

SISTEM MONITORING PERAWATAN KENDARAAN BERBASIS INTERNET OF THINGS (IOT)

Muchsin Attubel¹⁾, Diky Siswanto²⁾, Mohammad Mukhsim³⁾

¹⁾Jurusan Teknik Elektro, Fakultas Teknik, Universitas Widyagama Malang.

Email: sintubleee@gmail.com,

^{2,3)}Jurusan Teknik Elektro, Fakultas Teknik, Universitas Widyagama Malang.

Email: dsiswanto@widyagama.ac.id

ABSTRAK

Meningkatnya kebutuhan akan informasi, pemantauan, dan pengendalian jarak jauh terhadap peralatan, industri, otomotif, dan bahkan kota membuat teknologi *Internet of Things* semakin dibutuhkan. Namun produk-produk dengan teknologi *Internet of Things* masih tergolong mahal. Hal ini menjadi tantangan bagi penulis untuk merancang dan mengimplementasikan *Internet of Things* pada sepeda motor. Dalam menerapkan *Internet of Things* pada kendaraan bermotor, ada beberapa parameter kendaraan yang akan dihubungkan ke Internet antara lain, indikator ganti oli, lokasi kendaraan, dan kondisi mesin *on/off*. Semua sensor pada setiap parameter terhubung ke mikrokontroler. Selanjutnya data yang diperoleh akan dikirim ke server. Data aktual kendaraan pada server akan ditampilkan pada aplikasi Android dalam bentuk *maps* dan simbol-simbol indikator. Tujuan dari penelitian ini adalah untuk merancang system monitoring perawatan kendaraan bermotor. Dari pengujian diperoleh *battery* aki kendaraan mampu mencatu daya *hardware* selama 2,33 hari. Selain itu diperoleh jarak antara titik acuan dengan lokasi terdeteksi memiliki selisih rata-rata sebesar 1,75m. Juga diperoleh tingkat keakuratan *global positioning system* (GPS) pada kendaraan (perangkat yang dirancang) lebih tinggi jika dibandingkan dengan GPS *smartphone*. Berdasarkan pengujian keseluruhan menunjukkan rata-rata *delay* pengiriman data dari kendaraan hingga sampai ke user adalah 1,36 detik. Secara keseluruhan perangkat yang dibuat berfungsi sesuai dengan hasil rancangan.

Kata kunci: monitoring kendaraan, perawatan kendaraan, *Internet of Things*, *global positioning system*

ABSTRACT

The growing need for information, monitoring and remote control of equipment, industry, automotive, and even cities makes the Internet of Things technology increasingly needed. But the products with Internet of Things technology are still quite expensive. It is a challenge for writers to design and implement the Internet of Things on motorcycles. In implementing the Internet of Things on motor vehicles, there are several parameters of vehicles that will Connected to the Internet, among others, oil change indicators, vehicle locations, and engine conditions on/off. All sensors on each parameter are connected to the microcontroller. Furthermore the data obtained will be sent to the server. The actual Data of the vehicle on the server will be displayed in the Android application in the form of maps and indicator symbols. The purpose of this research is to design a monitoring system for motor vehicle maintenance. From the test acquired battery vehicles capable of Hardware power supply for 2.33 days. In addition, the distance between reference points and detected locations has an average difference of 1, 75m. Also obtained the level of accuracy of global Positioning System (GPS) on the vehicle (device designed) higher when compared to the smartphone GPS. Based on the overall testing shows the average delay of sending data from a vehicle up to the user is 1.36 seconds. Overall device created

Keywords: *vehicle monitoring, vehicle maintenance, Internet of Things, Global Positioning System*

PENDAHULUAN

Perkembangan teknologi di bidang *Internet of Things* yang didefinisikan sebagai segala teknologi yang memperbolehkan jaringan computer untuk berkomunikasi dengan perangkat keras lainnya melalui jaringan Internet (termasuk juga perangkat keras komputer). Jaringan komputer dalam hal ini meliputi *wired* (kabel), *wireless* (nirkabel), *peer to peer* (P2P), *cloud computing*, bluetooth, sensor *network*, *wireless sensor network* (WNS) dan lain-lain (Siswanto, Zhang, Navaie, & Deepak, 2016). Dengan adanya *Internet of Things* ini, bukan hal yang mustahil lagi bagi anda untuk bias mengendalikan rumah anda dari jarak jauh, membuat sensor kebakaran/banjir, mengendalikan pesawat terbang, mengetahui kedatangan musuh, dan sebagainya, baik pada kehidupan sehari-hari maupun pada lingkungan militer atau kondisi perang (Eka, 2014). *Internet of Things* secara umum ialah suatu konsep yang dapat menghubungkan benda-benda di sekitar kita dengan jaringan Internet yang membuatnya bisa berkomunikasi antar benda maupun dengan penggunanya. Sehingga konsep ini akan memudahkan penggunaannya jika diterapkan di kehidupan nyata (Putra, 2018). Selain mempermudah dan mengurangi *human error*, teknologi *Internet of Things* bisa digunakan untuk meningkatkan keamanan rumah.

Salah satu bagian penting yang harus di perhatikan dalam perawatan kendaraan bermotor adalah waktu yang tepat untuk mengganti oli. Yang mana hal tersebut ditentukan dari perhitungan jarak tempuh kendaraan. Namun karena sering terjadinya kelupaan oleh pengguna, permasalahan tersebut mendorong penulis untuk merancang sistem monitoring dan pengingat waktu perawatan kendaraan bermotor.

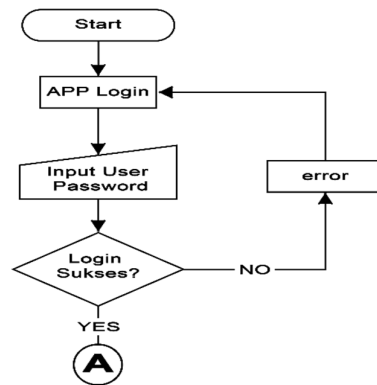
Artikel berikut memaparkan hasil penelitian dengan tema *Pengembangan Sistem Peringatan Ganti Oli Pada Sepeda Motor* (Trisetiyanto, 2012). Pada penelitian tersebut media pengingat yang digunakan berupa LCD yang menampilkan pesan peringatan dan lampu indikator yang diletakkan pada *dashboard* motor. Hal tersebut kurang efektif dikarenakan jaranganya pengguna memperhatikan dan mengecek tampilan pada *dashboard* motor.

Sementara itu, saat ini penggunaan *smartphone* terus meningkat bahkan dalam Laporan Internet Trends Kleiner Perkins Caufield & Byers's tersebut bahkan menyebutkan angka yang tinggi, yaitu pengguna rata-rata mengecek ponselnya 150 kali dalam sehari. Jika diakumulasi, dalam satu minggu rata-rata orang bisa menggunakan *smartphone*-nya lebih dari 1.050 kali (Gifary, 2015). Maka akan menjadi sangat tepat jika "sistem monitoring perawatan kendaraan" diterapkan pada *smartphone* sebagai *user interface*.

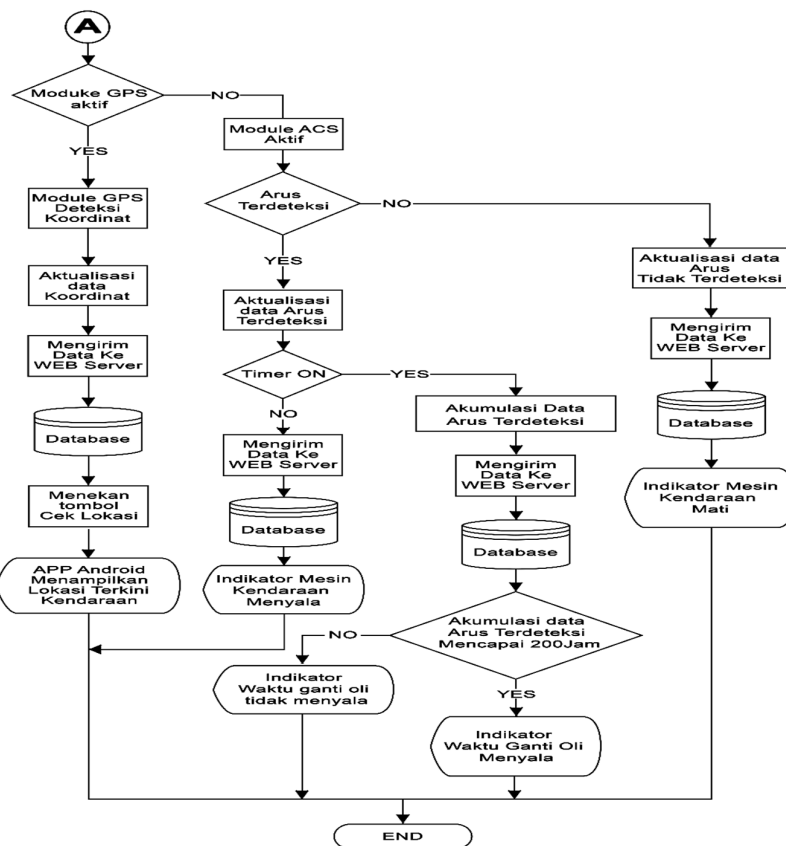
Terlambatnya melakukan perawatan pada kendaraan bermotor dapat mengakibatkan menurunnya kinerja kendaraan; bahkan bisa mengalami kerusakan. Sehingga *Internet of Things* adalah solusi tepat sebagai sistem pemantau serta pengingat waktu perawatan kendaraan bermotor. Yang mana perawatan kendaraan yang tepat waktu akan mencegah terjadinya kerusakan yang tidak diinginkan serta menghemat biaya perbaikan.

METODE PENELITIAN

Pada bagian ini diuraikan tentang disain sistem. Agar sejumlah parameter kendaraan dapat dipantau secara elektronik oleh pengguna, maka diterapkan teknologi *Internet of Things* dengan *smartphone* Android sebagai *user interface* (lihat Gambar 1).



Gambar 1. Diagram alir (flowchart).

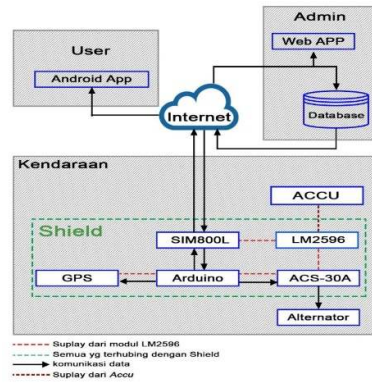


Gambar 2. Diagram alir (flowchart) Lanjutan

Rancangan Perangkat Keras

Rancangan perangkat keras terdiri dari 3 bagian yaitu pembuatan diagram blok, pembuatan *shield*, dan perakitan perangkat keras. Sebagaimana terdapat pada Gambar 2., diagram blok sistem terbagi menjadi 3 blok utama yaitu:

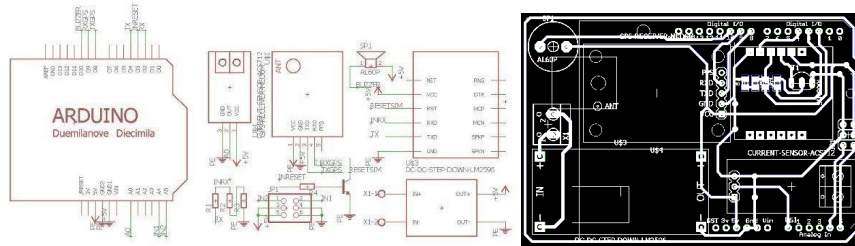
- Blok *User*: sebagai penerima informasi melalui *smartphone* Android yang terhubung ke Internet;
- Blok *Admin*: berupa *web server* yang hanya bisa diakses oleh admin; dan
- Blok kendaraan: berisi semua sensor, mikrokontroler, modul GSM sebagai penghubung ke Internet; dengan semua kebutuhan daya dari sejumlah komponen tersebut dicatu oleh *battery* aki kendaraan.



Gambar 3. Diagram Blok Sistem.

Pembuatan Shield Board

Pada tahap pembuatan *shield board* dimulai dengan mendesain skematik menggunakan *software* Eagle. Setelah menggambar skematik *shield board* pada Eagle, skematik diubah menjadi layout untuk dicetak pada *PCB board*. Gambar 3. memaparkan desain *shield board*.



a. Desain skematik *shield*;

b. Desain *layout* PCB;

Gambar 4. Desain *shield board*.

Perancangan Perangkat Lunak

Perancangan perangkat lunak sistem *monitoring* dan sistem pengingat waktu perawatan kendaraan terdiri dari 4 bagian yaitu desain *user interface*, pembuatan *block coding* Android App, pembuatan *web server*, dan pembuatan program mikrokontroler dengan Arduino IDE.

Perancangan Android App

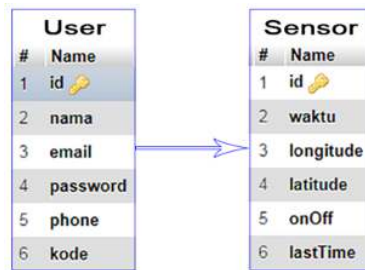
Desain *user interface* terdiri dari 5 form yaitu form registrasi, form login, form ganti password, form menu utama, dan form maps view. Gambar 4. menampilkan masing-masing *user interface form*.



Gambar 4. *User interface form*.

Perancangan Web Database

Database terdiri dari 2 tabel, yaitu tabel *user* dan tabel *sensor*. Gambar 5. menunjukkan struktur dari perancangan database.



Gambar 5. Struktur database.

Prosedur Pengujian

Pengujian dilakukan pada beberapa bagian sub-sistem, yaitu:

- a. Pengujian GPS
 Pengujian GPS dilakukan dengan membandingkan hasil koordinat yang terbaca oleh GPS kendaraan dan GPS *smartphone* Android.
- b. Pengujian konsumsi Daya *hardware*
 Pengujian konsumsi daya *hardware* dilakukan dengan cara mengukur kebutuhan daya *hardware* dan menghitung kemampuan *battery* aki dalam mencatu.
- c. Pengujian GSM
 Pengujian GSM dilakukan dengan mengukur *delay* waktu pengiriman dari kendaraan hingga ke user.

HASIL DAN PEMBAHASAN

Pengujian GPS

Untuk mengetahui jarak 2 lokasi yang berbeda di muka bumi dengan menggunakan data bujur dan lintang, maka dibutuhkan rumus haversine untuk menghitungnya, berikut adalah salah satu contoh perhitungannya:

Diketahui :

$$\begin{aligned}
 r &= 6371 \\
 \text{Lat1} &= -7.9741665 \\
 \text{Lon1} &= 112.6209697 \\
 \text{Lat2} &= -7.9741678 \\
 \text{Lon2} &= 112.62096
 \end{aligned}$$

Dicari:

$$\begin{aligned}
 d &= \text{Jarak} \\
 d &= 2r \cdot \left[\arcsin \sqrt{\sin^2 \left(\frac{\text{Lat2} - \text{Lat1}}{2} \right) + \cos(\text{Lat2}) \cdot \cos(\text{Lat1}) \cdot \sin^2 \left(\frac{\text{Lon2} - \text{Lon1}}{2} \right)} \right], \\
 d &= \\
 2r \cdot \left[\arcsin \sqrt{\sin^2 \left(\frac{-7.9741678 - (-7.9741665)}{2} \right) + \cos(-7.9741678) \cdot \cos(-7.9741665) \cdot \sin^2 \left(\frac{112.62096 - 112.6209697}{2} \right)} \right] \\
 d &= 2r \cdot \left[\arcsin \sqrt{(1.28701E - 16) + (0.990330715) \cdot (0.990330718) \cdot (7.16536E - 15)} \right], \\
 d &= 2r \cdot \left[\arcsin \sqrt{(1.28701E - 16) + (7.02747E - 15)} \right], \\
 d &= 2r \cdot \left[\arcsin \sqrt{(7.15617E - 15)} \right], \\
 d &= (2) \cdot (6371) \cdot \left[\arcsin(8.45941E - 08) \right], \\
 d &= 0.001077898 \text{ km} \cdot 1000, \\
 d &= 1.077898357 \text{ m}.
 \end{aligned}$$

Tabel 1. Perhitungan selisih jarak.

| Lokasi | GPS Android | | GPS Kendaraan | | Selisih (meter) |
|--------|-------------|-------------|---------------|-----------|-----------------|
| | Latitude | Longitude | Latitude | Longitude | |
| 1 | -7.9741665 | 112.6209697 | -7.9741678 | 112.62096 | 1.08 |
| | -7.9741699 | 112.6209191 | -7.9741673 | 112.62098 | 6.71 |
| | -7.9741396 | 112.6209346 | -7.9741731 | 112.62096 | 4.66 |
| | -7.9741639 | 112.6210488 | -7.9741716 | 112.62097 | 8.72 |
| | -7.974171 | 112.6210628 | -7.9741678 | 112.62096 | 11.33 |
| 2 | -7.9679155 | 112.6238845 | -7.9679561 | 112.62389 | 4.55 |
| | -7.9678887 | 112.6238801 | -7.9679503 | 112.62391 | 7.60 |
| | -7.9680461 | 112.6238736 | -7.9679575 | 112.62391 | 10.64 |
| | -7.9679499 | 112.6238514 | -7.9679456 | 112.6239 | 5.37 |
| | -7.9678802 | 112.623879 | -7.9679461 | 112.62389 | 7.43 |
| 3 | -7.9637233 | 112.6252193 | -7.9637032 | 112.62522 | 2.24 |
| | -7.9637389 | 112.6252546 | -7.9637079 | 112.62521 | 6.00 |
| | -7.9639702 | 112.6250977 | -7.9637051 | 112.62521 | 31.97 |
| | -7.9637165 | 112.6251237 | -7.9636974 | 112.62521 | 9.74 |
| | -7.9637666 | 112.6252083 | -7.9637117 | 112.62519 | 6.43 |
| 4 | -7.9638274 | 112.6219966 | -7.9637938 | 112.62205 | 6.97 |
| | -7.9638515 | 112.6220037 | -7.9637828 | 112.62196 | 9.03 |
| | -7.9638397 | 112.623396 | -7.9637771 | 112.62194 | 160.49 |
| | -7.9638727 | 112.621937 | -7.9637933 | 112.622 | 11.23 |
| | -7.9638252 | 112.6219469 | -7.9638019 | 112.622 | 6.40 |
| 5 | -7.9640039 | 112.6236069 | -7.9663424 | 112.61983 | 490.52 |
| | -7.9664332 | 112.6197917 | -7.9663315 | 112.61983 | 12.07 |
| | -7.9664031 | 112.6198107 | -7.9663396 | 112.61983 | 7.37 |
| | -7.9664461 | 112.6197681 | -7.9663544 | 112.61981 | 11.19 |
| | -7.9663626 | 112.6198579 | -7.9663754 | 112.6198 | 6.53 |

Dari Tabel diatas menunjukkan bahwa GPS pada *smartphone* dan GPS pada kendaraan memiliki tingkat keakuratan yang berbeda. Bahkan pada beberapa data yang diambil menunjukkan selisih jarak yang cukup jauh. Yang mana selisih jarak yang jauh terjadi karena GPS *smartphone* mendeteksi jarak lebih jauh dari lokasi sesungguhnya. Selisih jarak yang jauh ditunjukkan dengan *background* abu-abu pada Tabel.

Pengujian Konsumsi Daya Hardware

Pada pengujian ini bertujuan untuk mengetahui berapa lama *battery* aki mampu mencatu *hardware*, pengujian dilakukan dengan mengukur arus *hardware* lalu kapasitas Aki dibagi dengan arus yang dibutuhkan *hardware*.

Diketahui: Tegangan *battery* aki : 12 V.
 Kapasitas *battery* aki : 3.5 Ah.
 Arus *hardware* : 50 mA (0.05 A).

Waktu Pemakaian = kapasitas *battery* Aki (Ah) / arus dari *hardware* (A)
 = 3.5Ah / 0.05A = 70 jam.

Dengan efisiensi *battery* aki 20% (14 jam)
 = 70 jam - 14 jam = 56 jam,
 = 2.33 hari (2 hari, 7 jam, 55 menit, 12 detik).



Gambar 6. Pengukuran arus hardware.

Pengujian GSM

Pengujian ini dilakukan untuk mengetahui berapa *delay* pengiriman data menggunakan GSM dari kendaraan hingga ke *smartphone* Samsung Galaxy A3 2017. GSM akan mengirim data informasi kendaraan setiap 15 detik sekali. Pengujian dilakukan dengan menghidupkan kendaraan, lalu menghitung dengan *stopwatch* durasi waktu yang dibutuhkan untuk sampai pada *smartphone*. Tabel 2. menunjukkan hasil pengujian.

Tabel 2. Hasil pengujian pengiriman data ke Android.

| NO | Jarak (meter) | Total waktu kirim data dari kendaraan hingga ke Android (detik) | <i>Delay</i> (detik) |
|-------------------------------|---------------|---|----------------------|
| 1 | 1 | 16.95 | 1.95 |
| | | 16.30 | 1.3 |
| | | 16.02 | 1.02 |
| | | 17.27 | 2.27 |
| | | 15.67 | 0.67 |
| 2 | 100 | 16.09 | 1.09 |
| | | 17.19 | 2.19 |
| | | 16.87 | 1.87 |
| | | 15.60 | 0.6 |
| | | 15.58 | 0.58 |
| 3 | 200 | 15.34 | 0.34 |
| | | 16.13 | 1.13 |
| | | 17.06 | 2.06 |
| | | 16.68 | 1.68 |
| | | 16.61 | 1.61 |
| 4 | 300 | 15.86 | 0.86 |
| | | 16.59 | 1.59 |
| | | 15.38 | 0.38 |
| | | 18.13 | 3.13 |
| | | 16.07 | 1.07 |
| 5 | 400 | 15.80 | 0.8 |
| | | 15.98 | 0.98 |
| | | 16.79 | 1.79 |
| | | 17.30 | 2.3 |
| | | 15.78 | 0.78 |
| Rata-rata <i>delay</i> | | | 1.36 |
| <i>Delay</i> terlama | | | 3.13 |
| <i>Delay</i> Tercepat | | | 0.34 |

Hasil pengujian diatas menunjukkan rata-rata *delay* 1.36 detik, *delay* terlama 3.13 detik, dan *delay* tercepat adalah 0.34 detik. Selanjutnya, pengujian menyeluruh terhadap perangkat yang dirancang menunjukkan hasil sesuai dengan rancangan.

KESIMPULAN

Secara umum perangkat yang dirancang bisa beroperasi dan berfungsi sesuai dengan hasil perencanaan. Berdasarkan pengujian, selisih jarak antara titik acuan dengan lokasi terdeteksi memiliki nilai rata-rata sebesar 1.75 m. Selain itu didapat tingkat keakuratan GPS pada perangkat yang dirancang lebih tinggi jika dibandingkan dengan GPS *smartphone*, yang dalam pengujian menggunakan Samsung Galaxy A3 2017. Analisis konsumsi daya *hardware* menunjukkan kemampuan *battery* aki kendaraan dapat mencatu *hardware* selama 2.33 hari (2 hari, 7jam , 55 menit, 12 detik).

REFERENSI

- Cahyono, G. H. (2016). Internet Of Things (Sejarah,Teknologi dan Penerapannya). *Forum Teknologi*, 35-41.
- Eka, P. A. (2014). *Smart City beserta Cloud Computing*. Bandung: Informatika Bandung.
- Gifary, S. (2015). Intensitas Penggunaan Smartphone Terhadap Perilaku Komunikasi. *Jurnal Sositologi*, 2, 170.
- Kusnandar, F. D. (2015). Identifikasi Sistem Pengisian Pada Mobil Toyota Kijang Innova 1TR-FE. *Skripsi*, 6.
- Putra, F. H. (2018). Aplikasi IoT untuk Rumah Pintar dengan Fitur Prediksi Cuaca. *Ind. Journal on Computinng*, 5, 1746.
- Safaat, N. (2014). *Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung: Informatika Bandung.
- Sakti, R. (2017). Sistem Keamanan Motor Menggunakan Sensor MPU 6050 dan Tracking Lokasi Dengan GPS Android. *Skripsi*, 11.
- Saputra, I. A. (2017). Aplikasi Layanan Bengkel Mobil Berbasis Android di Kota Bandar Lampung. *Skripsi Ilmu Komputer*, 15.
- Silvia, A. F. (2014). Rancang Bangun Akses Kontrol Pintu Gerbang Berbasis Arduino dan Android. *Electrans*, 13, 1-10.
- Siswanto, D., Zhang, L., Navaie, K., & Deepak, G. (2016). Weighted sum throughput maximization in heterogeneous OFDMA network. *IEEE 83rd Vehicular Technology Conference (VTC-Spring)*, 1-5.
- Trisetiyanto, A. N. (2012). Pengembangan Sistem Peringatan Ganti Oli Pada Sepeda Motor. *Jurnal Teknik Elektro*, 1-6.
- Wicaksono, M. F., & Hidayat. (2017). *Mudah Belajar Mikrokontroler Arduino*. Bandung: Informatika Bandung.