*[4].

Kategori <i>Paket Loss</i>	Paket loss	indeks
Sangat Bagus	0%	4
Bagus	3%	3
Sedan	15%	2
Buruk	25%	1

Packet loss dapat dicari menggunakan persamaan :

$$Packet\ loss = \frac{Total\ Packet\ yang\ hilang}{Total\ Packet\ yang\ terkirim} \times 100\% \tag{2.2}$$

2.5.3 Delay

Delay Adalah waktu yang dibutuhkan data untuk menempuh jarak dari titik asal ke titik tujuan[3].

Tabel 2.3. Kategori *Delay*[4].

Kategori <i>Delay</i>	Besar <i>Delay</i> (ms)	indeks
Sangat Bagus	< 150 ms	4
Bagus	150 s/d 300 ms	3
Sedan	300 s/d 450 ms	2
Buruk	> 450 ms	1

Nilai delay dapat dicari dengan persamaan :

$$Delay = \frac{Total\ delay}{Total\ packet\ yang\ diterima} \tag{2.3}$$

2.5.4 Jitter

Jitter diakibatkan oleh variasi-variasi dalam panjang antrian, dalam waktu pengolahan data, dan juga dalam waktu penghimpunan ulang paket-paket diakhir perjalanan[4].

Tabel 2.4. Kategori *Delay*[4].

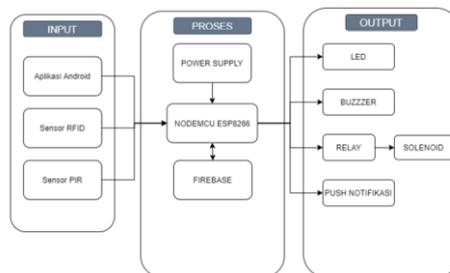
Kategori <i>Jitter</i>	<i>Jitter</i> (ms)	indeks
Sangat Bagus	0 ms	4
Bagus	0 s/d 75 ms	3
Sedan	75 s/d 125 ms	2
Buruk	125 s/d 225 ms	1

Nilai jitter dapat dicari dengan persamaan :

$$Delay = \frac{Total\ variasi\ delay}{Total\ packet\ yang\ diterima} \tag{2.4}$$

3. Perancangan dan Simulasi Sistem

3.1. Desain Sistem



Gambar 3.1. Diagram blok.

Sistem dimulai saat pengguna mendekati E-KTP (RFID Tag) ke RFID Reader, informasi dari RFID Reader dikirim ke NodeMCU ESP8266 dan mengecek data E-KTP yang terdaftar. Saat data sesuai, NodeMCU ESP8266 memerintahkan relay untuk menggerakkan solenoid untuk membuka pengaman pintu, dan memerintahkan buzzer agar mengeluarkan output suara (suara singkat). Sistem juga dapat menggunakan aplikasi android, pada aplikasi android terdapat beberapa fitur yaitu fitur lock, fitur open, fitur alert. Aplikasi android dapat memberikan push notification berupa alert yang berisi pesan, ketika sensor PIR mendeteksi gerakan di sekitar ruangan.

3.1.1 Komponen Pendukung Perangkat Keras.

Tabel 3.1. Komponen Pendukung Perangkat Keras.

No	Komponen	Kegunaan
1	RFID Reader	Mendeteksi E-KTP, Mengolah data <i>user</i>
2	E-KTP	Tanda pengenal <i>user</i> , Terdapat chip
3	Solenoid	Mengunci pintu secara <i>electric</i>
4	Relay	Berupa saklar atau <i>switch</i>
5	Buzzer	Sebagai tanda sistem berfungsi
6	NodeMCU	Mikrokontroler sistem
7	Sensor PIR	Sensor gerakan
8	Baterai Lithium Ion 18650	Daya pada <i>solenoid</i>
9	Powerbank	Daya pada <i>NodeMCU ESP8266</i>
10	LED	Sebagai indikator wifi terhubung

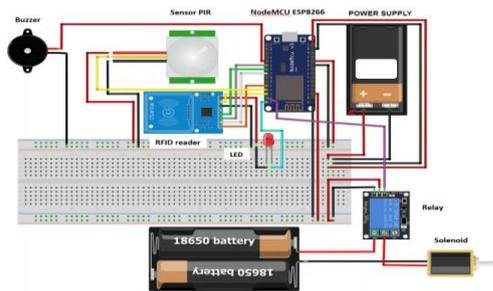
3.1.1 Aplikasi dan Software

Tabel 3.2. Komponen Pendukung Aplikasi.

No.	Perangkat lunak	Kegunaan
1	Android Studio	Perangkat lunak pembuat aplikasi
2	C++	Bahasa pemrograman pada sistem
3	Firebase	sebagai <i>database</i>
4	Arduino IDE	<i>Software</i> pemrograman
5	Pusher	platform <i>push notification</i>
6	Pycharm	<i>Software</i> pemrograman pesan
7	Wireshark	<i>Software</i> pengukuran Qos

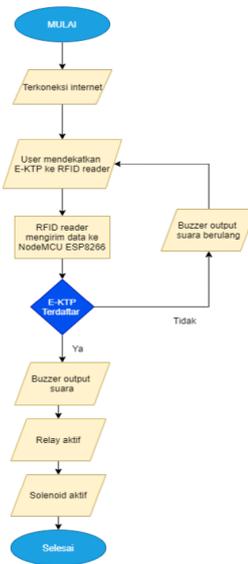
3.2. Desain Perangkat Keras

Berikut adalah beberapa desain dari komponen utama yang digunakan pada *prototype* dalam Tugas Akhir ini :



Gambar 3.2. Perancangan rangkaian perangkat keras.

3.2.1 Flowchart Pada Alat



Gambar 3.3. Flowchart proses penggunaan fitur dalam alat.

Pada gambar 3.3 dijelaskan user mendekatkan E-KTP ke RFID reader dan diproses dengan NodeMCU 8266 jika E-KTP terdaftar maka relay aktif sehingga solenoid terbuka dan mengaktifkan buzzer. dan jika E-KTP tidak terdaftar maka solenoid tidak aktif .

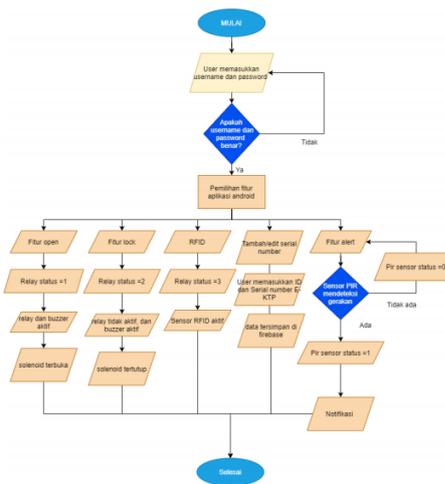
3.3 Desain Perangkat Lunak

Berikut adalah desain perangkat lunak (aplikasi android) yang digunakan pada Tugas Akhir ini:



Gambar 3.4. Tampilan aplikasi.

3.3.1 Flowchart Pada Aplikasi



Gambar 3.5. Flowchart pada aplikasi

Pada gambar 3.9 menjelaskan tentang proses penggunaan perangkat lunak aplikasi. pada aplikasi terdapat tiga fitur yaitu : *lock* untuk mengunci pintu, *open* untuk membuka pintu, dan *alert* untuk mengaktifkan sistem alarm dan memberikan info jika terdapat pergerakan di dalam ruangan ketika pemilik rumah sedang tidak ada di rumah. Sistem *alert* mengirimkan *push notification* dari data sensor PIR melalui aplikasi android.

4. Hasil dan Analisis

Pada pengujian digunakan 6 E-KTP (3 E-KTP yang terdaftar pada *firebase*, 3 E-KTP yang tidak terdaftar)

Tabel 4.1. Serial number E-KTP.

No	Nama Pemilik E-KTP	Serial Number	No	Nama Pemilik E-KTP	Serial Number
1	ANDI AINUN NAJIB	05 8E 3F 02 B8 61 00	1	FICHRAM ALFIANSYAH	05 80 E3 A1 73 A1 00
2	ASTRID MAYDIANA	04 7E 6F 32 FF 24 80	2	DWIEKA SEPTIAN ARIF PRASETYA	04 6A 60 9A 34 4E 80
3	KHOIR MU'ARIF	04 72 74 6A 1B 51 80	3	MUHAMMAD IMANSYAH BASUDEWA	04 74 2B E2 E7 2E 80

1. Pengujian pengukuran keluaran RFID :

(a) Pengujian pembacaan E-KTP tanpa penghalang.

Tabel 4.2. Pembacaan E-KTP Tanpa Penghalang.

Percobaan ke-	Jarak (cm)	E-KTP ANDI	E-KTP ASTRID	E-KTP KHOIR
1	0	Terdeteksi	Terdeteksi	Terdeteksi
2	0,5	Terdeteksi	Terdeteksi	Terdeteksi
3	1	Terdeteksi	Terdeteksi	Terdeteksi
4	1,5	Terdeteksi	Terdeteksi	Terdeteksi
5	2	Terdeteksi	Terdeteksi	Terdeteksi
6	2,5	Terdeteksi	Terdeteksi	Terdeteksi
7	3	Terdeteksi	Terdeteksi	Terdeteksi
8	3,5	Terdeteksi	Terdeteksi	Terdeteksi
9	4	Terdeteksi	Terdeteksi	Terdeteksi
10	4,5	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi

(b) Pengujian pembacaan E-KTP dengan penghalang.

Tabel 4.3. Pengujian dengan bahan kertas

Percobaan ke-	Jarak (cm)	E-KTP ANDI	E-KTP ASTRID	E-KTP KHOIR
1	0	Terdeteksi	Terdeteksi	Terdeteksi
2	0,5	Terdeteksi	Terdeteksi	Terdeteksi
3	1	Terdeteksi	Terdeteksi	Terdeteksi
4	1,5	Terdeteksi	Terdeteksi	Terdeteksi
5	2	Terdeteksi	Terdeteksi	Terdeteksi
6	2,5	Terdeteksi	Terdeteksi	Terdeteksi
7	3	Terdeteksi	Terdeteksi	Terdeteksi
8	3,5	Terdeteksi	Terdeteksi	Terdeteksi
9	4	Terdeteksi	Terdeteksi	Terdeteksi
10	4,5	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi

Tabel 4.4. Pengujian dengan bahan kain

Percobaan ke-	Jarak (cm)	E-KTP ANDI	E-KTP ASTRID	E-KTP KHOIR
1	0	Terdeteksi	Terdeteksi	Terdeteksi
2	0,5	Terdeteksi	Terdeteksi	Terdeteksi
3	1	Terdeteksi	Terdeteksi	Terdeteksi
4	1,5	Terdeteksi	Terdeteksi	Terdeteksi
5	2	Terdeteksi	Terdeteksi	Terdeteksi
6	2,5	Terdeteksi	Terdeteksi	Terdeteksi
7	3	Terdeteksi	Terdeteksi	Terdeteksi
8	3,5	Terdeteksi	Terdeteksi	Terdeteksi
9	4	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
10	4,5	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi

Tabel 4.5. Pengujian dengan bahan dompet

Percobaan ke-	Jarak (cm)	E-KTP ANDI	E-KTP ASTRID	E-KTP KHOIR
1	0	Terdeteksi	Terdeteksi	Terdeteksi
2	0,5	Terdeteksi	Terdeteksi	Terdeteksi
3	1	Terdeteksi	Terdeteksi	Terdeteksi
4	1,5	Terdeteksi	Terdeteksi	Terdeteksi
5	2	Terdeteksi	Terdeteksi	Terdeteksi
6	2,5	Terdeteksi	Terdeteksi	Terdeteksi
7	3	Terdeteksi	Terdeteksi	Terdeteksi
8	3,5	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
9	4	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
10	4,5	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi

(c) Pengujian pembacaan E-KTP terhadap derajat kemiringan.

Tabel 4.6. Pengujian dengan derajat kemiringan 10°.

Percobaan ke-	Jarak (cm)	E-KTP ANDI	E-KTP ASTRID	E-KTP KHOIR
1	0	Terdeteksi	Terdeteksi	Terdeteksi
2	0,5	Terdeteksi	Terdeteksi	Terdeteksi
3	1	Terdeteksi	Terdeteksi	Terdeteksi
4	1,5	Terdeteksi	Terdeteksi	Terdeteksi
5	2	Terdeteksi	Terdeteksi	Terdeteksi
6	2,5	Terdeteksi	Terdeteksi	Terdeteksi
7	3	Terdeteksi	Terdeteksi	Terdeteksi
8	3,5	Terdeteksi	Terdeteksi	Terdeteksi
9	4	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
10	4,5	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi

Tabel 4.7. Pengujian dengan derajat kemiringan 50°.

Percobaan ke-	Jarak (cm)	E-KTP ANDI	E-KTP ASTRID	E-KTP KHOIR
1	0	Terdeteksi	Terdeteksi	Terdeteksi
2	0,5	Terdeteksi	Terdeteksi	Terdeteksi
3	1	Terdeteksi	Terdeteksi	Terdeteksi
4	1,5	Terdeteksi	Terdeteksi	Terdeteksi
5	2	Terdeteksi	Terdeteksi	Terdeteksi
6	2,5	Terdeteksi	Terdeteksi	Terdeteksi
7	3	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
8	3,5	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
9	4	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
10	4,5	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi

(d) Pengujian pembacaan E-KTP yang tidak terdaftar.

Tabel 4.8. Pengujian dengan E-KTP tidak terdaftar.

Percobaan ke-	Jarak (cm)	E-KTP FICHRAM	E-KTP DWIEKA	E-KTP IMAN
1	0	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
2	0,5	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
3	1	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
4	1,5	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
5	2	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
6	2,5	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
7	3	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
8	3,5	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
9	4	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
10	4,5	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi

2. Pengujian kinerja aplikasi :

(a) Pengujian performasi Qos terhadap fitur *lock* dan *open*.**Tabel 4.9.** Tabel hasil perhitungan Qos pada fitur *lock* dan *open*.

Percobaan ke-	Throughput (bps)	Packet loss (%)	Delay (ms)	Jitter (ms)
Percobaan ke-1	27000	0	60,879	60,666
Percobaan ke-2	27000	0	63,918	63,658
Percobaan ke-3	22000	0	63,294	63,2934
Percobaan ke-4	13000	0	67,712	67,421
Percobaan ke-5	23000	0	49,912	49,754
Percobaan ke-6	16000	0	74,171	73,935
Percobaan ke-7	11000	0	66,978	66,688
Percobaan ke-8	28000	0	54,880	54,633
Percobaan ke-9	12000	0	67,666	67,417
Percobaan ke-10	15000	0	66,898	66,659
Percobaan ke-11	12000	0	68,937	68,738
Percobaan ke-12	18000	0	62,039	62,003
Percobaan ke-13	13000	0,3	69,892	69,723
Percobaan ke-14	21000	0	64,455	64,204
Percobaan ke-15	22000	0	64,239	63,983
Percobaan ke-16	19000	0	65,547	64,736
Percobaan ke-17	15000	0	67,092	66,983
Percobaan ke-18	23000	0	62,594	62,238
Percobaan ke-19	14000	0,6	67,878	67,698
Percobaan ke-20	20000	0	62,738	62,634
Percobaan ke-21	19000	0	66,892	66,773
Percobaan ke-22	23000	0	60,983	60,673
Percobaan ke-23	13000	0	68,234	67,829
Percobaan ke-24	19000	0	70,234	69,877
Percobaan ke-25	21000	0	63,023	63,002
Percobaan ke-26	13000	0	66,934	66,873
Percobaan ke-27	16000	0	70,209	70,023
Percobaan ke-28	14000	0	68,928	68,454
Percobaan ke-29	18000	0	67,343	67,143
Percobaan ke-30	24000	0	63,549	63,124
Rata-rata	18366,667	0,03	65,268	65,028

(b) Pengujian performasi Qos terhadap fitur *alert* pengirim notifikasi.**Tabel 4.10.** Tabel hasil perhitungan Qos pada fitur *lock* dan *open*.

Percobaan ke-	Throughput (bps)	Packet loss (%)	Delay (ms)	Jitter (ms)
Percobaan ke-1	19000	0	68,714	67,058
Percobaan ke-2	24000	0,7	71,689	71,943
Percobaan ke-3	11000	0,3	72,600	72,350
Percobaan ke-4	32000	0	63,044	71,814
Percobaan ke-5	17000	0,6	72,811	72,769
Percobaan ke-6	14000	0,8	67,924	67,825
Percobaan ke-7	16000	0	65,694	65,417
Percobaan ke-8	11000	0	73,070	72,346
Percobaan ke-9	12000	0	74,095	74,095
Percobaan ke-10	23000	0	62,038	70,485
Percobaan ke-11	12000	0	65,983	65,595
Percobaan ke-12	19000	0,3	69,836	69,547
Percobaan ke-13	21000	0	63,648	63,736
Percobaan ke-14	16000	0	65,783	65,583
Percobaan ke-15	12000	0	71,023	70,938
Percobaan ke-16	26000	0	61,864	61,647
Percobaan ke-17	21000	0	64,634	64,142
Percobaan ke-18	12000	0	74,004	73,890
Percobaan ke-19	18000	0	66,764	66,145
Percobaan ke-20	19000	0	65,927	65,564
Percobaan ke-21	24000	0	63,468	63,284
Percobaan ke-22	13000	0	66,823	66,462
Percobaan ke-23	28000	0	56,732	56,564
Percobaan ke-24	11000	0	74,537	74,273
Percobaan ke-25	27000	0	60,039	59,894
Percobaan ke-26	12000	0	75,835	75,536
Percobaan ke-27	22000	0	63,283	63,230
Percobaan ke-28	23000	0	49,918	49,733
Percobaan ke-29	16000	0	74,170	73,934
Percobaan ke-30	11000	0	71,100	71,022
Rata-rata	18066,667	0,09	67,235	67,561

5. Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan hasil dan pembahasan pada penelitian yang telah dilakukan, maka dapat disimpulkan bahwa:

1. Performansi dari sensor PIR dan RFID bekerja sangat baik dalam mendeteksi gerakan, dan mendeteksi E-KTP yang didekatkan pada RFID.
2. Sensor RFID dapat mendeteksi E-KTP dengan jarak jangkauan maksimum 4 cm tanpa penghalang, dan dengan penghalang kertas, kain, dan dompet dapat menjangkau jarak masing-masing 4 cm, 3,5 cm, dan 3 cm. Dikarenakan E-KTP merupakan RFID tag pasif, sehingga E-KTP hanya aktif jika induksi elektromagnetik yang dipancarkan oleh RFID *reader* mengenai E-KTP. Induksi elektromagnetik pada bahan yang lebih tebal seperti dompet hanya menjangkau jarak 3 cm karena induksi elektromagnetik dari RFID *reader* dihalangi oleh bahan tersebut. Derajat kemiringan berpengaruh terhadap pembacaan RFID *reader*.
3. Performansi Qos pada fitur *lock* dan fitur *open* menghasilkan nilai rata-rata *throughput* 18366,667 b/s (Sangat bagus), *packet loss* 0,03% (Sangat bagus), *delay* 65,268 ms (Sangat bagus), 65,028 ms *jitter* (Bagus).
4. Performansi Qos pada fitur *alert* pengiriman *push notification* menghasilkan nilai rata-rata *throughput* 18066,667 b/s (Sangat bagus), *packet loss* 0,09% (Sangat bagus), *delay* 67,235 ms (Sangat bagus), *jitter* 67,561 ms (Bagus).
5. Pada *firebase* dapat menerima data yang dikirim dari alat dan juga dapat menyimpan data secara *real-time*. dan juga data *Serial number* dapat dihapus dan diubah langsung di *firebase* sehingga memudahkan pemilik dalam mengatur *Serial number* yang digunakan.
6. Alat dan aplikasi android dapat bekerja dengan baik sesuai yang diharapkan. *NodeMCU ESP8266* yang digunakan dapat mengirim log data ke *firebase* dan mendownload data *Serial number* pada *firebase*.

Daftar Pustaka:

- [1] S. S. P. dan Keamanan, *Statistik Kriminal 2020*.
- [2] E-ktp, identitas penduduk yang unik dan otentik. [Online]. Available: <https://www.bppt.go.id/profil/sejarah/848-e-ktp-identitas-penduduk-yang-unik-dan-otentik>
- [3] Y. Yudhanto, "Apa itu iot (internet of things)," Diunduh di <http://ilmukomputer.org/wp-content/uploads/2015/05/apa-itu-iot-internet-of-things.vol.2,2007>.
- [4] F. R. Rivai, R. Munadi, and U. Sunarya, "Analisis dan implementasi prototipe pengatur kelembaban berbasis internet of things (iot) pada penyimpanan sayur," eProceedings of Engineering, vol. 5, no. 3, 2018