



Sistem Keamanan Helm Berbasis *Internet of Things* dengan Fitur Pelacakan Menggunakan Android

Panji Wiratama Santoso¹, I Nyoman Piarsa², Ni Made Ika Marini Mandenni³
^{1,2,3}Program Studi Teknologi Informasi, Fakultas Teknik, Universitas Udayana
¹panjisantoso20@gmail.com, ²manpits@unud.ac.id, ³made_ikamarini@unud.ac.id

Abstract

Helmet theft is a problem that is of concern to the public. Information Technology provides may resolve any problems, like Internet of Things-based system that can detect helmet theft and track the location of the helmet if it's stolen. This system is designed using a microcontroller device, namely Arduino which is attached to the helmet and motorbike with the help of the SIM800L v2, GPS Neo-6m, buzzer, and Bluetooth HC-05 which is connected to the master slave as an indicator of the safety of the helmet. The hardware on the helmet is connected to Firebase Realtime Database server so it can be connected with the user's Android application to monitor the state and location of the helmet. Android application displays maps to determine the position of the helmet, and can display notifications when the helmet is being stolen. The conclusion is this system can detect helmet theft with a maximum distance from the master and slave bluetooth connections of 10 meters, and the average data transmission from hardware to Firebase is 1,1 seconds, and can monitor status of the helmet and track the position of the helmet through the Android application with Android Jelly Bean (v4.3) operating system.

Keywords: Helmet theft, Internet of Things, Firebase, Android

Abstrak

Pencurian helm merupakan salah satu masalah yang menjadi perhatian di kalangan masyarakat saat ini. Pemanfaatan Teknologi Informasi memberikan banyak kemudahan untuk mengatasi banyak permasalahan, seperti dengan menggunakan sistem berbasis *Internet of Things* yang dapat mendeteksi pencurian helm dan melacak lokasi helm jika tercuri. Sistem keamanan helm ini dirancang menggunakan perangkat mikrokontroler yaitu, Arduino yang dipasangkan pada helm dan motor dengan bantuan modul SIM800L v2, GPS Neo-6m, buzzer, dan Bluetooth HC-05 yang dikoneksikan *master slave* sebagai indikator keamanan pada helm. Perangkat keras pada helm dihubungkan dengan server *Firebase Realtime Database* sehingga dapat dihubungkan dengan aplikasi Android pengguna untuk memonitor keadaan dan lokasi dari helm. Aplikasi Android menampilkan *maps* untuk mengetahui posisi helm, dan dapat menampilkan notifikasi apabila helm sedang dalam keadaan tercuri. Kesimpulan dari hasil perancangan sistem keamanan helm yaitu, rancangan sistem pendeteksi helm berbasis IoT dapat mendeteksi pencurian helm dengan jarak maksimal dari koneksi *bluetooth master* dan *slave* adalah 10 meter, dan rata-rata pengiriman data dari perangkat keras ke *Firebase* adalah 1,1 detik, serta dapat melakukan *monitoring* status helm dan pelacakan posisi helm melalui aplikasi Android dengan sistem operasi Android *Jelly Bean*(v4.3).

Kata kunci: Pencurian helm, Internet of Things, Firebase, Android

1. Pendahuluan

Sepeda motor merupakan kendaraan yang pada umumnya paling banyak digunakan mahasiswa, kendaraan tersebut di parkir pada tempat yang disediakan pada kampus. Kebanyakan dari tempat parkir kampus tidak menyediakan fasilitas penitipan helm, dan petugas yang menjaga tempat parkir tidak dapat menjaga atau mengawasi setiap saat, sehingga mahasiswa menyimpan helm pada sepeda motor mereka [1]. Peristiwa kehilangan helm terjadi karena kelalaian mahasiswa yang tidak mengunci helm pada sepeda

motor dengan aman. Terdapat kemungkinan terjadi kehilangan helm meskipun sudah dikunci pada sepeda motor.

Perkembangan teknologi saat ini telah memberikan pengaruh yang besar untuk mengatasi berbagai masalah dalam kehidupan sehari-hari, seperti di bidang Teknologi Informasi. Era industri 4.0 ini erat kaitannya dengan automasi kebutuhan yang difasilitasi dengan teknologi, seperti IoT (*Internet of Things*). *Internet of Things* adalah suatu konsep yang digunakan dalam memperluas pemanfaatan dari jaringan internet yang

berkembang cukup pesat mulai dari MEMS atau *micro-electromechanical systems*, konvergensi dari teknologi nirkabel, dan Internet [2]. *Internet of Things* dapat melakukan pengiriman maupun membaca data, melakukan *remote control*, dan lain sebagainya. Contohnya yaitu untuk benda dalam kehidupan sehari-hari seperti bahan pangan, elektronik, dan termasuk benda hidup yang dapat tersambung ke jaringan global dan juga lokal dengan menggunakan sensor yang dirancang dan dapat diaktifkan. IoT dapat mengarah pada suatu benda yang secara unik dapat diidentifikasi sebagai suatu representasi *virtual* yang berbasis internet.

Beberapa penelitian yang meneliti mengenai revolusi industri 4.0 dengan memanfaatkan teknologi Internet of Things dalam hal keamanan khususnya keamanan helm dengan perangkat *Internet of Things*, diantaranya yaitu [1] penerapan sistem untuk mencegah terjadinya peristiwa pencurian helm dengan menggunakan sensor magnetic reed switch sebagai pengunci helm, dan menggunakan Bluetooth HC-05 sebagai koneksi master dan slave sebagai indikator jarak antara helm dengan sepeda motor, sehingga dalam penelitian tersebut dapat menghasilkan suatu sistem yang dapat mendeteksi pencurian helm menggunakan protokol MQTT dan modul pendukung sistem berbasis Arduino. Penelitian lainnya [3] merancang sistem yang dapat melacak kendaraan menggunakan Modul *Relay*, SIM808, dan Arduino Uno dengan mengukur keakuratan dalam mengunci suatu sinyal satelit dan mengukur lama dari waktu yang dibutuhkan untuk menerima dan mengirimkan pesan singkat yaitu informasi posisi atau lokasi kendaraan pengguna dan memutus sistem kelistrikan sepeda motor [3], penelitian lainnya [4] berupa perancangan sistem untuk perancangan sistem keamanan sepeda motor sebagai panduan untuk perancangan sistem keamanan sepeda motor berbasis Arduino-Android yang dapat dikendalikan melalui *smartphone* dan dirancang menggunakan modul Bluetooth HC-06 yang dapat diintegrasikan dengan perangkat mikrokontroler Arduino Uno [4]. Penelitian yang telah dikaji sebelumnya terdapat beberapa kelemahan, yaitu seperti tidak terdapat suatu sistem yang dapat melacak keberadaan alat tersebut apabila terjadi pencurian, sehingga apabila terjadi pencurian nantinya pengguna tidak mengetahui posisi helm.

Berhubungan dengan IoT (*Internet of Things*) dan sistem pendeteksi ini serta melatarbelakangi masalah yang ada, dapat dilakukan penelitian dalam merancang perangkat sistem yang dapat mendeteksi terjadinya pencurian helm dan mengetahui lokasi atau posisi dari helm yang telah dicuri dengan menggunakan sistem pendeteksi berbasis teknologi *Internet of Things*. Perancangan sistem ini dilakukan dengan menggunakan modul SIM800L yang dapat melakukan koneksi jaringan internet sehingga dapat mengirimkan data dari sistem ke aplikasi Android dengan menggunakan Firebase. Selain itu, sistem ini menggunakan modul Bluetooth HC-05 sebagai koneksi

master (perangkat pada motor) dan *slave* (perangkat pada helm). Penggunaan koneksi *master* dan *slave* bertujuan sebagai indikator jarak antara helm dengan sepeda motor. Sistem dirancang dengan modul GPS NEO-6M yang digunakan pada perangkat helm untuk mengetahui lokasi dari helm, dan juga *buzzer* yang dapat digunakan sebagai alarm. Semua modul yang sudah dijelaskan sebelumnya, digunakan sebagai sumber data dalam perancangan sistem ini, sehingga dapat menghindari terjadinya pencurian helm dan dapat mengurangi kerugian bagi pemilik helm yang mengalami kejadian kehilangan atau pencurian helm.

Penelitian ini bertujuan untuk melakukan perancangan dan implementasi IoT (*Internet of Things*) untuk mendeteksi terjadinya pencurian helm, dan dapat melakukan integrasi sistem Android dengan perangkat keras IoT menggunakan *Firebase*. Perangkat keras pada motor dan helm dirancang dengan menggunakan modul Bluetooth HC-05 yang dapat membantu dalam perancangan sistem penelitian, karena modul tersebut dapat melakukan konfigurasi *master* dan *slave* yang digunakan untuk indikator jarak keamanan pada perangkat keras helm dan motor, yang memiliki dua mode dalam modul tersebut yaitu *Data Mode* dan juga *AT Command Mode* [5]. Perangkat keras pada motor menggunakan Arduino dan baterai 9V sebagai daya pada Arduino, didasarkan karena perangkat keras motor hanya menggunakan satu modul yang tidak menggunakan tegangan tinggi dan arus pada baterai, sehingga dapat lebih efisien dalam perancangan perangkat keras pada motor.

Perangkat keras pada helm menggunakan Arduino UNO sebagai perangkat *mikrocontroller*, karena memiliki kelengkapan *pin* dan *port* lebih lengkap dibandingkan Arduino Nano, dan modul SIM800L v2 dapat berjalan dengan stabil menggunakan Arduino UNO. Modul GPS NEO-6M yang digunakan untuk mengetahui posisi helm, dapat melacak hingga 22 satelit di 50 saluran dengan mengonsumsi arus 45mA [6]. Modul SIM800L v2 dapat digunakan untuk melakukan pengiriman dan menerima data menggunakan layanan pesan singkat atau SMS, dan koneksi jaringan internet GPRS menggunakan perintah AT [7]. Sehingga modul SIM800L v2 dapat mengirimkan data yang didapatkan dari tiap modul ke *Firebase Realtime Database* yang merupakan *database realtime* dan dapat menyimpan data pada *cloud*. *Firebase Realtime Database* memiliki dukungan berbagai *platform* seperti Android maupun *website* [8].

Data yang telah dikirimkan pada *Firebase* akan dibaca pada aplikasi Android, yang merupakan sistem operasi pada perangkat telepon pintar (*smartphone*) berbasis Linux, dan menyediakan berbagai macam *platform* terbuka untuk para pengembang agar dapat menciptakan suatu aplikasi [9]. *Firebase Realtime Database* dapat memungkinkan untuk menyimpan dan sinkronisasi data secara *realtime* dan data tersebut akan secara otomatis

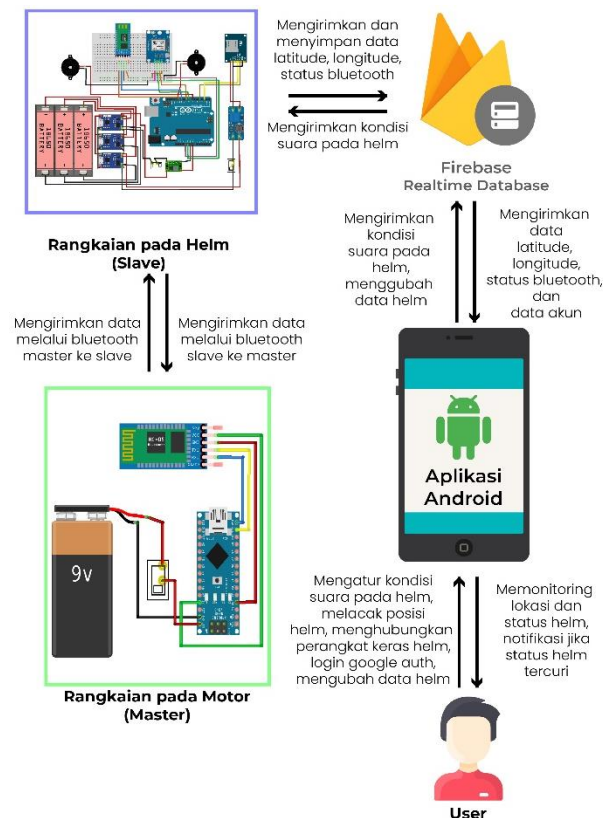
melakukan pembaruan dengan data terbaru ke pengguna [10], [11].

2. Metode Penelitian

Penelitian ini dilakukan melalui proses yang diawali dari melalui studi literatur, melakukan perancangan dan pembuatan alat, melakukan pengumpulan dan pengambilan data. Metode penelitian dari sistem pendeteksi pencurian helm berbasis IoT dijelaskan sebagai berikut.

2.1. Gambaran Umum Sistem

Gambaran umum bertujuan untuk menggambarkan alur dari sistem yang dirancang seperti pada Gambar 1.



Gambar 1. Gambaran Umum Sistem

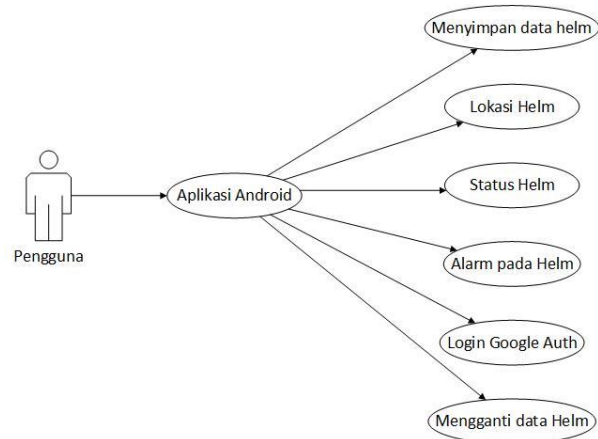
Alur proses sistem dimulai dari rangkaian motor (*master*) akan melakukan koneksi *master slave* menggunakan modul *Bluetooth* HC-05 ke rangkaian pada helm (*slave*). *Buzzer* pada perangkat motor akan berbunyi jika koneksi *master* dan *slave* tersebut terputus. Rangkaian pada motor akan mengirimkan data *latitude* & *longitude* pada modul GPS Neo-6m, dan data *bluetooth*. Data tersebut dikirimkan ke *Firebase* menggunakan koneksi dari SIM800L v2. Semua data yang telah disimpan pada *Firebase*, akan dibaca pada aplikasi Android pengguna, dan pengguna dapat melakukan *monitoring* lokasi dari helm tersebut.

Pengguna juga dapat mengatur kondisi suara pada helm yang digunakan untuk mengetahui lokasi helm jika sudah mendekati posisi helm. Pengguna juga dapat

mengatur kondisi alarm pada helm yang digunakan untuk mengetahui lokasi helm jika sudah mendekati posisi helm, dan pengguna dapat menekan tombol *switch* pada aplikasi Android dan akan mengubah data *alarm_helm* pada *Firebase*. Alarm akan berbunyi apabila data *alarm_helm* adalah “1”, dan jika “0” maka alarm helm akan mati, dan aplikasi Android akan menampilkan data helm ke pengguna. Aplikasi android dapat melakukan koneksi *bluetooth* ke perangkat keras helm (*slave*) untuk menyimpan data helm tersebut ke akun pengguna yang telah melakukan *login* menggunakan autentikasi Google dan mengirimkan data helm *key* baru. Perangkat keras pada helm akan mengirimkan data yaitu nama *bluetooth* yang selanjutnya akan disimpan pada *Firebase Realtime Database*.

2.2. Diagram Use Case

Use Case Diagram bertujuan untuk mengetahui fungsi yang terdapat dalam sistem dan dapat merepresentasikan interaksi antara aktor dengan sistem. Diagram *use case* penelitian ini dapat dilihat seperti pada Gambar 2.



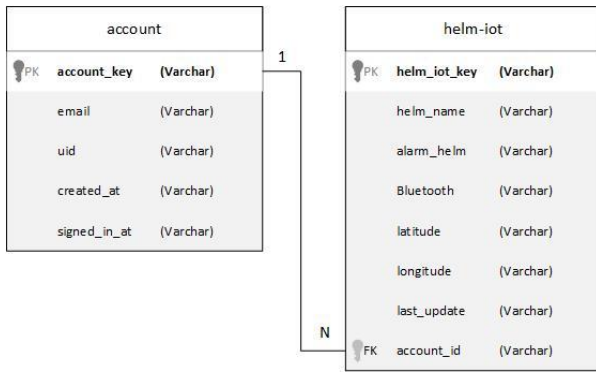
Gambar 2. Diagram Use Case

Pengguna dapat menggunakan aplikasi *mobile* berbasis Android dan dapat mengakses lokasi helm, status helm, dan suara pada *buzzer* di helm, yang telah dihubungkan dengan setiap perangkat pendukung yang ada pada sistem seperti modul GPS Neo-6m, *Bluetooth* HC-05, SIM800L, dan *buzzer*. Pengguna dapat melihat lokasi helm melalui perangkat modul GPS pada helm seperti *latitude* dan *longitude*, status helm didapatkan melalui koneksi antara *master* dan *slave* pada modul *Bluetooth* HC-05, dan suara pada *buzzer* dapat dihidupkan dan dimatikan melalui aplikasi Android. Pengguna dapat menyimpan dan mengubah data helm pada *Firebase Realtime Database*, dan dapat melakukan *login* menggunakan autentikasi Google.

2.3. Database Sistem

Perancangan *database* dalam Sistem Pendeteksi Pencurian Helm diperlukan karena dalam suatu aplikasi diperlukan basis data yang dapat menyimpan data di dalam aplikasi tersebut. Perancangan *database* sistem

dapat dilihat seperti pada Gambar 3.

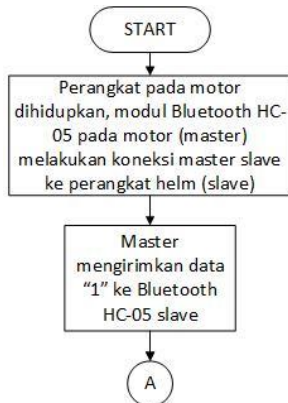


Gambar 3. Rancangan Database Sistem

Database sistem dirancang dalam *Firestore Realtime Database* dengan nama *database helm-iot-test-default-rtdb*, yang memiliki dua tabel yaitu *helm-iot* dan *account*. Tabel *account* bertujuan untuk menyimpan data akun yang telah melakukan *sign in* pada aplikasi Android pengguna, tabel *account* memiliki 5 atribut yang merepresentasikan tabel tersebut antara lain yaitu, *account_key*, *email*, *uid*, *created_at*, dan *signed_in_at*. Tabel *helm-iot* bertujuan untuk menyimpan data perangkat keras helm yang telah didaftarkan melalui aplikasi android pengguna, tabel *helm-iot* memiliki 8 atribut yang merepresentasikan tabel tersebut antara lain, *helm_iot_key*, *helm_name*, *alarm_helm*, *bluetooth*, *last_update*, *latitude*, *longitude*, dan *account_id* yang memiliki fungsi yang berbeda-beda.

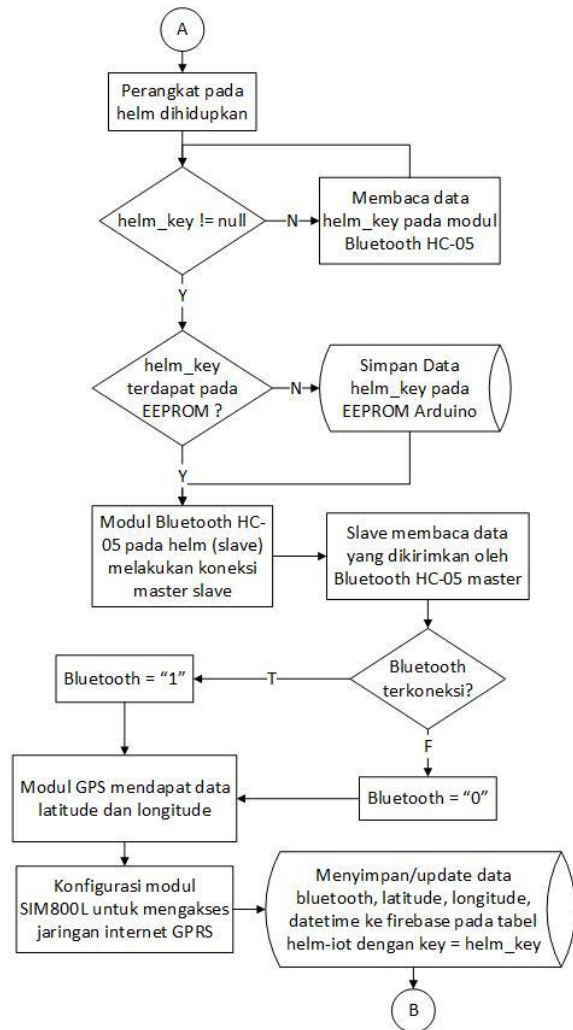
2.4. Flowchart Sistem

Flowchart rancangan sistem dari penelitian ini dapat dilihat seperti pada Gambar 4.



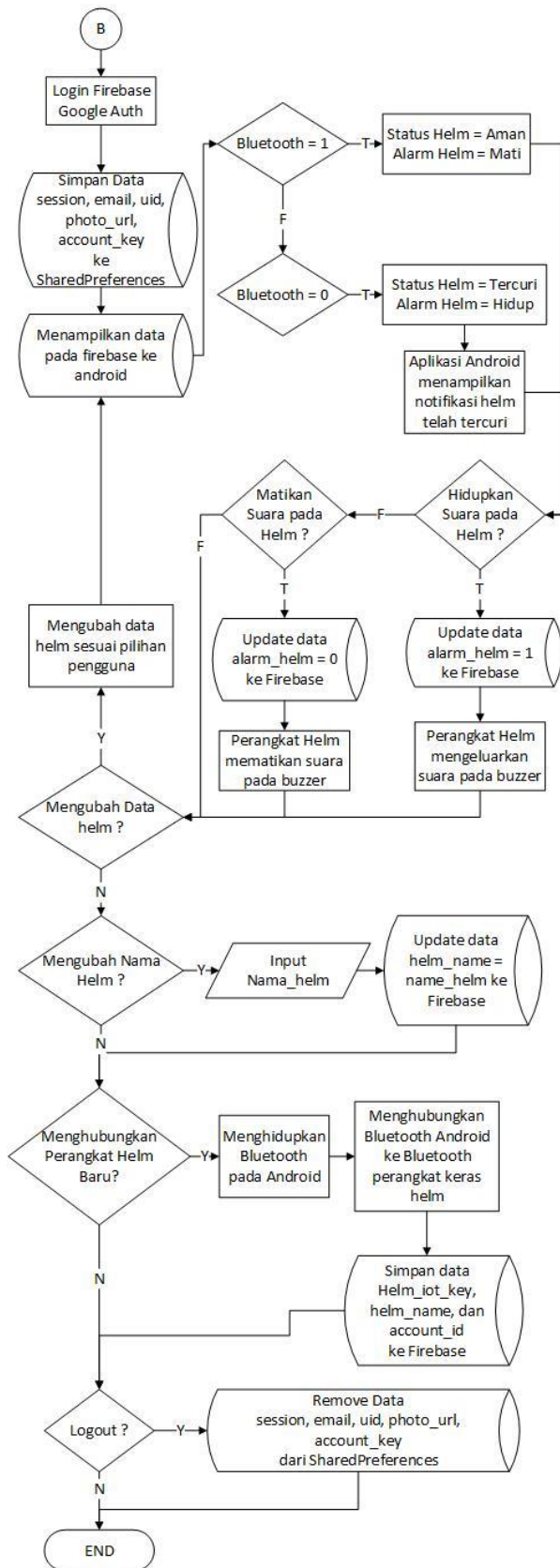
Gambar 4. Flowchart Rancangan Sistem pada Perangkat Motor

Alur flowchart dimulai dari perangkat pada motor dihidupkan, modul *Bluetooth* HC-05 pada motor (*master*) melakukan koneksi *master slave* ke perangkat motor (*slave*), dan *master* mengirimkan data “1” ke *slave*. Alur flowchart dari perancangan sistem pada perangkat helm dapat dilihat seperti pada Gambar 5.



Gambar 5. Flowchart Rancangan Sistem pada Perangkat Helm

Alur flowchart dimulai dari perangkat keras helm dihidupkan, kemudian akan membaca data *helm_key* apabila tersedia maka akan menyimpan data tersebut pada EEPROM yang terdapat pada Arduino. Modul *Bluetooth* HC-05 akan membaca kembali data *helm_key* apabila tidak menerima data atau data tidak tersedia. Selanjutnya modul *Bluetooth* HC-05 pada helm (*slave*) melakukan koneksi *master slave*. Kemudian *slave* membaca data yang dikirimkan oleh *master*, data *bluetooth* adalah “1” jika *bluetooth* terkoneksi maka, data *bluetooth* adalah “0” jika tidak terkoneksi. Modul GPS akan membaca data *latitude* dan *longitude*. Kemudian modul SIM800L akan dikonfigurasi untuk koneksi menggunakan internet GPRS. Semua data tersebut akan disimpan pada *database* *Firestore* pada tabel *helm-iot* sesuai dengan *id* atau *key* dari helm tersebut, dan akan dilanjutkan proses pada aplikasi Android. Alur Flowchart dari perancangan sistem pada aplikasi Android dapat dilihat seperti pada Gambar 6.



Gambar 6. Flowchart Rancangan Sistem pada Aplikasi Android

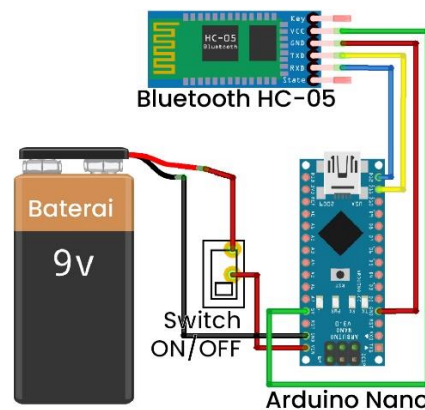
Alur flowchart dimulai dari pengguna melakukan login kedalam aplikasi menggunakan autentikasi Google dan

ketika berhasil maka akan menyimpan data *session*, *email*, *uid*, *photo_url* dan *account_key* ke *SharedPreferences*. Data pada *database Firebase* ditampilkan pada aplikasi Android, data *bluetooth* “1” berarti status helm adalah aman dan alarm keamanan mati. Data *bluetooth* adalah “0” berarti status helm adalah tercuri dan alarm keamanan helm akan menyala/hidup dan aplikasi Android akan menampilkan notifikasi bahwa helm telah tercuri. Pengguna dapat menghidupkan alarm suara pada helm, dan sistem akan memperbaharui data *alarm_helm* menjadi “1” pada *Firebase* dan *buzzer* akan berbunyi. Pengguna juga dapat mematikan alarm suara pada helm dan sistem akan memperbarui data *alarm_helm* menjadi “0” pada *Firebase* dan *buzzer* pada helm akan mematikan suara.

Proses dilanjutkan dengan kondisi pengguna apabila ingin mengubah data helm maka sistem akan mengubah data helm sesuai pilihan pengguna dan jika tidak maka akan melanjutkan prosesnya. Pengguna dapat mengubah nama helm dengan memasukkan data nama helm baru dan akan memperbarui data pada *Firebase* sesuai dengan data helm tersebut. Pengguna dapat menghubungkan perangkat helm baru dengan menghidupkan *bluetooth* pada android dan menghubungkan *bluetooth* pada android ke perangkat keras helm, kemudian sistem akan menyimpan data *helm_key* baru, *helm_name*, dan *account_id* ke *Firebase*. Pengguna dapat melakukan *logout* dari sistem dan data yang tersimpan pada *SharedPreferences* akan terhapus, dan selesai.

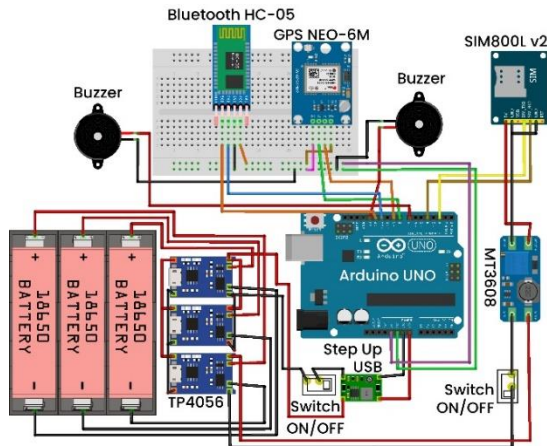
2.5. Perancangan Perangkat Keras

Rancangan perangkat keras dari sistem ini terdiri dari dua perangkat keras yaitu perangkat pada sepeda motor dan helm. Perancangan perangkat keras pada sepeda motor dapat dilihat seperti pada Gambar 7.



Gambar 7. Perancangan Perangkat Keras Sepeda Motor

Rangkaian tersebut terdiri dari beberapa perangkat yaitu, modul *Bluetooth HC-05* sebagai master, baterai 9v sebagai daya pada Arduino, *switch ON/OFF*, dan Arduino Nano sebagai perangkat *mikrocontroller*. Perancangan perangkat keras helm dapat dilihat seperti pada Gambar 8.



Gambar 8. Perancangan Perangkat Keras Helm

Rangkaian tersebut terdiri dari beberapa perangkat yaitu modul *Bluetooth* HC-05 sebagai *slave*, modul GPS NEO-6M untuk mendapatkan *latitude* dan *longitude*, tiga baterai 18650 sebagai daya pada Arduino. Terdapat SIM800L v2, dua *buzzer* sebagai *alarm* pada helm, dua *Switch ON/OFF*, tiga Modul *Charger TP4056* untuk mengisi daya baterai 18650. Terdapat modul *Step Up USB* untuk menaikkan tegangan baterai dari 3,7 volt menjadi 5 volt pada Arduino, Modul *Step Up MT3608* untuk menaikkan tegangan baterai dari 3.7 volt menjadi 5 volt pada SIM800L v2. Modul SIM800L v2 digunakan untuk mengirim data menggunakan jaringan internet dan Arduino UNO sebagai perangkat *mikrocontroller*.

3. Hasil dan Pembahasan

Sistem yang dihasilkan dari penelitian ini merupakan rancangan perangkat keras helm dan perangkat keras motor berbasis Internet of Things, dan perangkat lunak aplikasi Android. Aplikasi Android menggunakan *Firestore Realtime Database* sebagai basis data yang dapat menyimpan dan menampilkan data yang telah di proses secara *realtime*.

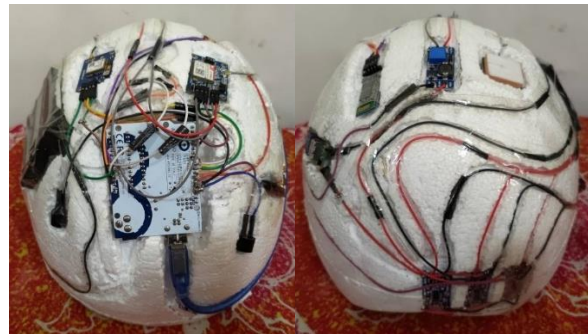
3.1. Perangkat Keras (*Hardware*)

Rancangan dari perangkat keras dalam sistem pendeteksi helm ini terdiri dari perangkat keras pada motor dan perangkat keras pada helm. Masing-masing perangkat keras tersebut memiliki komponen dan fungsi yang berbeda. Hasil dari perancangan perangkat keras pada motor dapat dijelaskan pada Gambar 9.



Gambar 9. Hasil Perancangan Perangkat Motor sebagai *Master*

Perangkat tersebut disusun dengan *casing* menggunakan kertas karton yang berukuran cukup kecil yaitu dengan panjang 20 cm dan lebar 15 cm, sehingga dapat diletakkan pada bagasi sepeda motor. Perangkat keras motor tersebut terdapat Arduino, kemudian modul *Bluetooth* HC-05 yang dikonfigurasi sebagai *master*, dan terdapat baterai 9v yang digunakan sebagai daya atau sumber energi dari perangkat tersebut, dan terdapat saklar atau *switch ON/OFF* untuk menyalakan atau mematikan perangkat tersebut. Hasil dari perancangan perangkat keras pada motor dapat dijelaskan seperti pada Gambar 10.



Gambar 10. Hasil Perancangan Perangkat Helm sebagai *Slave*

Perangkat tersebut disusun menggunakan *styrofoam* yang terdapat dalam helm yang terdapat Arduino Uno, modul *Bluetooth* HC-05 yang dikonfigurasi sebagai *slave*, GPS Neo-6m, SIM800L, dan dua *buzzer*. Baterai 18650 digunakan sebagai daya atau sumber energi dari perangkat tersebut, Modul *Charger TP4056* digunakan untuk mengisi daya baterai menggunakan kabel data *micro-USB*, dan Modul *Step Up USB*. Terdapat saklar atau *switch ON/OFF* untuk menyalakan atau mematikan perangkat keras pada helm. Ketika pengguna sedang tidak menggunakan helm tersebut atau diletakkan pada motor, perangkat pada helm dapat dihidupkan untuk mengaktifkan proses dari sistem pendeteksi tersebut. Perangkat keras ini dapat kompatibel dengan helm ukuran S keatas, dikarenakan perangkat yang ada pada helm tersebut cukup banyak dan memerlukan ruang yang cukup untuk merancang perangkat keras pada helm

3.2. Perangkat Lunak (*Software*)

Perangkat lunak yang telah dirancang menggunakan sistem operasi Android. Aplikasi Android ini berjalan dengan sistem operasi Android *Jelly Bean* (v4.3), yang bertujuan sebagai media penjelasan hasil dari data yang direkam dari perangkat keras. Tampilan dari aplikasi Android yang telah dirancang dapat dilihat seperti pada Gambar 11.

Tampilan tersebut terdapat *Toolbar* yang berisikan tombol *reload* di bagian atas kanan yang digunakan untuk me-*refresh* posisi dari *marker* pada *map* yang mewakili posisi dari helm tersebut. *Marker* pada *maps* terdapat *popup* yang dapat ditekan dan akan menuju ke aplikasi Google Maps yang dapat digunakan untuk melakukan *direction* dari posisi pengguna ke helm

tersebut. Bagian bawah aplikasi terdapat *Bottom Sheet* yang terdapat informasi detail dari helm pengguna, seperti status helm, *latitude*, *longitude*, lokasi helm, dan alarm helm yang terdapat tombol untuk mengaktifkan atau mematikan alarm helm tersebut.



Gambar 11. Tampilan Aplikasi Android Sistem

3.3. Hasil Uji Coba Modul Bluetooth HC-05

Modul *Bluetooth* HC-05 pada penelitian ini terdapat dua buah modul yang terdapat pada perangkat motor dan helm kedua modul tersebut digunakan sebagai koneksi *master* dan *slave*. Pengujian yang dilakukan dari modul ini yaitu seberapa jauh jarak yang dapat dilakukan agar *master* dan *slave* dapat terkoneksi dan *master* dapat mengirimkan data ke *slave*. Pengujian dari modul *bluetooth* ini dapat dilihat seperti pada Tabel 1.

Tabel 1. Pengujian Bluetooth HC-05

| Uji Coba Ke- | Jarak Master dan Slave (meter) | Master Mengirim | Slave Menerima | Keterangan | Alarm Buzzer |
|--------------|--------------------------------|-----------------|----------------|------------------|--------------|
| 1 | 2 | 1 | 1 | Terkoneksi | Mati |
| 2 | 4 | 1 | 1 | Terkoneksi | Mati |
| 3 | 6 | 1 | 1 | Terkoneksi | Mati |
| 4 | 8 | 1 | 1 | Terkoneksi | Mati |
| 5 | 9 | 1 | 1 | Terkoneksi | Mati |
| 6 | 10 | 1 | 1 | Terkoneksi | Mati |
| 7 | 11 | 1 | 0 | Tidak Terkoneksi | Menyala |
| 8 | 12 | 1 | 0 | Tidak Terkoneksi | Menyala |
| 9 | 13 | 1 | 0 | Tidak Terkoneksi | Menyala |
| 10 | 14 | 1 | 0 | Tidak Terkoneksi | Menyala |

Tabel 1 merupakan hasil dari pengujian modul *Bluetooth* HC-05 yang digunakan pada sistem sebagai *master* dan *slave*. Berdasarkan hasil pengujian yang dilakukan

sebanyak 10 kali, didapatkan hasil yaitu pada jarak awal hingga 10 meter *bluetooth* masih dapat melakukan koneksi *master* dan *slave*, dimana *master* mengirimkan data "1" dan *slave* menerima data tersebut, dan alarm *buzzer* pada perangkat keras helm tidak akan menyala atau mati. Pengujian *master* dan *slave* dengan jarak 11 meter, 12 meter, dan seterusnya *bluetooth* tidak dapat terkoneksi dan *slave* tidak menerima apa yang dikirimkan dari *master* sehingga *slave* akan menampilkan data "0" pada sistem yang berarti helm sedang tercuri, dan alarm *buzzer* pada perangkat keras helm akan menyala.

3.4. Hasil Uji Coba Mengirim dan Membaca Data dari Firebase

Setelah berhasil melakukan uji coba terhadap modul *Bluetooth* HC-05, selanjutnya yaitu menguji modul SIM800L v2 yang digunakan untuk mengkoneksikan sistem ke internet menggunakan jaringan GPRS. Pengujian ini terdiri dari dua tahapan yaitu mengirim data ke *Firestore* dan membaca data dari *Firestore*. Pengujian mengirim data ke *Firestore* dapat dilihat seperti pada Tabel 2 berikut.

Tabel 2. Pengujian Mengirim Data ke Firestore Menggunakan Sim800L V2

| Uji Coba Ke- | Waktu Pengiriman | Waktu diterima Firestore | Selisih Waktu (detik) | Keterangan |
|------------------|------------------|--------------------------|-----------------------|---------------|
| 1 | 18:13:22 | 18:13:23 | 1 | Berhasil |
| 2 | 18:13:48 | 18:13:49 | 1 | Berhasil |
| 3 | 18:14:01 | 18:14:02 | 1 | Berhasil |
| 4 | 18:14:13 | 18:14:15 | 2 | Berhasil |
| 5 | 18:14:26 | 18:14:27 | 1 | Berhasil |
| 6 | 18:14:39 | 18:14:40 | 1 | Berhasil |
| 7 | 18:14:52 | 18:14:53 | 1 | Berhasil |
| 8 | 18:15:05 | 18:15:06 | 1 | Berhasil |
| 9 | 18:15:17 | 18:15:18 | 1 | Berhasil |
| 10 | 18:15:30 | 18:15:31 | 1 | Berhasil |
| Rata-rata | | | 1,1 | |
| Tingkat Berhasil | | | | 100% Berhasil |

Tabel 2 merupakan hasil dari pengujian mengirim data ke *Firestore* menggunakan modul SIM800L v2. Berdasarkan hasil pengujian yang dilakukan sebanyak 10 kali, didapatkan waktu pengiriman dan waktu diterima *Firestore* yang berbeda-beda. Selisih waktu pengiriman rata-rata adalah 1,1 detik untuk menyimpan data dari perangkat keras ke *Firestore*, dan semua pengujian tersebut berhasil dilakukan untuk menyimpan data ke *Firestore*. Rata-rata selisih mengirim data tersebut juga lebih cepat dibandingkan penelitian sebelumnya yang disajikan dalam penulisan ini, yaitu 2,1 detik dan menggunakan metode penyimpanan data yang berbeda yaitu *Broker* MQTT Adafruit IO. Pengujian membaca data dari *Firestore* dapat dilihat seperti pada Tabel 3.

Tabel 3 merupakan hasil dari pengujian membaca data dari *Firestore* menggunakan modul SIM800L v2. Berdasarkan hasil pengujian yang dilakukan sebanyak 10 kali, didapatkan waktu membaca data yang berbeda-

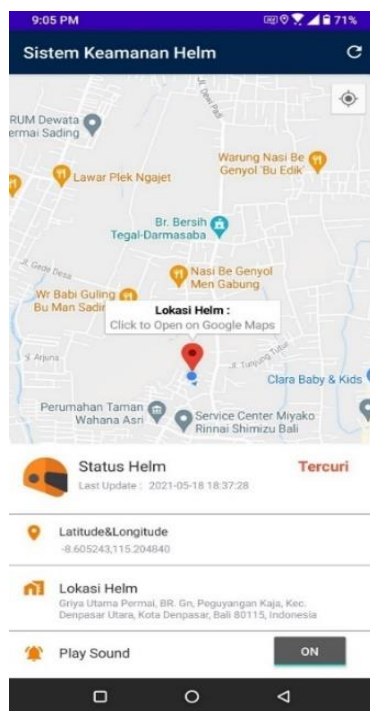
beda. Selisih waktu membaca data mendapatkan rata-rata 5,3 detik untuk membaca data dari *Firebase* ke perangkat keras, dan semua pengujian tersebut berhasil dilakukan untuk membaca data dari *Firebase*.

Tabel 3. Pengujian Membaca Data dari Firebase Menggunakan Sim8001 V2

| Uji Coba Ke- | Waktu Membaca Data | Waktu Data diterima Perangkat Helm | Selisih Waktu (Detik) | Keterangan |
|-------------------------|--------------------|------------------------------------|-----------------------|-------------------------|
| 1 | 18:13:34 | 18:13:44 | 10 | Berhasil |
| 2 | 18:13:53 | 18:13:57 | 4 | Berhasil |
| 3 | 18:14:05 | 18:14:10 | 5 | Berhasil |
| 4 | 18:14:18 | 18:14:23 | 5 | Berhasil |
| 5 | 18:14:31 | 18:14:36 | 5 | Berhasil |
| 6 | 18:14:44 | 18:14:49 | 5 | Berhasil |
| 7 | 18:14:57 | 18:15:01 | 4 | Berhasil |
| 8 | 18:15:09 | 18:15:14 | 5 | Berhasil |
| 9 | 18:15:22 | 18:15:27 | 5 | Berhasil |
| 10 | 18:15:35 | 18:15:40 | 5 | Berhasil |
| Rata-rata | | | 5,3 | |
| Tingkat Berhasil | | | | 100% Berhasil |

3.5. Hasil Uji Coba Aplikasi Android

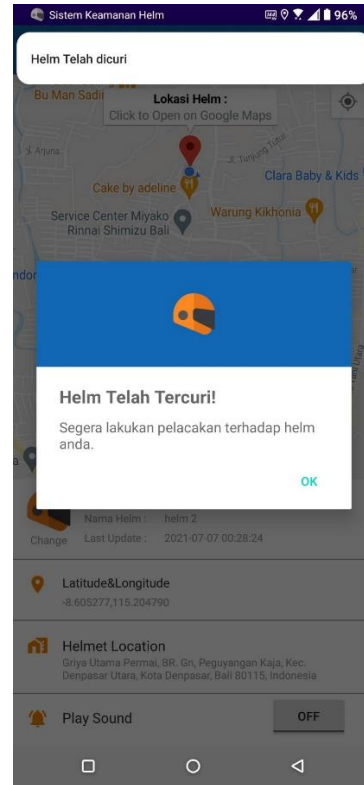
Setelah perangkat keras diuji coba dan berhasil untuk dijalankan, pengujian terakhir dilakukan pada perangkat lunak sistem atau aplikasi Android yang digunakan pengguna sistem. Pengujian ini dilakukan untuk mengetahui apakah sistem dapat berjalan sesuai dengan harapan pengguna. Pengujian dari aplikasi Android tersebut dapat dilihat seperti pada Gambar 12.



Gambar 12. Uji Coba Status Helm dan Suara Helm

Status pada helm tersebut ada dua yaitu “aman” dan “tercuri”. Status aman didapatkan dari koneksi antara *bluetooth master* dan *slave*, status pada helm adalah “aman” jika *bluetooth* terkoneksi. Pengguna juga dapat

menghidupkan alarm yang berfungsi untuk mengetahui posisi dari helm tersebut jika pengguna sudah mendekati helm tersebut. Alarm pada perangkat *slave* akan otomatis berbunyi jika koneksi *master* dan *slave* terputus dan pada aplikasi Android akan menampilkan status yaitu “tercuri”, selanjutnya sistem aplikasi Android tersebut akan menampilkan notifikasi ke pengguna secara langsung jika helm sedang tercuri. Notifikasi pada aplikasi Android dapat dilihat seperti pada Gambar 13.



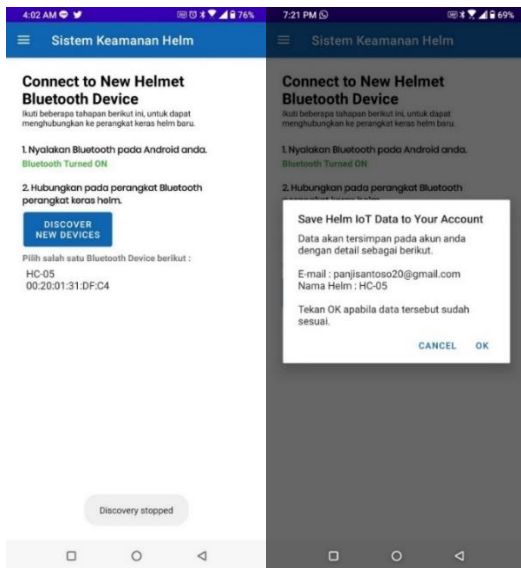
Gambar 13. Notifikasi Helm Tercuri

Notifikasi tersebut ditampilkan berdasarkan data dari *Firebase Realtime Database*. Perangkat pada *slave* akan mengirim/menyimpan data di *Firebase* yaitu “0” pada *bluetooth*, apabila koneksi *master* dan *slave* terputus. Sehingga pada sistem akan membaca data tersebut dan akan menampilkan notifikasi pada aplikasi Android.

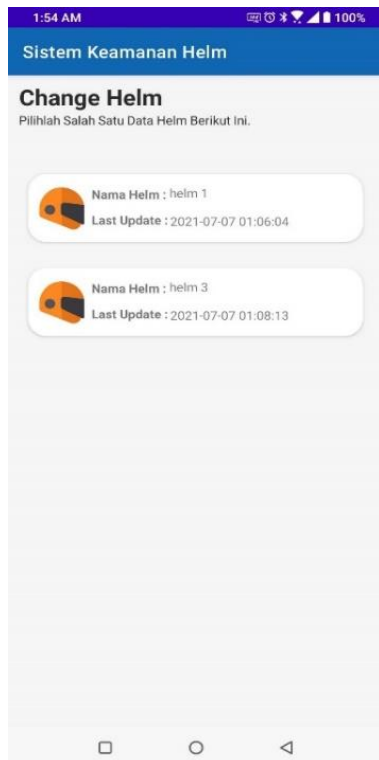
Proses tersebut dilanjutkan dengan menekan tombol “Save Data to Your Account” yang selanjutnya akan menampilkan *pop up* atau *dialog* dengan data yang ditampilkan yaitu *email* pengguna yang telah *login* dan nama *bluetooth* yang selanjutnya akan disimpan menjadi nama helm pada *Firebase*. Pengguna dapat menekan tombol OK apabila data yang tercantum sudah benar dan data akan tersimpan pada *Firebase Realtime Database*.

Tampilan tersebut (Gambar 14 dan 15) terdapat *list* dari helm yang dimiliki dari pengguna yang telah melakukan *login* dan mendaftarkan data helm ke *Firebase Realtime Database*. Pengguna dapat mengubah data helm yang digunakan dengan menekan salah satu data yang tersedia

dan data yang ditampilkan pada *maps* akan berubah sesuai dengan data helm yang dipilih.



Gambar 14. Fitur Menambahkan Data Helm Baru



Gambar 15. Fitur Mengubah Data Helm

4. Kesimpulan

Kesimpulan dari penelitian Sistem Pendeteksi Pencurian Helm Berbasis *Internet of Things* ini yaitu, perancangan sistem pendeteksi pencurian helm ini menggunakan teknologi berbasis *Internet of Things*. Perancangan perangkat keras terdiri dari dua perangkat yaitu perangkat keras motor yang terdiri dari Arduino Nano, *Bluetooth* HC-05, *Switch ON/OFF*, dan baterai 9v.

Perangkat keras helm terdiri dari *Bluetooth* HC-05, modul GPS NEO-6m, tiga baterai 18650, SIM800L v2, dua *Buzzer*, dua *Switch ON/OFF*, Modul *Charger* TP4056, Modul Step Up MT3608, Modul *Step Up* USB dan Arduino UNO. Modul *bluetooth* pada motor dikonfigurasi sebagai *master* dan pada helm sebagai *slave*, yang berfungsi sebagai indikator keamanan pada pencurian helm. Berdasarkan uji coba, koneksi *master slave* akan terputus apabila melebihi jarak 10m antara kedua *bluetooth* tersebut. Rancangan sistem dalam penelitian dapat mendeteksi adanya peristiwa pencurian helm menggunakan perangkat keras *Internet of Things*, dan dapat melakukan pelacakan posisi helm apabila tercuri, pada fitur *maps* menggunakan aplikasi Android yang sudah dirancang sebelumnya. Penggunaan *Firebase* dapat mendukung dalam penyusunan penelitian ini, khususnya pada fitur *Firebase Realtime Database*. *Firebase Realtime Database* dapat menyimpan data melalui perangkat keras kemudian dibaca pada aplikasi Android secara *realtime*. Data pada *Firebase* akan melakukan *update* secara otomatis, sehingga pada aplikasi Android akan memperbarui data tanpa harus melakukan *refresh* atau *reload* pada halaman aplikasi Android. Pengiriman data dari perangkat keras ke *Firebase* berhasil dilakukan, berdasarkan uji coba dihasilkan rata-rata pengiriman data yaitu 1,1 detik dan rata-rata waktu membaca data yaitu 5,3 detik. Penelitian ini terdapat kelemahan yaitu perangkat keras pada helm tidak dapat dirancang pada helm yang memiliki ukuran lebih kecil dari S. Hal tersebut dikarenakan komponen atau modul perangkat keras pada helm cukup banyak dan berukuran cukup besar. Solusi yang dapat diberikan untuk kelemahan tersebut yaitu menggunakan modul yang lebih *compact* atau berukuran lebih kecil, sehingga dapat dirancang pada helm yang memiliki ukuran lebih kecil dari S.

Saran yang dapat diberikan untuk pengembangan penelitian Sistem Pendeteksi Pencurian Helm ini adalah dapat dikembangkan untuk menambahkan modul IoT terbaru dan fitur yang baru agar dapat mendeteksi pencurian helm yang lebih efisien, akurat dan terbaharukan. Seperti menambahkan fitur untuk menghidupkan dan mematikan perangkat keras melalui aplikasi Android, dan dapat mengetahui indikator kapasitas baterai yang tersisa pada perangkat keras helm dan motor. Pengembangan sistem selanjutnya diharapkan untuk menambahkan modul kamera pada motor yang dapat digunakan untuk mendeteksi wajah disekitar sepeda motor pengguna sistem keamanan ini. Sehingga sistem dapat menyimpan *dataset* wajah pengguna atau pemilik sistem, dan dapat mendeteksi wajah yang bukan merupakan pengguna sistem dengan dugaan sebagai pencuri.

Daftar Rujukan

- [1] F. Aryaviocholda, M. Hannats, H. Ichsan, and A. S. Budi, "Rancangan Sistem Pendeteksi Pencurian Helm Menggunakan

- Protokol MQTT Dan Bluetooth HC-05 Berbasis Arduino,” 2020. <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/6969>
- [2] Y. Efendi, “Internet Of Things (Iot) Sistem Pengendalian Lampu Menggunakan Raspberry Pi Berbasis Mobile,” *Jurnal Ilmiah Ilmu Komputer*, vol. 4, no. 2, pp. 21–27, 2018. <https://doi.org/10.35329/jiik.v4i2.41>
- [3] E. A. Siddiq and H. Effendi, “Sistem Monitoring Sepeda Motor Berbasis Mikrokontroler Menggunakan GPS,” (*Jurnal Teknik Elektro Dan Vokasional*, vol. 06, no. 02, pp. 383–390, 2020. <https://doi.org/10.24036/jtev.v6i2.109397>
- [4] I. Kholilah and A. R. al Tahtawi, “Aplikasi Arduino-Android untuk Sistem Keamanan Sepeda Motor,” *Jurnal Teknologi Rekayasa*, vol. 1, no. 1, p. 53, 2017. <http://dx.doi.org/10.31544/jtera.v1.i1.2016.53-58>
- [5] P. V. G. Y. R. Kalshetty, “Bluetooth Based Smart Automation System Using Android,” *International Journal of Science and Research (IJSR)*, vol. 6, no. 5, pp. 1003–1006, 2017. <https://www.ijsr.net/archive/v6i5/3051709.pdf>
- [6] D. Wulandari and Willyansah, “Sistem Kontrol Kendaraan Berbasis IOT,” *Jaringan Sistem Informasi Robotik*, vol. 2, no. 02, pp. 1–6, 2018. <http://ojsamik.amikmitragama.ac.id/index.php/js/article/view/41>
- [7] D. Hermanto, Yamato, and A. R. Machdi, “Perancangan Sistem Keamanan Berkendara Roda Dua Menggunakan Arduino Uno Berbasis Sms,” *Jurnal Online Mahasiswa (JOM) Bidang Teknik Elektro*, vol. 1, no. 1, pp. 1–10, 2016, [Online]. Available: <http://jom.unpak.ac.id/index.php/teknikelektro/article/view/506>
- [8] D. Pangestu, A. Muid, and U. Ristian, “Purwarupa Sistem Informasi Titik Lokasi Dan Intensitas Curah Hujan Di Kota Pontianak Berbasis Website,” *Jurnal Coding, Sistem Komputer Untan*, vol. 6, no. 3, pp. 247–254, 2018. <http://dx.doi.org/10.26418/coding.v6i3.29059>
- [9] Dony Novaliendry and S. Andriani, “English Edugame Application for Childhood base on Android,” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 4, no. 1, pp. 187–192, 2020. <https://doi.org/10.29207/resti.v4i1.1622>
- [10] Ilham Firman Maulana, “Penerapan Firebase Realtime Database pada Aplikasi E-Tilang Smartphone berbasis Mobile Android,” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 4, no. 5, pp. 854–863, 2020. <https://doi.org/10.29207/resti.v4i5.2232>
- [11] K. N. M. Ngafidin, A. Arista, and R. N. S. Amriza, “Implementasi Firebase Realtime Database pada Aplikasi FeedbackMe,” *Resti*, vol. 1, no. 10, pp. 327–334, 2021. <https://doi.org/10.29207/resti.v5i2.2909>