

Perancangan Sistem Start Engine Dan Keamanan Sepeda Motor Menggunakan Arduino Uno Berbasis Android

Nahrowi^{1*}, Perani Rosyani¹

¹Fakultas Ilmu Komputer, Teknik Informatika, Universitas Pamulang, Jl. Raya Puspiptek No. 46, Kel. Buaran, Kec. Serpong, Kota Tangerang Selatan. Banten 15310, Indonesia

Email: ¹nahrowi.1604@gmail.com, ²dosen00837@unpam.ac.id

(* : coressponding author)

Abstrak– Pada kemajuan teknologi saat ini, masyarakat telah menggunakan kendaraan bermotor salah satunya ialah sepeda motor sebagai alat transportasi di dalam kehidupan sehari-hari sebagai fasilitas untuk beraktivitas seperti bekerja. Namun Sistem keamanan sepeda motor pada saat ini masih menggunakan kunci konvensional yang mudah dirusak oleh pencuri dalam waktu singkat yang dapat mengakibatkan kerugian materiil dan non-materiil bagi pemilik sepeda motor. Salah satu teknologi otomotif yang dapat meningkatkan kenyamanan dan kemudahan pengguna sepeda motor adalah dengan memperkenalkan sistem *start engine* dan keamanan sepeda motor yang lebih canggih dan mudah melalui *smartphone android*. Metode pengembangan sistem yang digunakan pada penelitian ini adalah metode waterfall, Model pengembangan ini bersifat linear dari tahap awal pengembangan sistem yaitu tahap perencanaan sampai tahap akhir pengembangan sistem yaitu tahap pemeliharaan. Tujuan dari penelitian ini adalah merancang sebuah alat menggunakan Mikrokontroler Arduino untuk mengontrol *Start Engine* dan *Alarm* Sepeda Motor, membuat sebuah aplikasi *android* yang berfungsi sebagai *remote* untuk mengontrol *Start Engine* dan *Alarm* Sepeda Motor, meningkatkan keamanan sepeda motor dengan adanya alarm sebagai sistem keamanannya, mempermudah saat mencari sepeda motor di dalam area parkir dengan adanya *alarm* dan meningkatkan kemudahan dan kenyamanan pengguna dalam mengoperasikan sepeda motor. Dari hasil penelitian ini diharapkan aplikasi dapat memudahkan pengguna untuk melakukan pengendalian terhadap kendaraan bermotor seperti sepeda motor melalui *smartphone android*.

Kata Kunci: Sepeda Motor, *Android*, *Arduino*, *Start Engine*, *Alarm*

Abstract– In today's technological advances, people have used motorized vehicles, one of which is a motorbike as a means of transportation in everyday life as a facility for activities such as work. However, the motorcycle security system currently still uses conventional locks that are easily damaged by thieves in a short time which can result in material and non-material losses for motorcycle owners. One of the automotive technologies that can increase the comfort and convenience of motorcycle users is by introducing a more sophisticated and easy engine start and motorcycle safety system through an *Android smartphone*. The system development method used in this study is the waterfall method. This development model is linear from the early stages of system development, namely the planning stage to the final stage of system development, namely the maintenance stage. The purpose of this research is to design a tool using the *Arduino Microcontroller* to control the *Start Engine* and *Motorcycle Alarm*, create an *android application* that functions as a *remote* to control the *Start Engine* and *Motorcycle Alarm*, increase motorcycle security with an alarm as a security system, make it easier when looking for motorbikes in the parking area with an alarm and increasing the ease and comfort of users in operating motorbikes. From the results of this study it is hoped that the application can make it easier for users to control motorized vehicles such as motorbikes via an *Android smartphone*.

Keywords: *Motorcycle*, *Android*, *Arduino*, *Start Engine*, *Alarm*

1. PENDAHULUAN

Perkembangan teknologi yang cukup pesat saat ini sudah memiliki banyak manfaat di dalam perkembangan dan kemajuan teknologi, diantaranya dengan cara menyalakan sepeda motor berbasis *android*. Permasalahan yang muncul atas ketidaktahuan manusia ataupun dari sebab lainnya, semakin lama sudah dapat teratasi seiring berjalannya waktu. Untuk manfaat yang dapat membantu memecahkan masalah tersebut adalah dengan menggunakan sistem kendali berbasis *smartphone*. Pada perkembangan teknologi yang semakin maju ini masyarakat sudah menggunakan *smartphone* yang memiliki sistem operasi *android* yang dikembangkan oleh *Google*. Mayoritas masyarakat

Indonesia menggunakan operasi sistem *android* karena untuk pengoperasiannya lebih mudah semacam mengoperasikan komputer pada umumnya. (Tharishny et al., 2016).

Mikrokontroler merupakan sebuah sistem komputer yang seluruh atau sebagian besar elemennya dikemas dalam satu chip IC, sehingga sering disebut single chip microcomputer (Nurus Sholeh et al., 2020). *Mikrokontroler* yang digunakan ialah *Arduino*. (Setiawan, 2018) *Arduino* adalah papan rangkaian elektronik open source atau kit elektronik yang terdiri dari komponen utama berupa chip mikrokontroler dengan jenis AVR dari perusahaan Atmel. Sistem keamanan sepeda motor pada saat ini masih menggunakan kunci konvensional yang mudah dirusak oleh pencuri dalam waktu singkat yang dapat mengakibatkan kerugian materiil dan non-materiil bagi pemilik sepeda motor. Meskipun ada beberapa produsen sepeda motor sudah menggunakan *Shutter Key* yaitu penguncian yang memiliki cover magnet dan juga dilengkapi dengan tuas penutup manual tetapi tampaknya keamanan tersebut masih tidak efisien untuk keamanan lantaran rentan di rusak dengan menggunakan kunci T. Oleh karena itu, ide untuk perancangan alat ini penulis menggunakan perangkat *Mikrokontroler Arduino Uno* dan juga *smartphone* sebagai pengendali dalam *start engine* sepeda motor secara mudah.

Di era digitalisasi ini banyak sistem keamanan baru yang lebih aman dibanding kunci konvensional. Misalnya sistem keamanan berbasis komputer. Dengan sistem keamanan berbasis komputer diharapkan dapat menciptakan perlindungan lebih dibanding keamanan konvensional.

2. METODOLOGI PENELITIAN

2.1 Metode Pengumpulan Data

Untuk mengumpulkan data pada penelitian ini terdapat beberapa metode pengumpulan data yang dapat digunakan, di antaranya:

a. Studi Pustaka

Metode pengumpulan data ini dilakukan dengan cara mengumpulkan informasi dari berbagai sumber yang terkait dengan sistem *start engine* dan keamanan sepeda motor, seperti buku, jurnal, artikel, dan internet. Studi pustaka dapat membantu dalam memperoleh informasi tentang teknologi dan fitur yang terkait dengan sistem tersebut.

b. Observasi

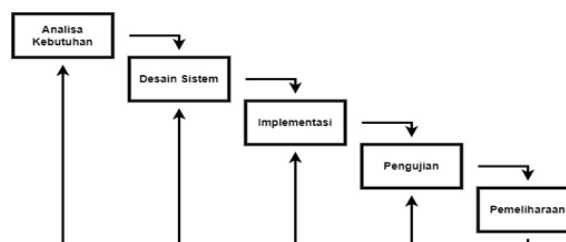
Metode pengumpulan data ini dilakukan dengan cara mengamati langsung kondisi dan perilaku pengendara sepeda motor pada saat melakukan pengoperasian sistem *start engine* dan keamanan sepeda motor.

c. Wawancara

Metode pengumpulan data ini dilakukan dengan cara mewawancarai pengendara sepeda motor yang menggunakan sistem *start engine* dan keamanan sepeda motor berbasis *Arduino Uno*. Wawancara dilakukan untuk mendapatkan informasi lebih detail tentang pengalaman pengendara dalam menggunakan sistem tersebut, serta untuk mendapatkan masukan dan saran dari pengendara.

2.2 Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan pada penelitian ini adalah metode *waterfall*. Keuntungan dari metode *waterfall* adalah mudah dipahami, mudah dikelola dan dapat memberikan hasil yang terukur.



Gambar 1. Tahapan *Waterfall*

Dalam perancangan sistem *Start Engine* dan Keamanan Sepeda Motor menggunakan *Arduino Uno* berbasis *Android*, Tahap - tahap pengembangan sistem yang dapat dilakukan dengan metode *Waterfall* adalah sebagai berikut:

a. Analisis Kebutuhan

Tahap pertama adalah menganalisis kebutuhan sistem secara rinci, termasuk fitur dan fungsi yang diperlukan dalam sistem.

b. Desain

Tahap selanjutnya adalah merancang sistem secara detail, termasuk arsitektur perangkat keras dan perangkat lunak.

c. Implementasi

Tahap ini melibatkan implementasi rancangan sistem menjadi bentuk fisik dan program perangkat lunak.

d. Pengujian

Tahap berikutnya adalah menguji sistem dan memastikan bahwa semuanya berfungsi dengan benar. Uji coba dapat melibatkan pengujian terhadap fitur-fitur yang telah dirancang sebelumnya.

e. Pemeliharaan

Tahap akhir adalah pemeliharaan sistem, yang melibatkan pemantauan dan perbaikan sistem secara berkala. Dan harus memastikan bahwa sistem selalu berfungsi dengan baik dan memperbarui perangkat lunak atau perangkat keras jika diperlukan.

3. ANALISA DAN PEMBAHASAN

3.1 Analisa Kebutuhan

Pada tahapan ini, akan dilakukan analisa kebutuhan sistem, yang terbagi ke dalam tiga bagian, diantaranya yaitu analisa kebutuhan input, proses dan output.

a. Analisa Kebutuhan Input

Untuk kebutuhan input di dalam penelitian ini diantaranya:

1. Menyambungkan *wireless* melalui *Bluetooth* yang digunakan sebagai komunikasi antar perangkat *smartphone android* dengan *Arduino Uno*.
2. Perintah untuk menyalakan dan mematikan *relay* yang dikirimkan dari aplikasi yang ada pada perangkat *smartphone android* ke rangkaian alat.

b. Analisa Kebutuhan Proses

Untuk kebutuhan proses pada penelitian ini adalah:

1. Rangkaian alat *start engine* akan menerima perintah dari perangkat *smartphone android* untuk melakukan perintah menyalakan dan mematikan *relay* yang digunakan untuk menyalakan dan mematikan sepeda motor serta *alarm*.
2. Modul *Bluetooth HC-05* akan mengirimkan data seperti perintah yang di berikan pada serangkaian alat, lalu mengirimkan datanya ke *relay*.
3. Aplikasi *start engine* mengirimkan data berupa perintah ke alat melalui koneksi *Bluetooth*.

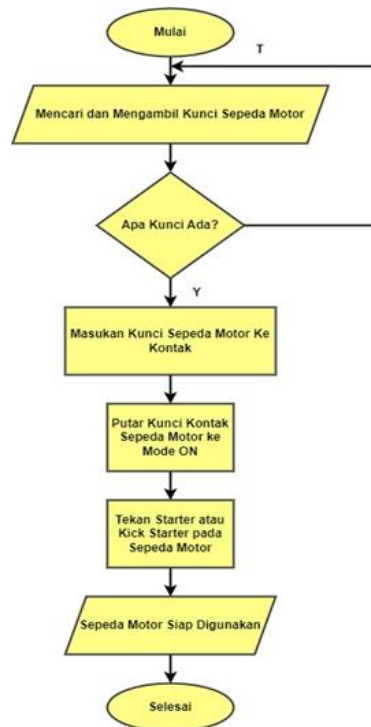
c. Analisa Kebutuhan Output

Untuk kebutuhan output pada penelitian ini adalah:

1. Rangkaian alat *start engine* dapat menyalakan dan mematikan *relay* yang terhubung dengan sistem kelistrikan sepeda motor pada saat menerima perintah dari aplikasi *start engine*.
2. Aplikasi *start engine* akan menampilkan tombol *on off* dan perintah suara untuk mengetahui alat dapat berfungsi dengan baik atau tidak.

3.2 Analisa Sistem Berjalan

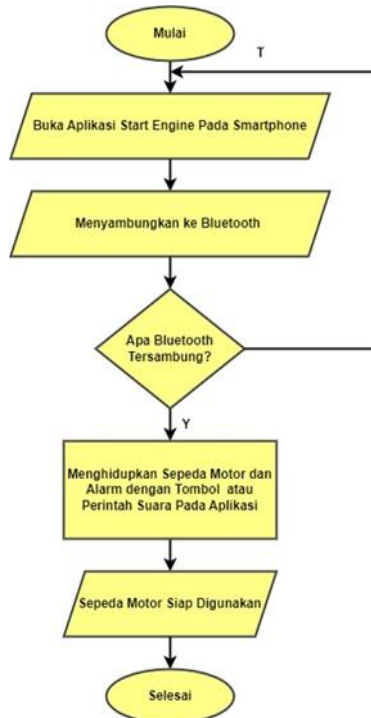
Analisis sistem berjalan merupakan tahapan untuk menganalisis sistem yang sudah berjalan, yang bertujuan untuk mengetahui dan mengamati apa saja yang terlibat di dalam suatu sistem yang saling berhubungan antara satu proses dengan proses yang lainnya.



Gambar 2. Flowchart Analisa Sistem Berjalan

3.3 Sistem Yang Diusulkan

Untuk tahapan ini akan dilakukan perancangan model sistem informasi secara menyeluruh, yang bertujuan untuk memberikan gambaran pada sistem analisis pembuatan program perihal input kedalam proses dan apa yang akan dihasilkan output.



Gambar 3. Flowchart Start Engine Sistem yang Diusulkan

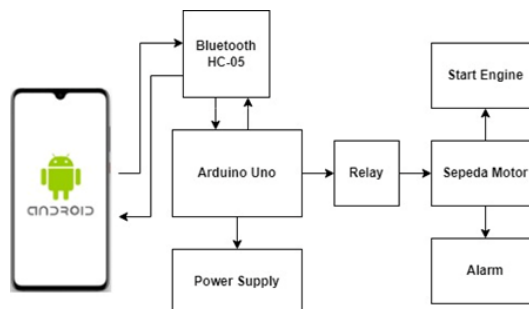
3.4 Perancangan Sistem

Untuk perancangan sistem pada penelitian ini terbagi menjadi dua yaitu Perancangan Perangkat Keras (*Hardware*) dan Perancangan Perangkat Lunak (*Software*).

3.4.1 Perancangan Perangkat Keras (*Hardware*)

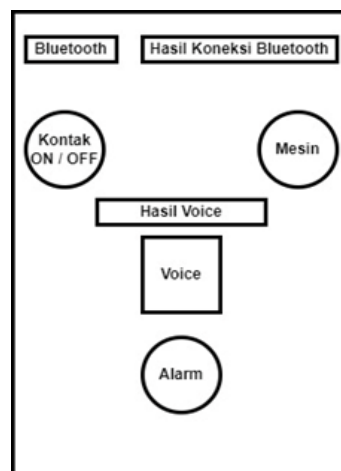
Perancangan Perangkat Keras (*Hardware*) merupakan rancangan atau rangkaian dari alat yang digunakan untuk membangun sistem *start engine* dan keamanan sepeda motor menggunakan *smartphone android* sebagai pengendalinya dan *Arduino uno* sebagai *mikrokontrolernya*. Untuk tahapan perancangan perangkat keras (*Hardware*) ini meliputi perancangan bagian elektronik dan perancangan bagian mekanik.

a. Desain Perangkat Keras (*Hardware*)



Gambar 4. Diagram Perangkat Keras (*Hardware*)

b. Desain Perangkat Lunak (*Software*)

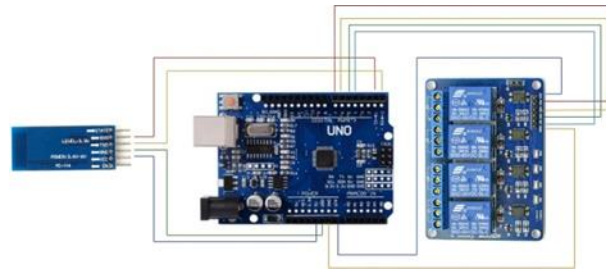


Gambar 5. Diagram Perangkat Lunak (*Software*)

3.4.2 Perancangan Alat *Start Engine*

Perancangan alat *start engine* pada sepeda motor ini dirancang menggunakan *Arduino uno* sebagai komponen utama yang akan di program menggunakan *software Arduino IDE (Integrate Development Enviroment)*. Untuk merancang alat *start engine* ini akan menggunakan beberapa tahapan diantaranya:

1. Perancangan alat melalui cara menggabungkan modul atau komponen utama yaitu *Relay Module 4 Channel* dan modul *Bluetooth HC-05* dengan *Arduino Uno* menggunakan kabel *jumper*.
2. Membuat program *Arduino* pada *software* menggunakan *Arduino IDE (Integrate Development Enviroment)* yaitu suatu *software* yang dipakai untuk membuat, mengedit suatu kode program, memverifikasi, dan mengunggah kode program ke *arduino*.
3. Pemasangan seluruh komponen alat pada kelistrikan sepeda motor.



Gambar 6. Skema Perancangan Alat *Start Engine*

3.4.3 Pemrograman *Arduino IDE*

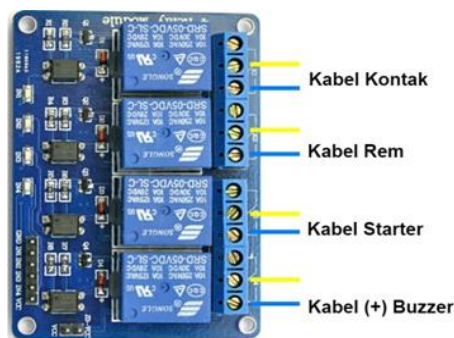
Pada tahapan pemrograman ini adalah dengan menjalankan perintah berbentuk codingan ke dalam *arduino uno* menggunakan *software Arduino IDE (Integrate Development Enviroment)*. Dan perintah yang digunakan bertujuan untuk menyalakan output *relay* yang terhubung pada kelistrikan sepeda motor sehingga dapat menjalankan *start engine*.

```
File Edit Sketch Tools Help  
#include <Arduino.h>  
#define RELAY_1 4  
#define RELAY_2 5  
#define RELAY_3 6  
#define RELAY_4 7  
  
void setup() {  
  pinMode(RELAY_1, OUTPUT);  
  pinMode(RELAY_2, OUTPUT);  
  pinMode(RELAY_3, OUTPUT);  
  pinMode(RELAY_4, OUTPUT);  
}  
  
void loop() {  
  digitalWrite(RELAY_1, HIGH);  
  digitalWrite(RELAY_2, HIGH);  
}
```

Gambar 7. Pemrograman Menggunakan *Software Arduino IDE*

3.4.4 Rangkaian Alat Kelistrikan Sepeda Motor

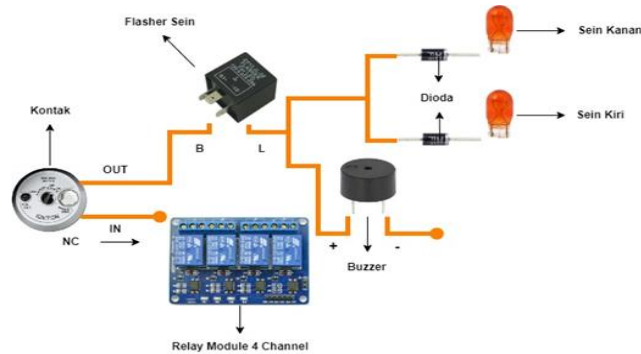
Untuk merangkai alat ke bagian kelistrikan pada sepeda motor ini, yaitu dengan cara menghubungkan *relay* ke kabel kelistrikan sepeda motor yang berguna untuk menjalankan *start engine* seperti menghubungkan kabel rem, kabel kontak, kabel *starter* dan juga kabel (+) *buzzer*.



Gambar 8. Rangkaian Relay Pada Kelistrikan Sepeda Motor

3.4.5 Perancangan *Alarm Keamanan Sepeda Motor*

Untuk perancangan sistem *alarm* keamanan sepeda motor ini sama dengan perancangan pada sistem *start engine* namun untuk sistem *alarm* keamanan ini harus ditambahkan *relay* atau *flasher sein* guna memutus sambungan ke kontak dan membunyikan *buzzer* sebagai *alarmnya*, dan jika ada yang memaksa untuk membobol kontak ke posisi ON maka *buzzer alarm* akan berbunyi dan motor tidak dapat di *starter* dan di *kick starter* sampai mengembalikan kontak ke posisi semula atau dengan mematikan *alarm* menggunakan aplikasi *start engine* di *smartphone android*.

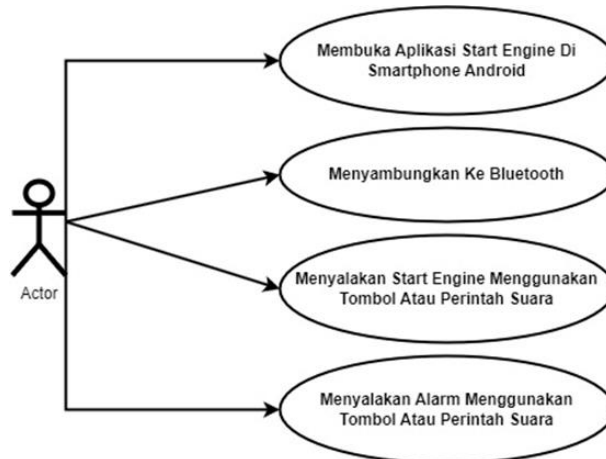


Gambar 9. Perancangan Alarm Keamanan Sepeda Motor

3.5 Perancangan Perangkat Lunak (Software)

Pada tahapan perancangan sistem *start engine* ini ialah menggunakan UML (*Unified Modeling Language*) yaitu sebagai jembatan dalam mengkomunikasikan beberapa aspek dalam sistem melalui jumlah elemen grafis yang bisa dikombinasikan menjadi diagram. UML (*Unified Modeling Language*) terdiri dari beberapa macam diagram diantaranya *Use Case Diagram*, *Activity Diagram* dan *Sequence Diagram*. Dan pada tahapan ini juga akan melakukan perancangan *user interface* yang menjadikan landasan awal untuk pembuatan tampilan pada aplikasi *start engine*.

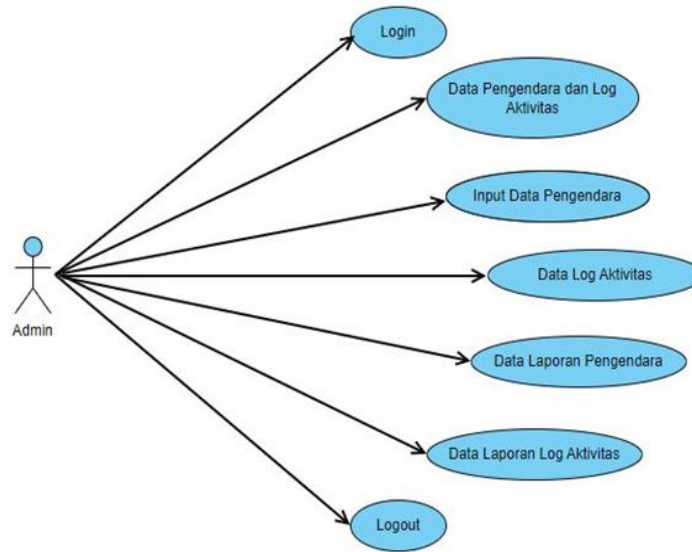
3.5.1 Use Case Diagram



Gambar 10. Use Case Diagram Aplikasi Start Engine

Tabel 1. Deskripsi Use Case Diagram Aplikasi Start Engine

Aktor	Keterangan
Pengguna	<ul style="list-style-type: none"> • Membuka Aplikasi Start Engine Di Smartphone Android: pengguna membuka aplikasi <i>start engine</i> pada <i>smartphone Android</i>. • Menyambungkan ke Bluetooth: Pengguna mencari dan menyambungkan perangkat <i>Bluetooth</i> dengan <i>Arduino uno</i> melalui modul <i>Bluetooth HC-05</i>. • Menyalakan Start Engine Menggunakan Tombol atau Perintah Suara: Pengguna dapat menggunakan tombol atau perintah suara untuk menyalakan sepeda motor pada <i>smartphone Android</i>. • Menyalakan Alarm Menggunakan Tombol atau Perintah Suara: Pengguna dapat menggunakan tombol atau perintah suara untuk menyalakan alarm sepeda motor pada <i>smartphone Android</i>.



Gambar 11. Use Case Diagram Web Admin Start Engine

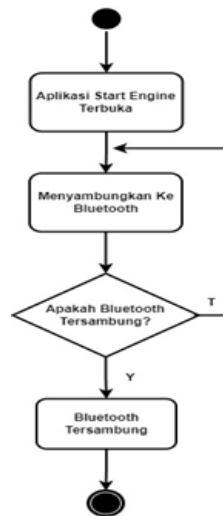
Tabel 2. Deskripsi Use Case Diagram Web Admin Start Engine

Aktor	Keterangan
Admin	<ul style="list-style-type: none"> • Login : Admin membuka web <i>start engine</i> di web browser. • Data pengendara dan log aktivitas : Admin dapat menambahkan data pengendara dan melihat log aktivitas. • Input data pengendara : Admin dapat menambahkan data pengendara. • Data log aktivitas : Admin dapat melihat data log aktivitas pengendara. • Data laporan pengendara : Admin dapat melihat data dan mencetak laporan pengendara. • Data laporan log aktivitas : Admin dapat melihat data dan mencetak laporan log aktivitas pengendara. • Logout : Admin keluar dari web <i>start engine</i>.

3.5.2 Activity Diagram



Gambar 12. Activity Diagram Buka Aplikasi Start Engine



Gambar 13. Activity Diagram Menyambungkan ke Bluetooth

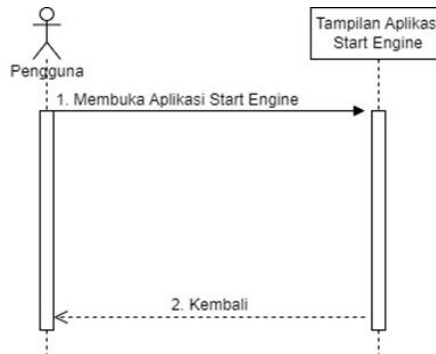


Gambar 14. Activity Diagram Menyalakan Start Engine

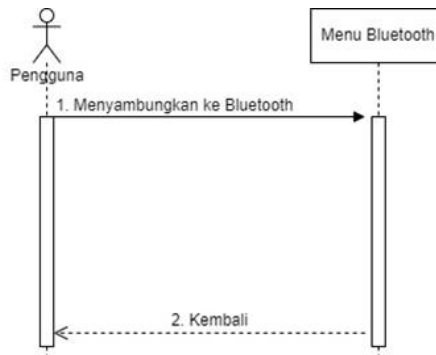


Gambar 15. Activity Diagram Menyalakan Alarm Sepeda Motor

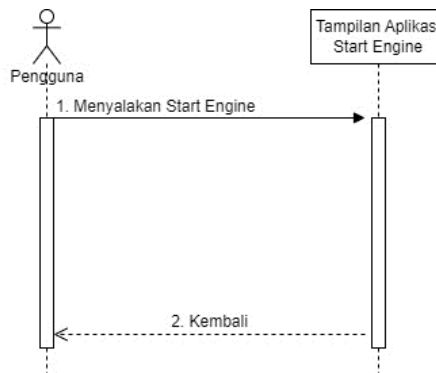
3.5.3 Sequence Diagram



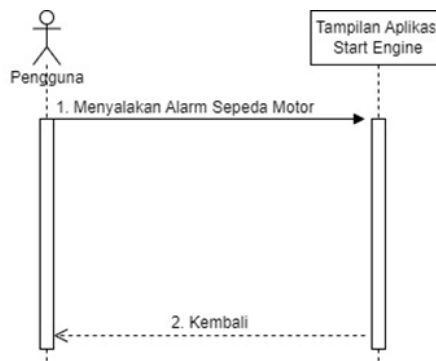
Gambar 16. Sequence Diagram Buka Aplikasi Start Engine



Gambar 17. Sequence Diagram Menyambungkan ke Bluetooth



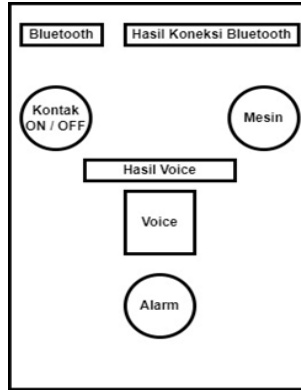
Gambar 18. Sequence Diagram Menyalakan Start Engine



Gambar 19. Sequence Diagram Menyalakan Alarm Sepeda Motor

3.5.4 Perancangan Tampilan *User Interface*

Perancangan user interface (*UI*) adalah proses merancang antarmuka pengguna yang menarik dan mudah digunakan untuk suatu sistem atau aplikasi. Tujuannya adalah untuk memastikan bahwa pengguna dapat berinteraksi dengan sistem atau aplikasi secara efektif, efisien, dan memuaskan.



Gambar 20. Tampilan Rancang *User Interface* Aplikasi *Start Engine*

Tabel 3. Deskripsi Tampilan *User Interface* Aplikasi *Start Engine*

Tombol	Keterangan
<i>Bluetooth</i>	Tombol <i>Bluetooth</i> berfungsi untuk mencari perangkat <i>Bluetooth</i> dan akan di tampilkan pada hasil koneksi <i>Bluetooth</i> jika berhasil tersambung oleh perangkat <i>Bluetooth</i> akan muncul tampilan <i>Connected</i> dan <i>Unconnected</i> jika tidak terhubung.
Kontak ON / OFF	Tombol Kontak ON / OFF berfungsi untuk menghidupkan dan mematikan kontak pada sepeda motor dan berfungsi untuk mematikan <i>alarm</i> keamanan jika kunci kontak sepeda motor dibobol dan juga berfungsi untuk mematikan mesin.
Mesin	Tombol Mesin berfungsi untuk menghidupkan mesin sepeda motor dan untuk mematikan mesin tekan tombol Kontak.
<i>Voice</i>	Tombol <i>Voice</i> berfungsi untuk menjalankan perintah dengan suara, seperti menghidupkan dan mematikan sepeda motor serta <i>alarm</i> , dan hasil perintah suara tersebut akan ditampilkan pada hasil voice
<i>Alarm</i>	Tombol <i>Alarm</i> berfungsi untuk mencari sepeda motor di dalam area parkir. Sedangkan untuk <i>alarm</i> keamanan pada sepeda motor akan berbunyi jika kunci kontak sepeda motor dibobol, dan untuk mematikan <i>alarm</i> keamanan tersebut dengan mengembalikan kontak ke posisi semula atau dengan mematikan <i>alarm</i> menggunakan aplikasi <i>start engine</i> di <i>smartphone android</i> dengan menekan tombol kontak.

3.5.5 Desain Aplikasi *Start Engine* Menggunakan *Software MIT App Inventor*

Untuk pembuatan desain aplikasi *Start Engine* ini menggunakan *Software MIT App Inventor* yang didesain sedemikian rupa sehingga menghasilkan tampilan aplikasi seperti gambar dibawah ini.



Gambar 21. Desain Aplikasi Menggunakan *Software MIT App Inventor*

3.5.6 Block Coding MIT App Inventor

Pembuatan block coding *start engine* pada *app inventor* ini digunakan sebagai fungsi utama dalam aplikasi *start engine*. Karena block ini sebagai bahasa program yang berhubungan dengan *Arduino*. Dan pembuatan Block Coding ini memberikan sebuah perintah yang dimasukkan ke dalam aplikasi supaya aplikasi dapat berjalan sesuai dengan yang diharapkan.



Gambar 22. Tampilan Block Coding MIT App Inventor

4. IMPLEMENTASI

4.1 Implementasi Perangkat Keras (*Hardware*)

Tabel 4. Spesifikasi Perangkat Keras (*Hardware*)

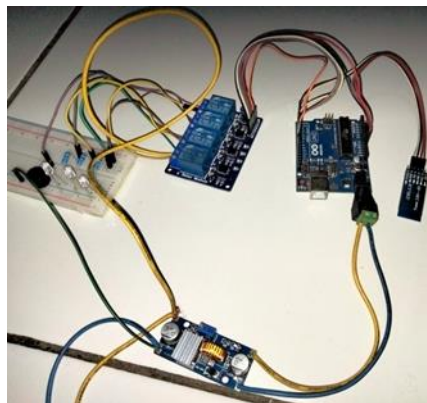
Perangkat Keras (<i>Hardware</i>)	Spesifikasi
Laptop HP 14-BW085TU / HP 14-BW086TU	Processor AMD A4-9120 RADEON R3, 4 COMPUTE CORES 2C+2G 2.20 GHz 4.00 GB RAM / 237 GB SSD Windows 10 Home Single Language 64-bit
Smartphone Android OPPO A53	Processor Qualcomm SM4250 Delapan-inti 4.00 GB RAM / 64.00 GB ROM Versi ColorOS V12.1 Versi Android 12
Arduino UNO	Mikrokontroler ATmega328P Tegangan Pengoperasian 5 Volt Tegangan Input (Rekomendasi) 7-12 Volt Batas Tegangan Input 6-20 Volt Pin I/O Digital 14 (6 diantaranya dapat di gunakan sebagai output PWM) Pin Digital PWM 6 Pin <i>Input</i> Analog 6 Arus DC Tiap Pin I/O 20 mA Arus DC untuk pin 3.3 Volt 50 mA Flash Memory 32 KB (ATmega328P) Sekitar 0.5 KB digunakan untuk bootloader SRAM 2 KB (ATmega328P) EEPROM 1 KB (ATmega328P) Clock Speed 16 MHz LED_BUILTIN 13 Panjang 68.6 mm Lebar 53.4 mm Berat 25 g
Modul Relay 4 Channel	Input relay 5 Volt DC Maksimum load 250 VAC/10A 30 VDC/10A

	<p>Dilengkapi dengan optocoupler isolation untuk melindungi board microcontroller dari tegangan AC</p> <p>Memiliki LED indikator</p> <p>Menggunakan terminal block sehingga pemasangan kabel menjadi mudah</p> <p><i>Output</i> keluaran 4 channel maksimal 10 Ampere</p> <p><i>Output</i> memiliki 3 pin terminal block yang ditandai dengan NO, COM dan NC.</p> <p>Keterangan:</p> <p>NO (<i>Normally Open</i>) = Tidak ada arus yang dialirkan (<i>OFF</i>), Jika ada signal <i>High / Low</i> dari <i>microcontroller</i> maka <i>ON</i>.</p> <p>COM (<i>Common</i>) = Sumber tegangan yang akan dihubungkan (Bisa arus AC maupun DC 10A max).</p> <p>NC (<i>Normally Close</i>) = Arus dialirkan (<i>ON</i>), Jika ada signal <i>High / Low</i> dari <i>microcontroller</i> maka <i>OFF</i>.</p>
<i>Module Bluetooth HC-05</i>	<p>Frekuensi kerja ISM 2.4 GHz</p> <p><i>Bluetooth protocol: Bluetooth</i> tipe v2.0+EDR</p> <p>Kecepatan dapat mencapai 1Mbps pada mode sinkron</p> <p>Kecepatan dapat mencapai 2.1 Mbps / 160 kbps pada mode asinkron maksimum</p> <p>Tegangan kerja pada 3,3 - 6 Volt DC</p> <p>Konsumsi arus kerja yaitu 50 mA</p> <p>Memiliki modulasi Gaussian Frequency Shift Keying (GFSK)</p> <p>Sensitivitas -84dBm (0.1% BER)</p> <p>Daya emisi 4 dBm</p> <p>Suhu operasional range -20°C — +75°C</p> <p>Memiliki keamanan dengan enkripsi data dan enkripsi</p> <p>Dimensi modul 15.2×35.7×5.6 mm</p> <p>Module ini dapat digunakan sebagai mode slave (Rx), maupun mode master (TX) dan memiliki 2 metode konfigurasi yaitu AT Mode dan Communication Mode.</p> <p>Pada AT Mode berfungsi sebagai pengaturan konfigurasi dari HC-05, sedangkan pada Communication Mode berfungsi sebagai komunikasi nirkabel dengan perangkat atau piranti lainnya.</p>
<i>Buzzer 12 Volt</i>	<p>Rated voltage = 12 Volt DC</p> <p>Operation voltage = 3-24 Volt DC</p> <p>Rated current = < 30mA</p> <p>Sound output = > 90 dB</p> <p>Resonant freq = 3000 +/- 500 Hz</p> <p>Operating temp = -20C s/d +60C</p> <p>Storage temp = -20C s/d +70C</p> <p>Diameter: 2.8 cm</p> <p>Tebal : 1.5 cm</p>
<i>BreadBoard</i>	<p>Distribution Strips are two</p> <p>Wire Size is 21 to 26 AWG wire</p> <p>Tie Points are two hundred</p> <p>Withstanding Voltage is 1,000 Volt AC</p> <p>Tie points within IC are 630</p> <p>Insulation Resistance is DC 500 Volt or 500MΩ</p> <p>Dimension is 6.5*4.4*0.3 inch</p> <p>Rating is 5Amps</p> <p>ABS plastic through color legend</p> <p>ABS heat Distortion Temperature is 183° F (84° C) Hole or Pitch Style is 2.54mm</p>
<i>Step Down DC</i>	<p>Bahan: papan PCB dan komponen elektronik</p> <p>Tegangan masukan: 5-40 Volt</p>

	<p>Tegangan keluaran: 1.2-35 Volt (terus menerus disesuaikan) Arus Output: 9 Ampere (suhu tabung daya melebihi 65 ° C, tambahkan kipas pendingin) Rentang arus konstan: 0.2-9Ampere (dapat disesuaikan) Daya Keluaran: Daya maksimum sekitar 300W (suhu tabung daya melebihi 65 °C, tambahkan kipas pendingin) Ukuran: Perkiraan. 65 x 48 x 24 mm</p>
<i>Flasher</i>	<p>Voltage 12 Volt Power 0.1-150 Watt Body Material Plastic Weight 1 8-20 Gm</p>
<i>Dioda IN5402</i>	<p>Arus max = 3 Ampere Tegangan max = 200 Volt</p>
<i>Jack DC Connector</i>	<p>Packing dimensions LxWxH - 36x135x12 mm Type - DC plug adapter Type - DC (male) 2-pin Cable type - DC connection cable</p>

4.2 Implementasi Alat

Implementasi alat merupakan tahap meletakkan sistem sehingga siap untuk dioperasikan. Implementasi bertujuan untuk mengkonfirmasi modul-modul perancangan, sehingga pengguna dapat melihat hasil dari alat yang dibuat.



Gambar 23. Implementasi Rangkaian Alat

4.3 Implementasi Perangkat Lunak (*Software*)

Tabel 5. Spesifikasi Perangkat Lunak (*Software*)

Perangkat Lunak (<i>Software</i>)	Keterangan
<i>Arduino IDE</i>	Sebuah perangkat lunak open-source yang berfungsi untuk menuliskan coding kemudian dimasukkan kedalam board. Versi yang digunakan adalah
<i>MIT App Inventor</i>	Merupakan <i>platform</i> berbasis <i>web</i> yang dapat digunakan untuk proses pembuatan aplikasi / perangkat lunak untuk <i>smartphone</i> .
<i>Draw.io</i>	Sebuah <i>website</i> yang didesain khusus untuk menggambarkan diagram secara online.
Aplikasi <i>Start Engine</i> dan <i>Alarm</i> Sepeda Motor	Aplikasi <i>Android</i> yang berfungsi sebagai <i>remote</i> untuk mengontrol <i>Start Engine</i> dan <i>Alarm</i> Sepeda Motor.

4.4 Implementasi Aplikasi

a. User Interface Aplikasi Start Engine



Gambar 24. User Interface Aplikasi Start Engine

b. User Interface Aplikasi Web Admin Start Engine



Gambar 25. User Interface Aplikasi Web Admin Start Engine




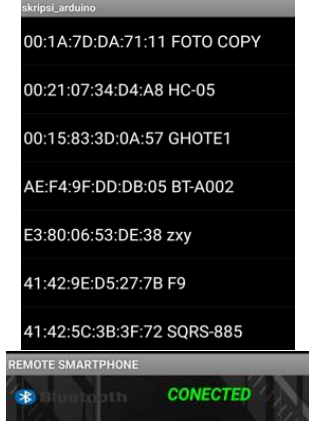

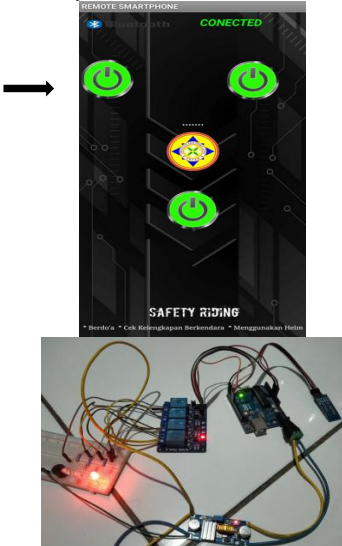
4.5 Pengujian Sistem





Pada pengujian ini menggunakan metode *Black Box Testing*, yaitu pengujian fungsional software tanpa mengetahui struktur internal program (Permatasari, 2020). Dengan kata lain pengujian ini tidak melibatkan struktur program yang ada pada aplikasi atau pun alat.




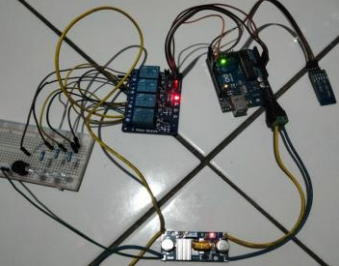


4.5.1 Black Box Testing

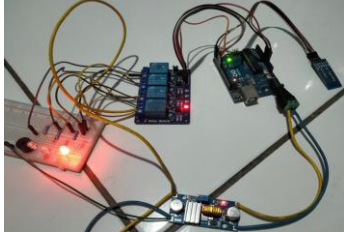


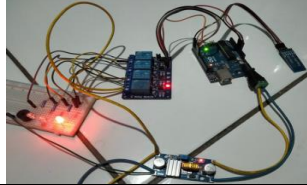
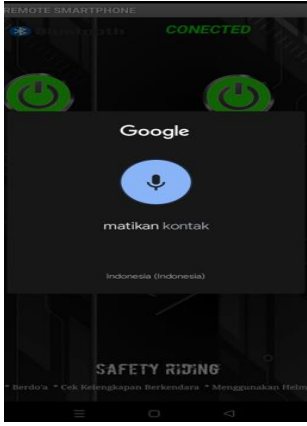

Tabel 6. Black Box Testing Aplikasi Start Engine Sepeda Motor

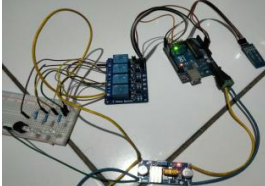
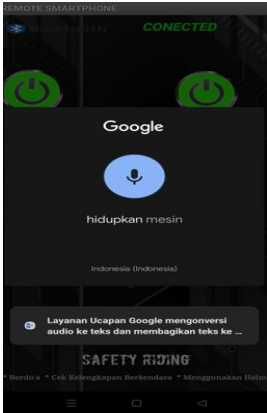

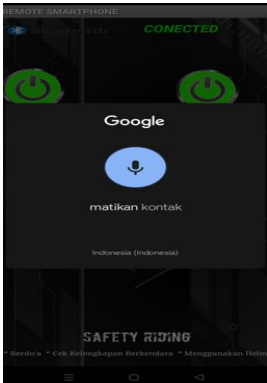

No	Skema Pengujian	Hasil pengujian yang diharapkan	Keterangan
1.	Menyandingkan <i>Bluetooth</i> pada perangkat <i>android</i> dengan modul <i>Bluetooth HC-05</i> menggunakan <i>Password</i> . <i>Test Case:</i>	Pada aplikasi <i>start engine</i> akan muncul perangkat <i>Bluetooth HC-05</i> .	Berhasil





			
<p>2.</p>	<p>Menyambungkan <i>Bluetooth</i> dari aplikasi ke alat dengan menekan tombol <i>Bluetooth</i> di aplikasi <i>start engine</i>.</p> <p><i>Test Case:</i></p> 	<p>Aplikasi <i>start engine</i> akan menampilkan daftar nama perangkat <i>Bluetooth</i>, dan memilih perangkat <i>Bluetooth</i> alat dengan nama <i>HC-05</i> untuk mengkoneksikan <i>Bluetooth</i>.</p> 	<p>Berhasil</p>
<p>3.</p>	<p>Menyalakan kontak sepeda motor dengan menekan tombol pada aplikasi <i>start engine</i>.</p> <p><i>Test Case:</i></p> 	<p>Sistem akan menyalakan kontak sepeda motor ditandai dengan warna tombol menjadi hijau di aplikasi dan juga muncul suara “kontak dinyalakan” dari aplikasi <i>start engine</i>, dan lampu <i>Relay IN1</i> menyala.</p> 	<p>Berhasil</p>

<p>4.</p>	<p>Mematikan kontak sepeda motor dengan menekan tombol pada aplikasi <i>start engine</i>. <i>Test Case:</i></p> 	<p>Sistem akan mematikan kontak sepeda motor ditandai dengan warna tombol menjadi merah di aplikasi dan juga muncul suara “kontak dimatikan” dari aplikasi <i>start engine</i>, dan lampu Relay IN1 mati.</p> 	<p>Berhasil</p>
<p>5.</p>	<p>Menyalakan mesin sepeda motor dengan menekan tombol pada aplikasi <i>start engine</i>. <i>Test Case:</i></p> 	<p>Sistem akan menyalakan mesin sepeda motor dan juga muncul suara “mesin dihidupkan” dari aplikasi <i>start engine</i>, lampu Relay IN2 dan IN3 menyala selama 1,5 detik.</p> 	<p>Berhasil</p>
<p>6.</p>	<p>Mematikan mesin sepeda motor dengan menekan tombol pada aplikasi <i>start engine</i>. <i>Test Case:</i></p>	<p>Sistem akan mematikan mesin sepeda motor ditandai dengan warna tombol menjadi merah di aplikasi dan juga muncul suara “kontak dimatikan” dari aplikasi <i>start engine</i>, dan lampu Relay IN1 mati.</p>	<p>Berhasil</p>

			
<p>7.</p>	<p>Menyalakan <i>alarm</i> parkir sepeda motor dengan menekan tombol pada aplikasi <i>start engine</i>. <i>Test Case:</i></p> 	<p>Sistem akan menyalakan <i>alarm</i> parkir sepeda motor dan juga muncul suara “<i>alarm</i> dinyalakan” dari aplikasi <i>start engine</i>, lampu <i>Relay</i> IN4 berkedip sebanyak 4 kali dan <i>Buzzer</i> berbunyi selama 5 detik.</p> 	<p>Berhasil</p>
<p>8.</p>	<p>Mematikan <i>alarm</i> keamanan sepeda motor jika kunci kontak sepeda motor dibobol dengan menekan tombol pada aplikasi <i>start engine</i>. <i>Test Case:</i></p> 	<p>Sistem akan mematikan <i>alarm</i> keamanan sepeda motor ditandai dengan warna tombol menjadi hijau di aplikasi <i>start engine</i> dan juga muncul suara “kontak dinyalakan” dari aplikasi, lampu <i>Relay</i> IN1 menyala dan <i>alarm</i> keamanan pun mati.</p> 	<p>Berhasil</p>

			
<p>9.</p>	<p>Menyalakan kontak sepeda motor dengan <i>voice</i> atau perintah suara “Nyalakan kontrak” pada aplikasi <i>start engine</i>. <i>Test Case:</i></p> 	<p>Sistem akan menyalakan kontak sepeda motor ditandai dengan warna tombol menjadi hijau di aplikasi dan juga muncul suara “kontak dinyalakan” dan hasil <i>voice</i> “Nyalakan kontak” dari aplikasi <i>start engine</i>, dan lampu Relay IN1 menyala.</p>  	<p>Berhasil</p>
<p>10.</p>	<p>Mematikan kontak sepeda motor dengan <i>voice</i> atau perintah suara “matikan kontrak” pada aplikasi <i>start engine</i>. <i>Test Case:</i></p> 	<p>Sistem akan mematikan kontak sepeda motor ditandai dengan warna tombol menjadi merah di aplikasi dan juga muncul suara “kontak dimatikan” dan hasil <i>voice</i> “matikan kontak” dari aplikasi <i>start engine</i>, dan lampu Relay IN1 mati.</p> 	<p>Berhasil</p>

			
<p>11.</p>	<p>Menyalakan mesin sepeda motor dengan <i>voice</i> atau perintah suara “hidupkan mesin” pada aplikasi <i>start engine</i>. <i>Test Case:</i></p> 	<p>Sistem akan menyalakan mesin sepeda motor dan juga muncul suara “mesin dihidupkan” dan hasil <i>voice</i> “hidupkan mesin” dari aplikasi <i>start engine</i>, lampu <i>Relay</i> IN2 dan IN3 menyala selama 1,5 detik.</p> 	<p>Berhasil</p>
<p>12.</p>	<p>Mematikan mesin sepeda motor dengan <i>voice</i> atau perintah suara “matikan kontak” pada aplikasi <i>start engine</i>. <i>Test Case:</i></p> 	<p>Sistem akan mematikan mesin sepeda motor ditandai dengan warna tombol menjadi merah di aplikasi dan juga muncul suara “kontak dimatikan” dan hasil <i>voice</i> “matikan kontak” dari aplikasi <i>start engine</i>, dan lampu <i>Relay</i> IN1 mati.</p> 	<p>Berhasil</p>

<p>13.</p>	<p>Menyalakan <i>alarm</i> parkir sepeda motor dengan <i>voice</i> atau perintah suara “parkir” pada aplikasi <i>start engine</i>. <i>Test Case:</i></p> 	<p>Sistem akan menyalakan <i>alarm</i> parkir sepeda motor dan juga muncul suara “<i>alarm</i> dinyalakan” dan hasil <i>voice</i> “parkir” dari aplikasi <i>start engine</i>, lampu <i>Relay</i> IN4 berkedip sebanyak 4 kali dan <i>Buzzer</i> berbunyi selama 5 detik.</p> 	<p>Berhasil</p>
<p>14.</p>	<p>Mematikan <i>alarm</i> keamanan sepeda motor jika kunci kontak sepeda motor dibobol dengan menggunakan <i>voice</i> pada aplikasi <i>start engine</i>. <i>Test Case:</i></p> 	<p>Sistem akan mematikan <i>alarm</i> keamanan sepeda motor ditandai dengan warna tombol menjadi hijau di aplikasi dan juga muncul suara “kontak dinyalakan” dan hasil <i>voice</i> “Nyalakan kontak” dari aplikasi <i>start engine</i>, lampu <i>Relay</i> IN1 menyala dan <i>alarm</i> keamanan pun mati.</p> 	<p>Berhasil</p>

5. KESIMPULAN

Kesimpulan yang didapatkan pada penelitian ini adalah Perancangan dari alat sistem *Start Engine* dan keamanan sepeda motor pada komunitas ojek online ini ialah menggunakan mikrokontroler *Arduino uno* dengan cara menciptakan rangkaian dari beberapa modul atau komponen-komponen tambahan diantaranya Modul *Bluetooth HC-05, Relay Module 4 Channel, Step Down DC, Buzzer, Jack DC Connector*. Kemudian sebagai alat pengendali atau remote menggunakan aplikasi yang sudah terpasang pada *smartphone android*. Selain itu, Pembuatan aplikasi *Start Engine* pada *smartphone android* menggunakan *Software MIT App Inventor*, pada aplikasi *start engine* tersebut didesain dan diprogram sedemikian rupa sehingga dapat berfungsi sebagai *remote* untuk mengontrol *Start Engine* dan *Alarm Sepeda Motor*.

REFERENCES

- Abraham Jason Santoso, N. (2021). *Seminar Nasional Hasil Riset Prefix-RTR SISTEM PENGAMAN SEPEDA MOTOR MENGGUNAKAN SIDIK JARI DAN GPS*. *Ciastech*, 513–520.
- Afiq, M., Industri, F. T., Sains, I., Moh, J., Ii, K., Jagakarsa, S. S., & Indonesia, J. S. (2018). *Perancangan Sistem Start & Pengaman Sepeda Motor Via Smartphone (Android) Berbasis Arduino Nano*. *XX*(3), 1–13.
- Ahmadar, M., Perwito, P., & Taufik, C. (2021). PERANCANGAN SISTEM INFORMASI PENJUALAN BERBASIS WEB PADA RAHAYU PHOTO COPY DENGAN DATABASE MySQL. *Dharmakarya*, *10*(4), 284. <https://doi.org/10.24198/dharmakarya.v10i4.35873>
- Ajar Rohmanu, A. (2018). *Www.Jurnal.Stmikcikarang.Ac.Id 7. Sistem Sensor Jarak Aman Pada Mobil Berbasis Mikrokontroler Arduino Atmega328*, *3*(1), 9.
- Alfina, O., & Harahap, F. (2019). Ommi Alfina , Fitriana Harahap. *Pemodelan Uml Sistem Pendukung Keputusan Dalam Penentuan Kelas Siswa Siswa Tunagrahita*, *3*(2), 143–150. <https://doi.org/10.46880/jmika.Vol3No2.pp143-150>
- Arifin, S. (2022). Pemanfaatan Fingerprint dan Voice Recognition Untuk Menghidupkan Sepeda Motor Berbasis Arduino. *Jurnal Ilmiah Teknologi Informasi Asia*, *16*(2), 123–132.
- Aryaviocholda, F., Hannats, M., Ichsan, H., & Budi, A. S. (2020). *Rancangan Sistem Pendeteksi Pencurian Helm Menggunakan Protokol MQTT Dan Bluetooth HC-05 Berbasis Arduino*. *4*(2), 517–525. <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/6969>
- Banjarnahor, J. (2020). Penerapan Metode Profile Matching dalam Sistem Pendukung Keputusan Penilaian Kinerja Karyawan (Studi Kasus : PT Jaya Prima Plastik). *Remik*, *5*(1), 47–53. <https://doi.org/10.33395/remik.v5i1.10655>
- Budiyanto, A., Pramudita, G. B., & Adinandra, S. (2020). Kontrol Relay dan Kecepatan Kipas Angin Direct Current (DC) dengan Sensor Suhu LM35 Berbasis Internet of Things (IoT). *Techne : Jurnal Ilmiah Elektroteknika*, *19*(01), 43–54. <https://doi.org/10.31358/techne.v19i01.224>
- Dharma, I. P. L., Tansa, S., & Nasibu, I. Z. (2019). Perancangan Alat Pengendali Pintu Air Sawah Otomatis dengan SIM800I Berbasis Mikrokontroler Arduino Uno. *Jurnal Teknik*, *17*(1), 40–56. <https://doi.org/10.37031/jt.v17i1.25>
- Hartanto, B., Anna, E. I., & Septiawan, R. N. (2021). Perancangan Aplikasi Sistem Informasi Pengelolaan Barang Inventaris Berbasis Android. *Jurnal Teknologi Dan Informatika (JEDA)*, *2*(2), 13–23.
- Ismail. (2021). *Tutorial Cascading Style Sheets (CSS)*. *1*(2), 24–31. <https://books.google.co.id/books?id=exclEAAAQBAJ>
- Kasrian, D., & Susanto, A. (2020). Rancang Bangun Autostarter Sepeda Motor Menggunakan Sidik Jari Berbasis Arduino. *GATOTKACA Journal (Teknik Sipil, Informatika, Mesin Dan Arsitektur)*, *1*(1), 58–70. <https://doi.org/10.37638/gatotkaca.v1i1.63>
- Khakim, L., Afriliana, I., Nurohim, N., & Rakhman, A. (2022). Proteksi Kebocoran Gas LPG Rumah Tangga Berbasis Mikrokontroler. *Komputika : Jurnal Sistem Komputer*, *11*(1), 40–47. <https://doi.org/10.34010/komputika.v11i1.4977>
- Londjo, M. F. (2021). Implementasi White Box Testing Dengan Teknik Basis Path Pada Pengujian Form Login. *Jurnal Siliwaangi*, *7*(2), 35–40.

- Masnur, M., Alam, S., & Fikri Nasir, M. (2021). Rancang Bangun Sistem Keamanan Motor Dengan Pengenalan Sidik Jari Berbasis Arduino Uno. *Jurnal Sintaks Logika*, 1(1), 2775–412. <https://jurnal.umpar.ac.id/index.php/sylog>
- Masripah, S., & Ramayanti, L. (2020). Penerapan Pengujian Alpha Dan Beta Pada Aplikasi Penerimaan Siswa Baru. *Swabumi*, 8(1), 100–105. <https://doi.org/10.31294/swabumi.v8i1.7448>
- Nurus Sholeh, Koko Joni, & Miftachul Ulum. (2020). Sistem Monitoring Kondisi Kendaraan Motor Injeksi Berbasis Mikrokontroler. *Jurnal JEETech*, 1(1), 37–42. <https://doi.org/10.48056/jeetech.v1i1.6>
- Pasaribu, F. I., & Efendi, I. R. & Y. (2019). Utilizing Exhaust Heat of Motorcycle As a Source. *JESCE (Journal of Electrical and System Control Engineering)*, 3(1), 13–29.
- Pela, M. F., & Pramudita, R. (2021). Sistem Monitoring Penggunaan Daya Listrik Berbasis Internet of Things Pada Rumah Dengan Menggunakan Aplikasi Blynk. *Infotech: Journal of Technology Information*, 7(1), 47–54. <https://doi.org/10.37365/jti.v7i1.106>
- Permatasari, D. I. (2020). Pengujian Aplikasi menggunakan metode Load Testing dengan Apache JMeter pada Sistem Informasi Pertanian. *Jurnal Sistem Dan Teknologi Informasi (JUSTIN)*, 8(1), 135. <https://doi.org/10.26418/justin.v8i1.34452>
- Pertiwi, I. P., Fedinandus, F., & Limantara, A. D. (2019). Sistem Pendukung Keputusan Penerima Program Keluarga Harapan (PKH) Menggunakan Metode Simple Additive Weighting. *CAHAYATEch*, 8(2), 182. <https://doi.org/10.47047/ct.v8i2.46>
- Prasetya, A. F., Sintia, S., & Putri, U. L. D. (2022). Perancangan Aplikasi Rental Mobil Menggunakan Diagram UML (Unified Modelling Language). *Jurnal Ilmiah Komputer ...*, 1(1), 14–18.
- Prasetya, D. I., & Mushlihudin, M. (2018). Sistem Keamanan Sepeda Motor Menggunakan Kata Sandi Berbasis Arduino Nano. *Jurnal Ilmiah Teknik Elektro Komputer Dan Informatika*, 4(1), 11. <https://doi.org/10.26555/jiteki.v4i1.8985>
- Raju Rizkyana, & Awang Surya. (2021). Sistem Keamanan Sepeda Motor Dengan Mengganti Saklar Starter Menggunakan Fingerprint. *JTTM : Jurnal Terapan Teknik Mesin*, 2(1), 43–51. <https://doi.org/10.37373/jttm.v2i1.90>
- Ridha Fauzi, M. (2020). Rancang Bangun Pengaman Sepeda Motor Menggunakan RFID Berbasis Arduino. *Jurnal Surya Teknika*, 7(2), 164–171. <https://doi.org/10.37859/jst.v7i2.2384>
- S. Samsugi, D. E. S. (2018). Purwarupa Controlling Box Pembersih Wortel Dengan Mikrokontroler. *Journal Artikel*, 2018(November), 1–7.
- Sari, R., & Hamidy, F. (2021). Sistem Informasi Akuntansi Perhitungan Harga Pokok Produksi Pada Konveksi Sjm Bandar Lampung. *Jurnal Teknologi Dan Sistem Informasi (JTSI)*, 2(1), 65–73. <http://jim.teknokrat.ac.id/index.php/JTSI>
- Setiawan, N. D. (2018). Otomasi Pencampur Nutrisi Hidroponik Sistem NTF (Nutrient Film Technique) Berbasis Arduino Mega 2560. *Jurnal Teknik Informatika Unika St. Thomas (JTIUST)*, 03(2), 78–82.
- Susanto, H., Nauri, I. M., & Sumarli. (2016). Hubungan Nilai Kapasitas Kapasitor Terhadap Frekuensi Kedipan Lampu Sein Pada Honda Supra X 125 Helm-In. *Jurnal Teknik Mesin*, 24(2), 1–5.
- Tharishny, S., Selvan, S., & Nair, P. (2016). Android based smart house control via wireless communication. *International Journal of Scientific Engineering and Technology*, (5), 5, 323–325.
- Wahid, A. A. (2020). Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi. *Jurnal Ilmu-Ilmu Informatika Dan Manajemen STMIK*, November, 1–5.
- Wijaya, T. K. (2019). Perancangan Panel Aotomatic Transfer Switch Dan Auotomatic Dengan Kontrol Berbasis Arduino Main Failure. *Sigma Teknika*, 2(2), 207. <https://doi.org/10.33373/sigma.v2i2.2058>