

## RANCANG BANGUN SISTEM KEAMANAN MOTOR DENGAN GPS DAN NOTIFIKASI ANDROID

Nadhif Fauzan<sup>1</sup>, Kiki Prawiroredjo<sup>2</sup>

<sup>1,2</sup>Jurusan Teknik Elektro Fakultas Teknologi Industri Universitas Trisakti

Jln. Kyai Tapa No.1 Grogol Pertamburan, Jakarta Barat

[nadhiffauzan21@gmail.com](mailto:nadhiffauzan21@gmail.com), [kiki.prawiroredjo@trisakti.ac.id](mailto:kiki.prawiroredjo@trisakti.ac.id)\*

### Abstract

*The large number of motorbikes used has an impact on the case of losses. More than 100 cases of motorbike theft occur every week. The police have urged the use of double locks on motorcycle users. This security system can recognize the motorbike's owner, designed using a microcontroller, a fingerprint sensor, a GPS module, relays, a buzzer and a vibration sensor. To ignite his motorbike, the owner must verify himself using a fingerprint sensor. The GPS tracking system can locate the motorbike if it is stolen in the middle of the road. The relay disconnects and connects the electrical contacts of the motor. The vibration sensor will detect the movement of the motorbike during parking and will trigger the buzzer and send notifications to the owner's smartphone. Based on the test results, it is known that the average response time for sending data from the security device to the Android application is 17.863 seconds. The average response time required for the security device to send an active vibration sensor alert notification to Android application is 7.16 seconds. The average response time to start the motor using the fingerprint sensor until the motor starts is 4.77 seconds. The average response time to start the motor using the Android application is 10.51 seconds.*

**Keywords :** *GPS tracking, motorcycle security system, NodeMCU, fingerprint sensor*

### Abstrak

Banyaknya penggunaan sepeda motor berdampak pada meningkatnya jumlah kasus pencurian. Lebih dari 100 kasus pencurian sepeda motor terjadi setiap minggunya. Aparat kepolisian telah menghimbau untuk mengaplikasikan kunci ganda pada pengguna sepeda motor. Sistem keamanan ini dibuat agar motor dapat mengenali pemiliknya, dirancang menggunakan mikrokontroler, sensor sidik jari, modul GPS, relai, buzzer dan sensor getar. Untuk menyalakan sepeda motornya, selain menyalakan kunci kontak pemilik harus memverifikasi dirinya menggunakan sensor sidik jari. Sistem tracking GPS membuat pemilik mengetahui keberadaan motornya apabila terjadi pencurian ditengah perjalanan. Komponen relai memutuskan dan menyambungkan kelistrikan kontak motor, sensor getar akan mendeteksi gerakan motor pada saat parkir yang akan memicu buzzer untuk berbunyi dan mengirimkan notifikasi pada smartphone pemilik motor. Berdasarkan hasil pengujian diketahui bahwa waktu respon rata-rata pengiriman data dari alat pengaman sampai ke aplikasi Android rata-rata selama 17,863 detik. Waktu respon rata-rata yang dibutuhkan alat untuk mengirimkan notifikasi peringatan sensor getar aktif pada aplikasi Android 7,16 detik. Waktu respon rata-rata untuk menyalakan motor menggunakan sensor sidik jari sampai motor menyala adalah 4,77 detik. Waktu respon rata-rata untuk menyalakan motor menggunakan aplikasi Android adalah 10,51 detik.

**Kata kunci :** *GPS tracking, keamanan motor, nodeMCU, sensor sidik jari*

## 1. PENDAHULUAN

Sepeda motor berperan penting dalam memfasilitasi mobilitas masyarakat ibu kota dan sebagian besar sudah menjadi kebutuhan primer. Walaupun sudah tersedia berbagai macam moda transportasi umum di Jakarta, nyatanya masih banyak masyarakat yang menganggap sepeda motor jauh lebih menghemat waktu dalam menunjang mobilitas kegiatan setiap hari. Di tahun 2019, jumlah kendaraan bermotor di Jakarta tercatat sebanyak 11.839.921 kendaraan yang terdiri dari empat jenis kendaraan yaitu mobil, sepeda motor, bus, dan truk. Dari keempat jenis kendaraan di tahun 2019, persentase jumlah kendaraan sepeda motor di Jakarta adalah yang tertinggi dibandingkan dengan jenis kendaraan lainnya yaitu 69% atau sebanyak 8.194.590 unit [1]. Sepeda motor masih menjadi moda transportasi masyarakat Jakarta yang paling digemari dikarenakan beberapa alasan seperti harga yang relatif terjangkau, irit bahan bakar, lebih hemat biaya parkir dan juga praktis dalam mencari tempat parkir.

Pencurian sepeda motor makin marak terjadi seiring dengan tingginya penggunaan sepeda motor sebagai mobilitas penduduk. Data kepolisian mencatat masih banyak kasus pencurian kendaraan bermotor khususnya kendaraan roda dua. Lebih dari 100 kasus pencurian motor terjadi selama sepekan. Menurut kepala biro Penerangan masyarakat Divisi Humas Polri Brigjen Awi Setiyono pencurian kendaraan sepeda motor roda dua pada minggu ke-26 (pekan keempat Juni 2020) sebanyak 177 kasus, pada minggu ke-24 (pekan kedua Juni 2020) tercatat sebanyak 226 kasus, dan pada minggu ke-25 (pekan ketiga bulan Juni 2020) sebanyak 160 kasus [2]. Kejadian pencurian sepeda motor biasanya relatif singkat, membuat pemiliknya tak menyangka sepeda motornya raib seketika. Tidak hanya di malam hari dan di tempat yang sepi saja, di siang hari dan di keramaian pun para pencuri dapat melakukan aksinya dengan mudah.

Dalam upaya pencegahan terjadinya pencurian motor, aparat kepolisian mengimbau untuk selalu menggunakan kunci ganda [3]. Pada penelitian ini dibuat sebuah sistem keamanan motor berbasis aplikasi Android yang terhubung dengan jaringan internet. Bila sistem mendeteksi adanya tindakan pencurian, alarm pada motor akan otomatis berbunyi dan pemilik mendapatkan notifikasi langsung pada *smartphon*enya. Untuk dapat menyalakan mesin motor pemilik harus melakukan verifikasi terlebih dahulu ke sistem dengan menggunakan

sidik jarinya. Modul GPS dapat menginformasikan posisi keberadaan motor kepada pemiliknya secara *real time*.

## 2. TINJAUAN PUSTAKA DAN TEORI

### 2.1. Penelitian Terdahulu

Pada penelitian sebelumnya Dwiyanu Indra Prasetya dkk. telah membuat Sistem Keamanan Sepeda Motor Menggunakan Kata Sandi Berbasis Arduino Nano [4]. Sistem keamanan ini menggunakan *keypad* untuk memasukkan *password* sebagai keamanannya untuk menyalakan kontak motor. Santo Tjhin, dkk. [5] juga telah membuat sistem keamanan sepeda motor melalui *short message service* menggunakan mikrokontroler ATmega8 dimana sistem mengontrol kontak motor melalui sms dari perangkat seluler. Yoki Purnama Putra juga yang membuat sistem keamanan menggunakan *hotspot* WiFi dengan *smartphone* sebagai *interfacenya* [6]. Sistem diatas memiliki kekurangan dimana sistem hanya dapat mengontrol kelistrikan kontak motornya saja, sehingga hanya dapat mencegah pencuri untuk menyalakan mesin tanpa menggunakan kunci dan tidak dapat mendeteksi adanya tindak pencurian. Motor tetap bisa dibawa kabur pencuri dengan cara menggotong motornya tanpa menyalakan mesin.

Beberapa penelitian lain ada yang memanfaatkan modul GPS untuk sistem keamanan motor seperti yang dibuat oleh Dendy Pratama, dkk. [7]. Sistem tersebut mengirimkan koordinat lokasi motor yang diperoleh dari modul GPS menggunakan sms. Ada juga yang berbasis *web service* untuk sistem *tracking*-nya yang dibuat oleh Ari Nugroho [8]. Kekurangan dari sistem tersebut adalah tidak praktis karena data yang diterima pemilik motor hanya berupa angka koordinatnya saja dengan *interface* yang kurang efektif, sehingga untuk melakukan *tracking* posisi motor pun menjadi sulit karena tidak dapat dilakukan secara *real time*. Pada sistem *tracking* GPS yang dibuat, sistem dapat menampilkan posisi motor langsung pada tampilan *interface* Google Maps, serta dapat mengirimkan notifikasi pada aplikasi Android *smartphone* pemiliknya.

### 2.2. Sistem Informasi Berbasis Cloud Database Internet

Sistem informasi berbasis *cloud database* adalah sebuah sistem menggunakan teknologi internet untuk mengirim dan menerima informasi atau data melalui *interface* yang

ditujukan kepada pengguna untuk mengakses berbagai informasi yang disimpan pada *cloud* atau *server*. Pada intinya teknologi ini menjadikan internet sebagai tempat menyimpan data dan mengelola data-data pengguna.

Adapun syarat-syarat agar terbentuknya sebuah sistem informasi berbasis *cloud database* ini adalah:

1. Tersedianya jaringan internet.
2. Tersedianya *software* pemrograman berbasis *server*.
3. Tersedianya *cloud database*.

Tujuan-tujuan sistem informasi berbasis *cloud database* dimaksudkan untuk mencapai 3 tujuan utama, yaitu:

1. Dapat mengakses data atau informasi di manapun dan kapanpun, melalui sebuah *interface* yang terkoneksi dengan jaringan internet.
2. Perubahan data menjadi lebih cepat tersampaikan. Setelah data pembaruan terpasang di *cloud server*, semua pengguna dapat langsung mengaksesnya.
3. Keamanan data pengguna dapat disimpan dengan aman pada *cloud server* yang disediakan oleh penyedia layanan *cloud*.

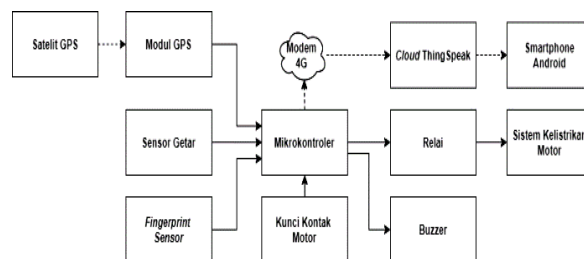
### 2.3. Sistem Keamanan

Sistem keamanan tradisional dinilai berhasil dalam melindungi kendaraan motor seperti menggunakan kunci ganda, namun hal tersebut masih terdapat celah keamanan yang dapat dijebol pencuri. Ada beberapa teknik yang dapat digunakan untuk mengamankan kendaraan motor seperti pengaplikasian sistem keamanan motor berbasis GPS, notifikasi otomatis ke pemilik apabila ada pencuri mengaktifkan kontak motor, menggunakan sensor sidik jari untuk menyalakan motor, serta menggunakan sensor getar untuk mendeteksi adanya gerakan pada saat motor sedang diparkir.

## 3. METODOLOGI PENELITIAN

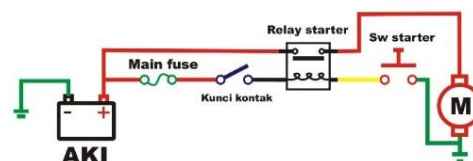
### 3.1. Perancangan Perangkat Keras

Perangkat keras yang digunakan pada sistem ini diantaranya adalah Mikrokontroler Wemos D1, modul sensor sidik jari AS608, modul GPS Neo 6m, sensor getaran SW 420, relai SPST, dan *buzzer*.



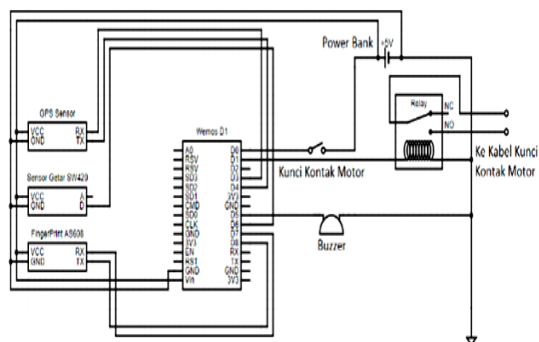
Gambar 1. Diagram Blok Sistem Keamanan Motor

Garis terputus pada blok diagram Gambar 1 menunjukkan komunikasi *wireless*, sedangkan garis penuh merupakan komunikasi menggunakan kabel. Sistem keamanan motor ini memiliki tiga buah *input* yaitu modul GPS, sensor getar, sensor sidik jari dan sebuah relai sebagai pembaca *input* yang didapat dari keadaan posisi kunci kontak motor. Ketiga input dan relai tersebut dihubungkan ke mikrokontroler Wemos D1 sebagai pemrosesnya. Sistem terhubung dengan jaringan internet menggunakan modem 4G untuk dapat terkoneksi dengan *interface* pada aplikasi Android melalui *cloud database* ThingSpeak. Modul GPS menerima sinyal dari satelit dan memprosesnya menjadi data *latitude* dan *longitude* yang akan di tampilkan pada aplikasi Android. Sistem keamanan motor ini memiliki output sebuah relai yang menggantikan peran kunci kontak motor dan sebuah *buzzer* sebagai indikator suara.



Gambar 2. Wiring Diagram Starter Sepeda Motor

Sistem keamanan motor ini disisipkan pada *wiring* kelistrikan motor seperti pada Gambar 2. *Switch* kunci kontak pada motor akan diputus dari kelistrikan motor dan dihubungkan ke relai *input* pada sistem dan relai output pada sistem dihubungkan ke kelistrikan sepeda motor sebagai pengganti kunci kontak motor seperti pada skema rangkaian Gambar 3.



Gambar 3. Skema Rangkaian Sistem Keamanan Motor

Pada sistem keamanan motor ini apabila kunci kontak diputar pada posisi *on* maka motor tidak akan langsung bisa dinyalakan karena kunci kontak motor tidak lagi terhubung langsung dengan kelistrikan motor. Kunci kontak hanya akan mengubah relai *input* ke posisi NC yang akan dideteksi oleh sistem. Sistem keamanan ini menggunakan *supply* tegangan yang terpisah dengan kelistrikan motor agar sistem dapat terus mengirimkan data ke *interface* Android walaupun aki dari motor itu sendiri dilepas. Sensor sidik jari berfungsi untuk menyimpan dan memverifikasi data identitas pemilik motor, sensor getar berfungsi untuk mendeteksi adanya gerakan dari sepeda motor. Sensor getar hanya akan mendeteksi gerakan apabila posisi kunci kontak *off* atau motor sedang diparkir. Modul GPS berfungsi untuk mengirimkan data *latitude* dan *longitude* posisi motor dari satelit GPS [9]. Data dari ketiga *input* tersebut akan diproses oleh mikrokontroler Wemos D1 dan terhubung ke WiFi berupa modem 4G melalui modul ESP8266 yang terintegrasi pada Wemos D1.

### 3.2. Perancangan perangkat Lunak

Perangkat lunak pada sistem ini terdiri dari dua bagian, yaitu mikrokontroler dan aplikasi Android. Perancangan perangkat lunak pada mikrokontroler bertujuan untuk membuat algoritma pensaklaran pada kontak kelistrikan motor yang sudah terintegrasi dengan sensor GPS, sensor sidik jari, dan juga sensor getar. Perancangan aplikasi Android bertujuan untuk memberikan informasi posisi motor menggunakan Google Map API untuk pemetaan digital, sekaligus dapat mengontrol sistem pensaklaran secara jarak jauh [10]. Terdapat 3 mode pada sistem keamanan motor ini yaitu Mode Parkir, Mode Otomatis dan Mode Manual.

#### Mode Parkir

Pada saat sistem keamanan motor berada di mode parkir, sistem akan mencegah mesin motor untuk menyala saat kunci diputar paksa ke posisi *on* tanpa proses verifikasi menggunakan sidik jari. Apabila sistem tidak mendapatkan verifikasi selama 10 detik setelah kunci kontak diputar pada posisi *on*, maka akan memicu *buzzer* untuk berbunyi dan mengirimkan notifikasi kepada pemilik motor melalui aplikasi Android pada *smartphonenya*. Pada mode ini juga dapat mencegah pencuri dari menggotong paksa motor karena sensor getar yang mendeteksi adanya gerakan pada motor yang akan memicu *buzzer* untuk berbunyi dan juga mengirimkan notifikasi kepada *smartphone* pemiliknya.

#### Mode Otomatis

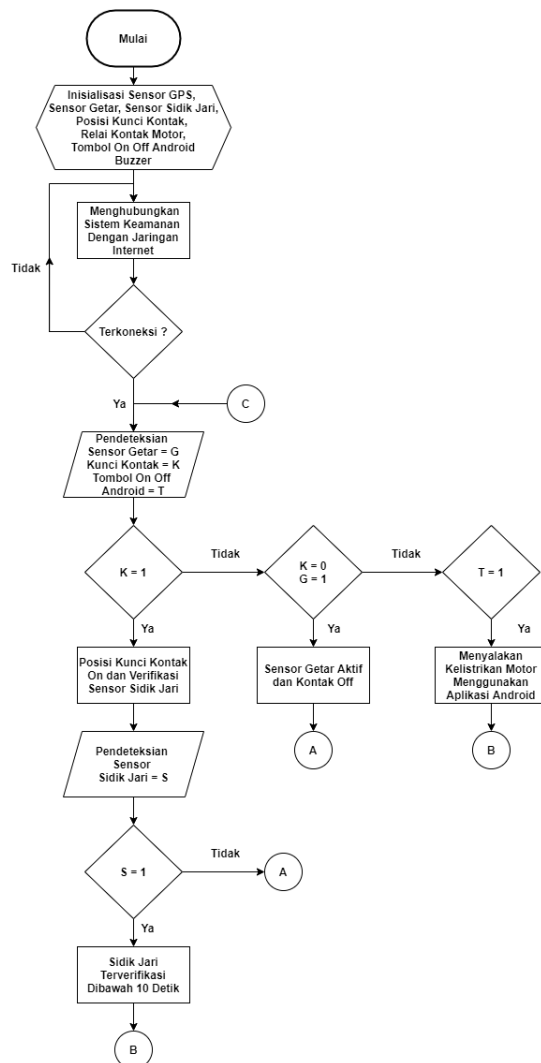
Mode ini digunakan untuk menyalakan atau mematikan mesin motor dengan aplikasi Android tanpa melakukan proses verifikasi sidik jari. Mode ini digunakan apabila terjadi hal-hal yang tidak memungkinkan untuk melakukan proses verifikasi sidik jari, misalnya pada saat jari pengguna terluka.

#### Mode Manual

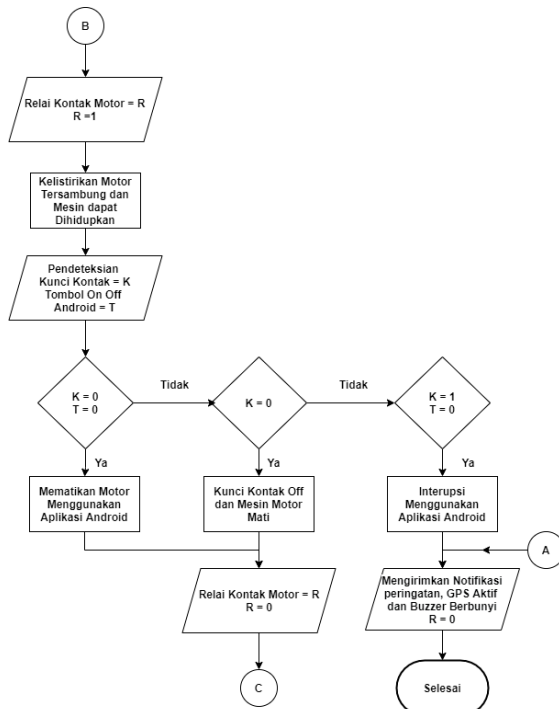
Untuk dapat menyalakan mesin motor menggunakan mode manual adalah dengan cara memutar kunci kontak pada posisi *on* lalu memverifikasi sidik jari pengguna yang sudah di daftarkan pada sistem sehingga kontak motor akan menyala. Untuk mematikan mesin motor dapat memutar kunci kontak ke posisi *off*.

Diagram alir dari perangkat lunak terdapat pada Gambar 4. Proses pertama pada diagram alir adalah menginisialisasi nilai awal dari sensor-sensor yang digunakan. Setelah nilai awal terinisialisasi, modul wifi ESP8266 menghubungkan mikrokontroler Wemos D1 dengan jaringan internet dan apabila sudah terhubung, sistem akan membaca data dari pendeteksian sensor getar, posisi kunci kontak dan kontrol *on off* dari Android. Pada kondisi ini disebut sebagai Mode Parkir. Pada proses selanjutnya, terdapat 3 kemungkinan kondisi yang terjadi. Kondisi pertama yaitu apabila kunci kontak di putar ke posisi *on* maka mikrokontroler akan mendapatkan data bahwa  $K = 1$  dan akan melanjutkan ke proses berikutnya yaitu verifikasi sidik jari sehingga proses masuk ke Mode Manual. Kondisi kedua yaitu sensor getar mendeteksi adanya getaran pada saat kunci kontak berada di posisi *off* atau  $K = 0$  maka proses berlanjut ke titik A yaitu sistem akan mengirimkan notifikasi ke Android, menyalakan GPS *live tracking*, *buzzer* pada motor berbunyi

dan relai kontak motor atau  $R = 0$  yang berarti mesin tidak dapat dinyalakan. Kondisi ketiga adalah apabila pemilik akan menyalakan kunci kontak motor menggunakan aplikasi Android maka data pembacaan  $T = 1$  dan proses berlanjut ke titik B yaitu  $R = 1$  yang artinya kelistrikan motor tersambung dan mesin dapat dinyalakan dan proses masuk ke Mode Otomatis. Pada kondisi yang pertama atau Mode Manual, sistem akan melakukan pembacaan sensor sidik jari dan akan memberikan data  $S = 1$  yang berarti sidik jari terverifikasi dibawah 10 detik dari saat nilai  $K = 1$  dan sistem akan memberikan nilai  $R = 1$  dan motor menyala. Saat Mode Manual mikrokontroler melakukan pembacaan sensor-sensor kembali, dan mendapatkan 3 kemungkinan yang terjadi saat  $R = 1$ . Kemungkinan pertama yaitu apabila kunci kontak di putar ke posisi *off* atau  $K = 0$  maka kelistrikan motor akan terputus atau  $R = 0$  dan proses kembali lagi ke titik C atau Mode Parkir. Kemungkinan kedua yaitu saat ingin mematikan motor menggunakan aplikasi Android pada saat ini  $K$  dan  $T = 0$ . Pada kondisi yang ketiga yaitu saat ingin mematikan mesin melalui aplikasi Android oleh pemilik motor pada saat menggunakan Mode Manual. Ini berarti motor sedang dibawa lari pencuri sehingga aplikasi akan mematikan relay kunci kontak  $R = 0$  dan sistem mengirimkan notifikasi peringatan, *buzzer* dan GPS *live tracking* menyala seperti pada proses A, karena dianggap sebagai pencurian ditengah jalan.





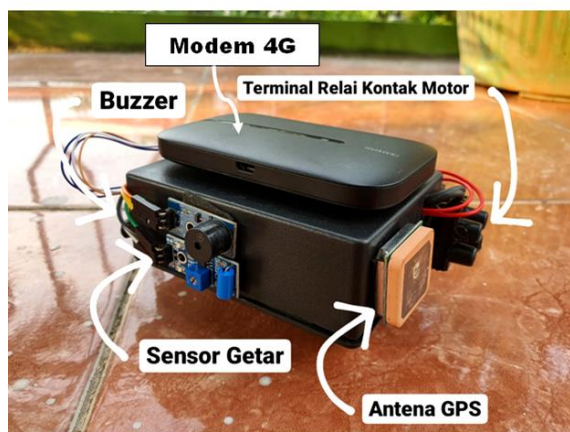


Gambar 4. Diagram Alir Perangkat Lunak

## 4. HASIL DAN PEMBAHASAN

### 4.1 Pembuatan Alat

Perancangan alat dari sistem keamanan motor ini memiliki bentuk kotak dengan ukuran  $15 \times 10 \times 8$  cm yang dimasukkan pada sistem kelistrikan motor. Alat ini juga memiliki sumber tegangan yang terpisah dari motor dan sebuah koneksi internet dengan menggunakan salah satu provider telekomunikasi telpon genggam.



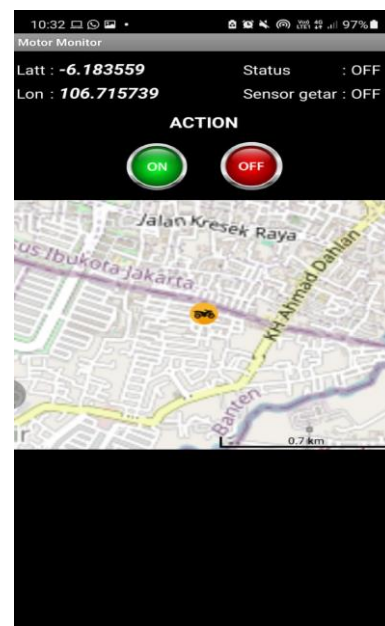
Gambar 5. Foto Alat

Gambar 5 memperlihatkan alat sistem keamanan motor dan pada Gambar 6 diperlihatkan pemasangan alat pada motor.



Gambar 6. Pemasangan Alat Pada Motor

Pemrograman pada mikrokontroler dibuat dengan *basic C++* yang ditulis pada perangkat lunak Arduino IDE. Komunikasi antara mikrokontroler dengan *interface* pada *smartphone* menggunakan kanal *cloud database* dari fitur yang disediakan oleh ThingSpeak. *Interface* pada aplikasi Android ditulis menggunakan MIT App Inventor. Gambar 7 memperlihatkan tampilan aplikasi Android pada *smartphone*.



Gambar 7. Tampilan Aplikasi Android

## 4.2 Pengoperasian Alat

Cara mengoperasikan alat sistem keamanan adalah sebagai berikut:

1. Menghubungkan alat ke sumber tegangan.
2. Membuka aplikasi Android yang telah dibuat dan diinstall pada *smartphone*.
3. Memeriksa koneksi internet dengan melihat data koordinat GPS pada aplikasi Android. Apabila sudah muncul data maka alat dan sistem keamanan sudah terhubung dengan internet.

Cara tersebut dilakukan hanya saat pertama kali alat dipasang pada motor.

### 4.2.1 Pengoperasian Mode Parkir

Pada saat motor hendak diparkir dan aplikasi pengaman hendak digunakan, pemilik motor harus sudah menyalakan sistem pengaman di motor dan menyalakan *smartphone*, maka mode parkir akan bekerja.

### 4.2.2 Pengoperasian Sistem Menggunakan Aplikasi Android Sebagai Mode Otomatis

Dalam keadaan tertentu apabila sensor sidik jari tidak dapat memverifikasi pengguna, maka pengguna dapat menyalakan kontak motor menggunakan aplikasi Android dengan langkah-langkah pengoperasiannya sebagai berikut:

1. Untuk menyalakan kontak motor adalah dengan membuka aplikasi Android pada *smartphone* dan menekan tombol *on*.
2. Untuk mematikan kontak motor dengan menekan tombol *off* seperti yang terdapat pada Gambar 7.

### 4.2.3 Pengoperasian Sistem Menggunakan Kunci Kontak Sebagai Mode Manual

Untuk penggunaan motor dalam mode manual, pengguna tidak perlu membuka aplikasi yang diinstall pada Android. Cara pengoperasiannya adalah:

1. Memutar kunci kontak pada posisi *on*.
2. Melakukan verifikasi dengan menggunakan sensor sidik jari.
3. Kontak motor menyala, dan motor dapat dijalankan.
4. Cara mematikan kontak motornya adalah dengan memutar kontak motor pada posisi *off*.

## 4.3 Pengujian Alat

Pada pengujian ini dilakukan untuk mengetahui alat dan sistem yang dibuat telah berjalan sesuai dengan perancangan.

### 4.3.1 Pengujian Konektifitas

Pengujian ini bertujuan untuk mengetahui waktu yang dibutuhkan dari pengiriman data yang diterima dari sensor hingga data itu sampai ke aplikasi Android.

### Pengujian Pengiriman Data dari Alat ke Cloud Thingspeak

Pengujian ini dilakukan dengan cara menyalakan dan mematikan kunci kontak motor kemudian melihat grafik *private update* pada kanal *database* ThingSpeak berupa data *latitude*, *longitude*, status kelistrikan motor, status sensor getar dan waktu responnya yang hasilnya terdapat pada Tabel 1. Untuk status kelistrikan motor dan sensor getar akan tampil tulisan ON apabila sedang aktif atau OFF apabila sedang tidak aktif. Data ini dilihat pada layar *smartphone* melalui *provider* telepon genggam Smartfren sebagai jaringan internet.

TABEL 1. HASIL PENGUJIAN MENGIRIM DATA DARI ALAT KE CLOUD THINGSPEAK

Pengujian Ke-	Latitude	Longitude	Waktu Respon (Detik)
1	-6.183559	106.715739	13,55
2	-6.183559	106.715739	14,1
3	-6.183579	106.715704	10,05
4	-6.183570	106.715660	8,1
5	-6.183564	106.715629	8,45
6	-6.183555	106.715601	11,5
7	-6.183567	106.715629	9,34
8	-6.183572	106.715656	10,27
9	-6.183577	106.715684	11,21
10	-6.183583	106.715712	8,44

Pada Tabel 1 data *latitude* dan *longitude* adalah data yang diterima pada kanal *database* ThingSpeak yang dikirimkan oleh mikrokontroler dari hasil pembacaan modul GPS.

Berdasarkan data hasil pengujian pada Tabel 1 didapatkan hasil rata-rata waktu respon pengiriman data dari alat ke *cloud* ThingSpeak selama 10,5 detik menggunakan rumus *arithmetic mean* sebagai berikut :

$$t = \frac{\sum ti}{N} = \frac{105}{10} = 10,5 \text{ detik}$$

- t = Rata-rata waktu respon
- $\sum ti$  = Penjumlahan waktu respon dari semua data
- N = Jumlah data pengamatan

#### Pengujian Pengambilan Data dari Thingspeak ke Android

Pengujian ini dilakukan dengan cara melihat aplikasi Android di layar *smartphone* yang dibuat dengan nama Motor Monitor dimana terdapat tampilan navigasi yang berfungsi untuk memonitor posisi motor dengan memanfaatkan Google Map API. Posisi motor pada tampilan navigasi adalah hasil konversi data *latitude* dan *longitude* yang diambil dari kanal *cloud database* ThingSpeak ke aplikasi Android. Status kelistrikan motor dan sensor getar yang diupdate dari data terakhir yang ditampung pada *cloud database* ThingSpeak juga dapat dilihat pada tampilan layar *smartphone* seperti yang terdapat pada Gambar 7. Pengujian dilakukan di daerah Pulo Indah Asri Jakarta Barat menggunakan jaringan internet telpon genggam Smartfren dengan kecepatan 5 MBps dan ping sebesar 33ms.

TABEL 1. PENGUJIAN PENGAMBILAN DATA DARI THINGSPEAK KE SMARTPHONE

Pengujian Ke-	Latitude	Longitude	Waktu Respon (Detik)
1	-6.183559	106.715739	6,33
2	-6.183559	106.715739	5,03
3	-6.183579	106.715704	8,12
4	-6.183570	106.715660	7,44
5	-6.183564	106.715629	6,53
6	-6.183555	106.715601	7,37
7	-6.183567	106.715629	8,24
8	-6.183572	106.715656	7,53
9	-6.183577	106.715684	6,48
10	-6.183583	106.715712	8,32

Berdasarkan data pada Tabel 2 diketahui tampilan data *latitude* dan *longitude* di layar *smartphone* berhasil dikirimkan dari kanal ThingSpeak dengan waktu respon rata-rata sebesar 7,13 detik dihitung dari saat kanal ThingSpeak menerima data sampai data itu diterima pada aplikasi Android.

Waktu *delay* rata-rata dari alat sampai ke aplikasi Android untuk mengirim data *longitude* dan *latitude* motor, status kelistrikan, dan status sensor gerak adalah 17,63 detik.

#### 4.3.2 Pengujian Sistem Keamanan

##### Pengujian Mode Parkir

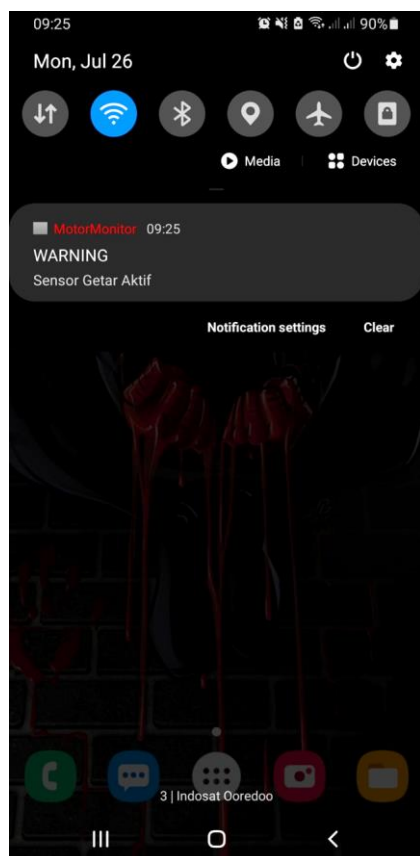
Pengujian ini bertujuan untuk mengetahui respon *output* yang dihasilkan oleh sistem yang dikirimkan pada notifikasi Android dan *buzzer*. Pengujian dilakukan dengan cara melihat respon dari pendeteksian sensor getar pada saat motor diparkir dan motor digerak-gerakan.

TABEL 2. PENGUJIAN MODE PARKIR

Pengujian Ke-	Notifikasi Android	Delay (Detik)	Buzzer
1	Aktif	6,64	Aktif
2	Aktif	6,80	Aktif
3	Aktif	7,09	Aktif
4	Aktif	7,33	Aktif
5	Aktif	6,87	Aktif
6	Aktif	7,37	Aktif
7	Aktif	8,11	Aktif
8	Aktif	7,41	Aktif
9	Aktif	6,58	Aktif
10	Aktif	7,42	Aktif

Tabel 3 memperlihatkan hasil pengujian sensor SW 420 yang mendeteksi gerakan yang diberikan pada motor sehingga *buzzer* aktif dan direkam waktu responnya. Sistem ini membutuhkan waktu untuk mengirimkan notifikasi pada aplikasi Motor Monitor dengan *delay* waktu rata-rata selama 7,16 detik. Walaupun terdapat *delay* pada pengiriman data ke aplikasi Android namun sistem keamanan tetap bekerja dengan baik karena pengaturan relay kunci kontak motor yang berada pada pemrograman mikrokontroler dibuat *off*, sehingga mesin motor tidak dapat dinyalakan. Pada Gambar 8 dapat dilihat notifikasi yang muncul di aplikasi Android pada saat sensor getar mendeteksi adanya getaran pada saat kunci kontak berada di posisi *off*.





Gambar 8. Notifikasi Sensor Getar Aktif

### Pengujian Mode Manual

Pengujian ini bertujuan untuk melihat respon dari sistem mode manual yang telah dibuat dengan cara menyalakan motor menggunakan kunci kontak dan melakukan verifikasi sidik jari pada saat motor berada dalam mode parkir. Pengujian dilakukan dengan verifikasi jari yang berhasil dan tidak berhasil. Hasil pengujian terdapat pada Tabel 4 dan Tabel 5.

TABEL 3. PENGUJIAN MODE MANUAL YANG BERHASIL

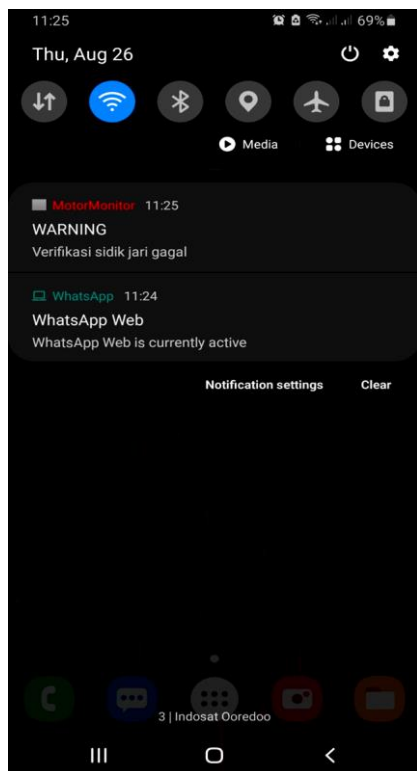
Pengujian Ke-	Verifikasi Pengguna	Delay (Detik)	Kontak Motor
1	Berhasil	4,30	Menyala
2	Berhasil	5,22	Menyala
3	Berhasil	5,35	Menyala
4	Berhasil	4,52	Menyala
5	Berhasil	5,15	Menyala
6	Berhasil	4,01	Menyala
7	Berhasil	6,10	Menyala
8	Berhasil	3,55	Menyala
9	Berhasil	4,21	Menyala
10	Berhasil	5,32	Menyala

Pengujian dilakukan dengan memutar kontak ke posisi *on* dan memverifikasi sidik jari pengguna yang sudah didaftarkan sebelumnya. Dari hasil pengujian dapat dilihat sensor sidik jari berhasil mendeteksi sidik jari dan membutuhkan waktu *delay* rata-rata dihitung dari saat menempelkan jari sampai relai kontak motor tersambung selama 4,77 detik.

TABEL 4. PENGUJIAN MODE MANUAL YANG TIDAK BERHASIL

Pengujian Ke-	Kontak Motor	Buzzer	Notif. Android	Delay Notif. (Detik)
1	Mati	Aktif	Verifikasi sidik jari gagal	15,30
2	Mati	Aktif	Verifikasi sidik jari gagal	17,29
3	Mati	Aktif	Verifikasi sidik jari gagal	16,10
4	Mati	Aktif	Verifikasi sidik jari gagal	14,50
5	Mati	Aktif	Verifikasi sidik jari gagal	15,05
6	Mati	Aktif	Verifikasi sidik jari gagal	17,01
7	Mati	Aktif	Verifikasi sidik jari gagal	15,30
8	Mati	Aktif	Verifikasi sidik jari gagal	14,20
9	Mati	Aktif	Verifikasi sidik jari gagal	13,55
10	Mati	Aktif	Verifikasi sidik jari gagal	15,36

Pada Tabel 5 hasil pengujian diatas dilakukan tanpa melakukan verifikasi sidik jari pengguna. Dalam rancangan sistem yang dibuat, *buzzer* akan menyala setelah 10 detik dari waktu diputarnya kunci kontak motor ke posisi *on* dan sistem akan mengirimkan notifikasi pada *smartphone*. Pengukuran waktu *delay* dihitung pada saat *buzzer* mulai berbunyi sampai dengan munculnya notifikasi pada aplikasi Android. Nilai rata-rata *delay* pada pengujian ini adalah selama 15,36 detik. Walaupun terdapat *delay* yang cukup lama namun mesin motor tidak dapat dinyalakan karena relai kontak motor dikontrol oleh mikrokontroler dan *buzzer* pada motor akan berbunyi seketika setelah sidik jari gagal terverifikasi. Tampilan layar *smartphone* untuk pengujian mode manual yang tidak berhasil dapat dilihat pada Gambar 9.



Gambar 9. Notifikasi verifikasi sidik jari gagal

### Pengujian Mode Otomatis

Pengujian ini dilakukan dengan cara menekan tombol yang ada pada aplikasi Android yaitu tombol *on* dan *off* dan melihat kondisi kontak kelistrikan motor berdasarkan *action* yang dilakukan.

TABEL 5. PENGUJIAN MODE OTOMATIS

Pengujian Ke-	Tombol yang ditekan	Keadaan kontak motor	Waktu respon (Detik)
1	On	Nyala	8,22
2	Off	Mati	12,20
3	On	Nyala	11,34
4	Off	Mati	9,55
5	On	Nyala	9,57
6	Off	Mati	1,21
7	On	Nyala	11,10
8	Off	Mati	9,59
9	On	Nyala	10,10
10	Off	Mati	11,23

Dari data hasil pengujian pada Tabel 6 didapatkan hasil rata-rata waktu respon pada aplikasi Android dari saat tombol *on* atau *off* ditekan sampai motor menyala atau mati adalah selama 10,51 detik.

## 5. KESIMPULAN DAN SARAN

Berdasarkan hasil perancangan, pembuatan dan pengujian sistem keamanan motor ini didapatkan beberapa kesimpulan :

1. Waktu respon rata-rata dari alat pengaman motor untuk mengirimkan data sampai menghasilkan tampilan aplikasi Android adalah 17,63 detik. Walaupun dibutuhkan waktu memperbaharui data sekitar 17,63 detik, namun motor tetap aman karena kunci kontak diatur *off* oleh program.
2. Waktu respon rata-rata yang dibutuhkan alat untuk mengirimkan notifikasi peringatan sensor getar aktif pada aplikasi Android saat mendeteksi adanya gerakan pada motor pada Mode Parkir sebesar 7,16 detik. Pada mode parkir ini motor tidak dapat dinyalakan karena program mengatur kunci kontak dalam keadaan *off*.
3. Waktu respon rata-rata untuk menyalakan motor menggunakan sensor sidik jari pada Mode Manual dari waktu menempelkan sidik jari pada area sensor sampai motor menyala adalah 4,77 detik. Pada proses ini memakan waktu yang lebih sedikit dikarenakan prosesnya berada pada mikrokontroler tanpa melewati kanal *cloud database* ThingSpeak. Sedangkan waktu respon rata-rata apabila menggunakan mode manual tetapi tidak berhasil memverifikasi sidik jari adalah 15,36 detik. Dalam keadaan ini motor tidak dapat dinyalakan karena program mengatur kunci kontak motor dalam kondisi *off*.
4. Waktu respon rata-rata untuk menyalakan motor menggunakan aplikasi Android pada Mode Otomatis sebesar 10,51 detik.
5. Waktu respon dipengaruhi oleh banyak hal diantaranya yaitu jaringan koneksi internet yang tidak stabil, *server cloud database* yang lambat, *wiring* pada perangkat keras yang kendor, maupun dari mikrokontroler yang tidak merespon. Namun waktu respon dari sistem dan kinerja alat sudah baik untuk skala pengujian.

Saran yang dapat dilakukan untuk meningkatkan kinerja dari alat sistem keamanan motor ini :

1. Menggunakan *server cloud database* yang memiliki *ping* atau *latency* respon yang lebih cepat. Karena ThingSpeak yang digunakan

penulis disini bersifat *free* maka ketersediaan fitur nya juga minim dan respon lambat.

2. Menggunakan jaringan internet telepon genggam yang lebih stabil.

#### Daftar Pustaka:

- [1] D. P. Sari, "Peningkatan Jumlah Kendaraan Bermotor di DKI Jakarta - Unit Pengelola Statistik," *Unit Pengelola Statistik DKI Jakarta*, 2020. Online. <https://statistik.jakarta.go.id/peningkatan-jumlah-kendaraan-bermotor-di-dki-jakarta/> (Accessed Aug. 25, 2021).
- [2] Edi Wahyono, "Dalam Seminggu Ada 177 Kasus Pencurian Motor!," *DetikOto*, 2020. Online. <https://oto.detik.com/motor/d-5075692/dalam-seminggu-ada-177-kasus-pencurian-motor> (Accessed Aug. 25, 2021).
- [3] Dio Dananjaya, "Curanmor Meningkatkan saat New Normal, Polisi Imbau Kunci Ganda," *Kompas.com*, 2020. Online. <https://otomotif.kompas.com/read/2020/06/18/094200415/curanmor-meningkat-saat-new-normal-polisi-imbau-kunci-ganda> (accessed Aug. 25, 2021).
- [4] D. indra prasetya and M. Mushlihudin, "Sistem Keamanan Sepeda Motor Menggunakan Kata Sandi Berbasis Arduino Nano," *J. Ilm. Tek. Elektro Komput. dan Inform.*, vol. 4, no. 1, pp. 11–19, Jun. 2018, doi: 10.26555/JITEKI.V4I1.8985.
- [5] S. Tjhin, M. Amami, M. T. Ahmad, and A. Faqih, "Sistem Keamanan Sepeda Motor Melalui Short Message Service Menggunakan AVR Mikrokontroler ATMEGA8," *Semin. Nas. Teknol. Inf. dan Komun.*, 2014.
- [6] Y. P. Putra and E. Edidas, "Pengembangan Sistem Keamanan Sepeda Motor Menggunakan Arduino Uno Berbasis Smartphone Android," *Voteteknika (Vocational Tek. Elektron. dan Inform.)*, vol. 8, no. 1, pp. 106–115, Mar. 2020, doi: 10.24036/VOTETEKNIKA.V8I1.107779.
- [7] D. Pratama, E. D. Febrianto, D. A. Hakim, T. Mulyadi, R. W. Halfiah, and U. Fadlilah, "Sistem Keamanan Ganda pada Sepeda Motor untuk Pencegahan Pencurian dengan SMARTY (Smart Security)," *J. Ilmu Komput. dan Inform.*, vol. 3, no. 1, p. 31, Jul. 2017, doi: 10.23917/KHIF.V3I1.4205.
- [8] Ari Nugroho, "Sistem Keamanan Sepeda Motor Menggunakan Arduino Berbasis Android," *J. Ilm. Tek. Elektro*, vol.1, no.1, 2017.
- [9] Resmiaini, A. Sulisty, and R. Umar, "APLIKASI GPS MOBILE UNTUK PEMODELAN AREA RAWAN DEMAM BERDARAH DENGUE," *J. Inform. Rekayasa Elektron.*, vol. 4, no. 1, pp. 51–29, 2021, doi: <https://doi.org/10.36595/jire.v4i1.291>.
- [10] M. L. Hakim, W. Bagye, H. Fahmi, and K. Imtihan, "PEMANFAATAN TEKNOLOGI GOOGLE MAPS API UNTUK APLIKASI PENDETAKSIAN LOKASI RAWAN KRIMINALITAS BERBASIS ANDROID KECAMATAN PRAYA TIMUR, KABUPATEN LOMBOK TENGAH," *J. Inform. Rekayasa Elektron.*, vol. 2, no. 1, pp. 52–59, 2019, doi: <https://doi.org/10.36595/jire.v2i1.90>.