

SKRIPSI
RANCANG BANGUN MOBILE ROBOT DISINFEKTAN
MENGGUNAKAN UVC (ULTRAVIOLET C) BERBASIS IOT (INTERNET
OF THINGS)

Disusun dan Diajukan oleh:

RAYNALDI S. KUMAR

D021171525



DEPARTEMEN TEKNIK MESIN

FAKULTAS TEKNIK

UNIVERSITAS HASANUDDIN

GOWA

2022

SKRIPSI
RANCANG BANGUN MOBILE ROBOT DISINFEKTAN
MENGGUNAKAN UVC (ULTRAVIOLET C) BERBASIS IOT (INTERNET
OF THINGS)

Disusun dan Diajukan oleh:

RAYNALDI S. KUMAR
D021171525

Merupakan salah satu syarat untuk memperoleh gelar Sarjana Teknik
Mesin pada Fakultas Teknik Universitas Hasanuddin

DEPARTEMEN TEKNIK MESIN

FAKULTAS TEKNIK

UNIVERSITAS HASANUDDIN

GOWA

2022

LEMBAR PENGESAHAN

Tugas akhir ini diajukan untuk memenuhi salah satu persyaratan Mengikuti Ujian Akhir guna memperoleh gelar Sarjana Teknik Mesin pada Departemen Mesin Fakultas Teknik Universitas Hasanuddin

JUDUL:

**RANCANG BANGUN MOBILE ROBOT DISINFEKTAN
MENGUNAKAN UVC (ULTRAVIOLET C) BERBASIS IOT (INTERNET
OF THINGS)**

RAYNALDI S. KUMAR

D021171525

Telah diperiksa dan disetujui oleh:

Gowa, 30 Mei 2022

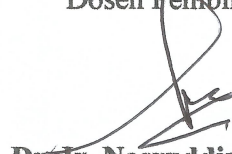
Dosen Pembimbing I



Dr. Eng. Andi Amijoyo Mochtar, ST., MSc

NIP. 19760216 201012 1 002

Dosen Pembimbing II



Dr. Ir. Nasruddin Azis., M.Si

NIP. 19611017 198503 1 004

Mengetahui,

Ketua Departemen Teknik Mesin Fakultas Teknik
Universitas Hasanuddin



Dr. Eng. Jalaluddin, ST., MT

NIP. 19720825 200003 1 001

PERNYATAAN KEASLIAN

Saya yang bertanda tangan dibawah ini:

Nama : Raynaldi S. Kumar
NIM : D021171525
Program Studi : Teknik Mesin
Jenjang : S-1
Judul Skripsi : Rancang Bangun Mobile Robot Disinfektan Menggunakan UVC (Ultraviolet C) Berbasis IoT (Internet of Things)

Menyatakan dengan sesungguhnya bahwa Skripsi ini merupakan hasil penelitian, pemikiran, dan pemaparan asli dari saya sendiri. Saya tidak mencantumkan tanpa pengakuan bahan-bahan yang telah dipublikasikan sebelumnya atau ditulis oleh orang lain atau sebagai bahan yang pernah diajukan untuk gelar atau ijazah pada Universitas Hasanuddin atau perguruan tinggi lainnya.

Apabila dikemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik sesuai dengan peraturan yang berlaku di Universitas Hasanuddin

Demikian pernyataan ini saya buat

Gowa, 30 Mei 2022

Yang membuat pernyataan



Raynaldi S. Kumar

ABSTRAK

Ultraviolet C (UVC) merupakan salah satu alat disinfektan yang terbukti efektif dalam membunuh mikroorganisme, salah satunya adalah virus COVID-19. Dengan dosis $250\text{J}/\text{m}^2$ terbukti dapat menurunkan populasi virus COVID-19 pada permukaan sebesar 99,99%. Pada penelitian ini telah dibuat rancangan mobile robot disinfektan menggunakan lampu UVC yang dapat dikontrol melalui internet. Penelitian ini bertujuan untuk merencanakan pembuatan mobile robot disinfektan menggunakan UVC berbasis IOT, mengetahui perbandingan kecepatan respon mobile robot melalui internet dengan provider internet yang berbeda, mengetahui Jarak dan lama penyinaran UVC agar efektif dalam melakukan sterilisasi ruangan. Desain mobile robot ini menggunakan 2 buah lampu UVC 8W, ESP-32 sebagai kontrol utama dan menggunakan server dari DIGITALOCEAN sebagai pusat kontrol. Hasil penelitian untuk pengujian respon time rata-rata tiap provider diperoleh antara lain Telkomsel 0,491s, Indosat 0,554s, XL 0,622s dan Tri 1,022s. Untuk pengujian intensitas UVC, hasil terbaik berada pada arah depan mobile robot dengan nilai $2,78\text{W}/\text{m}^2$ pada jarak 0,05m sedangkan untuk jarak 0,5m nilai intensitas sebesar $0,17\text{W}/\text{m}^2$. Untuk lama penyinaran UVC berdasarkan intensitas yang didapat diatas dan nilai dosis $250\text{j}/\text{m}^2$ didapat hasil terbaik pada arah depan mobile robot dengan waktu 89 detik pada jarak 0,05m dan waktu 1470 detik pada jarak 0,5m.

Kata kunci: UVC, mobile robot, IoT, COVID-19, disinfektan

ABSTRACT

Ultraviolet C (UVC) is a disinfectant tool that has been proven effective in killing microorganisms, one of which is the COVID-19 virus. With a dose of 250J/m², it has been shown to reduce the population of the COVID-19 virus on surfaces by 99.99%. In this research, a mobile robotic disinfectant has been designed using UVC lamps that can be controlled via the internet. This study aims to design the manufacture of a mobile robot disinfectant using IOT-based UVC, to compare the response speed of a mobile robot via the internet with different internet providers, to determine the distance and duration of UVC irradiation to be effective in sterilizing the room. This mobile robot design uses 2 8W UVC lamps, ESP-32 as the main control and uses a server from DIGITALOCEAN as the control center. The results for testing the average response time of each provider were obtained, including Telkomsel 491.05ms, Indosat 554.95ms, XL 622.48ms and Tri 1022.63ms. For UVC intensity testing, the best results are in the forward direction of the mobile robot with a value of 2.78W/m² at a distance of 0.05m, while for a distance of 0.5m the intensity value is 0.17W/m². For the duration of UVC irradiation based on the intensity obtained above and the dose value of 250j/m², the best results were obtained at the front of the mobile robot with a time of 89 seconds at a distance of 0.05m and a time of 1470 seconds at a distance of 0.5m.

Keywords: UVC, mobile robot, IoT, COVID-19, disinfectant

KATA PENGANTAR

Puji dan syukur senantiasa kita panjatkan ke-hadirat Tuhan Yang Maha Esa karena atas rahmat dan berkat-Nya penulis dapat menyelesaikan skripsi dengan judul “Pengaruh Variasi Laju Alir Udara terhadap Efisiensi Termal pada Proses Gasifikasi dengan Sistem Updraft” yang mana merupakan salah satu syarat untuk memperoleh gelar sarjana teknik pada Departemen Teknik Mesin Fakultas Teknik Universitas Hasanuddin. Selama proses pengerjaan skripsi ini penulis menerima begitu banyak bantuan dari berbagai pihak. Untuk itu peneliti ingin mengucapkan terimakasih kepada:

1. Kedua orang tua tercinta Bapak dan Ibu yang selalu mendampingi, memberi semangat dan mendoakan.
2. Dr.Eng. Jalaluddin, ST., MT. sebagai Ketua Departemen Teknik Mesin Fakultas Teknik Universitas Hasanuddin beserta seluruh staf Departemen Teknik Mesin Fakultas Teknik Universitas Hasanuddin atas segala bantuan dan kemudahan yang diberikan
3. Dr.Eng. Andi Amijoyo Mochtar, ST., MSc dan Dr. Ir. Nasruddin Azis., M.Si. sebagai Dosen Pembimbing I dan Dosen Pembimbing II yang telah memberikan waktu, arahan, dan saran selama proses pengerjaan skripsi ini.
4. Ir. Mukhtar Rahman, MT. dan Arfandi, ST., MT. selaku penguji yang telah memberikan saran-saran selama proses pengerjaan skripsi.
5. Dr. Eng Lukmanul Hakim Arma, ST., MT. sebagai Dosen Pembimbing Akademik yang telah memberikan ilmu dan nasehat sejak menjadi mahasiswa baru.
6. Bapak/Ibu dosen Departemen Teknik Mesin Universitas Hasanuddin yang telah memberikan ilmu, nasehat dan pengalaman kepada penulis selama menempuh studi di dunia perkuliahan
7. Teman-teman Teknik Mesin angkatan 2017 / ZYNCROMEZH'17 yang senantiasa mendukung dan berjuang bersama sejak mahasiswa baru hingga saat ini.

8. Serta seluruh pihak yang telah membantu yang tidak bisa disebutkan satu per satu.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna walaupun telah menerima bantuan dari berbagai pihak. Apabila terdapat kesalahan-kesalahan dalam skripsi ini sepenuhnya menjadi tanggung jawab penulis dan bukan para pemberi bantuan. Kritik dan saran yang membangun akan lebih menyempurnakan skripsi ini.

Gowa, 30 Mei 2022

Penulis

DAFTAR ISI

| | |
|---|-----|
| SAMPUL | i |
| HALAMAN JUDUL | ii |
| LEMBAR PENGESAHAN | iii |
| PERNYATAAN KEASLIAN | iv |
| ABSTRAK | v |
| ABSTRACT | vi |
| KATA PENGANTAR | vii |
| DAFTAR ISI | ix |
| DAFTAR GAMBAR | xi |
| DAFTAR TABEL | xiv |
| DAFTAR SIMBOL | xv |
| DAFTAR LAMPIRAN | xvi |
| BAB I PENDAHULUAN | 1 |
| 1.1. Latar Belakang | 1 |
| 1.2. Rumusan Masalah | 2 |
| 1.3. Tujuan Penelitian | 3 |
| 1.4. Batasan Masalah | 3 |
| 1.5. Manfaat Penelitian | 4 |
| BAB II TINJAUAN PUSTAKA | 5 |
| 2.1. Mobile Robot | 5 |
| 2.2. Internet of Things (IoT) | 10 |
| 2.3. UVC (Ultraviolet C) | 12 |
| BAB III METODOLOGI PENELITIAN | 15 |
| 3.1. Waktu dan Tempat Penelitian | 15 |
| 3.2. Desain Mobile Robot | 15 |
| 3.3. Desain Perangkat Elektronik | 16 |
| 3.4. Desain Perangkat Lunak | 27 |
| 3.5. Alat dan Bahan | 28 |
| 3.6. Metode Pengambilan Data | 30 |

| | |
|---|----|
| 3.7. Flow Chart Penelitian | 23 |
| BAB IV HASIL DAN PEMBAHASAN | 31 |
| 4.1. Desain dan Pembuatan Mobile Robot Disinfektan UVC | 31 |
| 4.2. Data Hasil Pengujian | 31 |
| BAB V PENUTUP | 48 |
| 5.1 Kesimpulan | 48 |
| 5.2 Saran | 48 |
| DAFTAR PUSTAKA | 49 |
| LAMPIRAN | 51 |

DAFTAR GAMBAR

| | |
|--|----|
| Gambar 2.1. Orientasi Posisi Mobile Robot Dalam Sistem Koordinat Cartesia.. | 7 |
| Gambar 2.2. Ilustrasi pergerakan robot berdasar titik pusat rotasi | 8 |
| Gambar 2.3. Pinout ESP32 | 9 |
| Gambar 2.4. Lampu UVC | 12 |
| Gambar 2.5. Spektrum lampu UVC dan cara kerja UVC terhadap DNA | 13 |
| Gambar 3.1. Skema Desain Mobile Robot Disinfektan UVC | 15 |
| Gambar 3.2. Skema Rangkaian Elektronik Mobile Robotic | 16 |
| Gambar 3.3. Sasis dasar robot | 17 |
| Gambar 3.4. Motor DC | 17 |
| Gambar 3.5. ESP-32 | 18 |
| Gambar 3.6. ESP-32CAM | 18 |
| Gambar 3.7. Lampu UVC | 18 |
| Gambar 3.8. Baterai Li-ion 18650 | 19 |
| Gambar 3.9. Motor Driver L298N..... | 19 |
| Gambar 3.10. Modul XH-M603 | 19 |
| Gambar 3.11. Modul BMS | 20 |
| Gambar 3.12. Sensor ML8511 | 20 |
| Gambar 3.13. Inverter 40W | 20 |
| Gambar 3.14. Modul LM2596 | 21 |
| Gambar 3.15. Modul Relay | 21 |
| Gambar 3.16. Modem | 21 |
| Gambar 3.17. Diagram Alir Penelitian Mobile Robot | 23 |
| Gambar 4.1. Tampilan perangkat lunak Fusion 360 | 24 |
| Gambar 4.2. Proses pencetakan salah satu body menggunakan Printer 3D..... | 25 |
| Gambar 4.3. <i>Motor DC housing</i> | 25 |
| Gambar 4.4. Casis mobile robot | 26 |
| Gambar 4.5. Body cover mobile robot | 26 |
| Gambar 4.6. Camera Housing | 26 |
| Gambar 4.7. ESP-32 Wroom..... | 27 |

| | |
|---|----|
| Gambar 4.8. Modul BMS (kiri) dan Baterai 18650 (kanan) | 28 |
| Gambar 4.9. Kamera ESP-32 | 29 |
| Gambar 4.10. Modul motor driver L298N | 29 |
| Gambar 4.11. Inverter DC-AC 40w | 29 |
| Gambar 4.12. Modul Relay | 30 |
| Gambar 4.13. Lampu UVC | 30 |
| Gambar 4.14. Modul LM2596 | 30 |
| Gambar 4.15. Modul XH-M603 | 31 |
| Gambar 4.16. Motor DC JGA25-370 | 31 |
| Gambar 4.17. Skema Rangkaian Elektronik Mobile Robot | 32 |
| Gambar 4.18. Tampilan dashboard Digitalocean | 33 |
| Gambar 4.19. Tampilan website kontrol mobile robot | 33 |
| Gambar 4.20. Tunneling EPS-32 CAM melalui Ngrok | 34 |
| Gambar 4.21. Tampilan software Arduino IDE | 35 |
| Gambar 4.22. Diagram Alir Program Mobile Robot | 36 |
| Gambar 4.23. Respon time (delay) provider Telkomsel | 37 |
| Gambar 4.24. Respon time (delay) provider Tri | 38 |
| Gambar 4.25. Respon time (delay) provider XL | 38 |
| Gambar 4.26. Respon time (delay) provider Indosat | 39 |
| Gambar 4.27. Respon time (delay) tiap provider | 40 |
| Gambar 4.28. Jarak terhadap intensitas lampu UVC pada arah depan | 41 |
| Gambar 4.29. Jarak terhadap intensitas lampu UVC pada arah kiri | 41 |
| Gambar 4.30. Jarak terhadap intensitas lampu UVC pada arah belakang | 42 |
| Gambar 4.31. Jarak terhadap intensitas lampu UVC pada arah kanan | 43 |
| Gambar 4.32. Jarak terhadap lama waktu penyinaran lampu UVC pada arah depan..... | 44 |
| Gambar 4.33. Jarak terhadap lama waktu penyinaran lampu UVC pada arah kiri..... | 45 |
| Gambar 4.34. Jarak terhadap lama waktu penyinaran lampu UVC pada arah belakang..... | 45 |

| | |
|--|----|
| Gambar 4.35. Jarak terhadap lama waktu penyinaran lampu UVC pada arah kanan..... | 46 |
| Gambar 4.36 Uji dosis sinar UVC..... | 47 |

DAFTAR TABEL

| | |
|---|----|
| Tabel 2.1. Efektivitas Dosis Ultraviolet terhadap Coronavirus..... | 14 |
|---|----|

DAFTAR SIMBOL

| | | |
|----------|------------------------|----------------|
| <i>I</i> | Intensitas ultraviolet | W/m^2 |
| <i>D</i> | Dosis UV | J/m^2 |
| <i>t</i> | Waktu paparan | s |

DAFTAR LAMPIRAN

| | |
|---|----|
| LAMPIRAN I Program pada ESP-32, ESP32-CAM, dan Website | 51 |
| LAMPIRAN II Foto Kegiatan..... | 67 |

BAB I

PENDAHULUAN

1.1. Latar Belakang

Wabah corona virus yang disebabkan oleh virus SARS-Cov2 telah menyebar ke seluruh dunia. Penyebaran di Indonesia sendiri pertama kali terdeteksi sejak Februari 2020, salah satu warga negara Jepang terbukti menjadi pembawa virus akibat dari aktivitas pertemuan pesta dansa di Indonesia sendiri. Diyakini bahwa penyebaran virus corona pertama kali terjadi di pasar makanan laut di Wuhan dan terus menyebar melalui kontak orang ke orang (Adhikari et al., 2020).

Virus corona dapat menyebar melalui berbagai cara. Manusia merupakan salah satu sumber utama penularan virus melalui droplet atau partikel yang disebabkan oleh batuk atau bersin. Selain itu, media lain seperti logam, kertas, dan kaca juga bisa menjadi tempat penyebaran virus corona. Namun, karena virus corona sensitif terhadap panas, disinfektan mengandung klorin dan pelarut lemak pada suhu 56°C selama 30 menit, eter, alkohol, asam perasetat, formalin, oksidan dan kloroform dapat membunuh virus corona (Wang et al., 2019).

Untuk alasan ini, banyak disinfektan aerosol digunakan untuk mencegah penularan virus corona. Selama wabah virus corona, penggunaan disinfektan yang disemprotkan pada manusia atau lingkungan di mana virus dapat menyebar sangat populer. Namun, Kementerian Kesehatan RI telah melarang penggunaan disinfektan aerosol secara terus-menerus sebagai tanggapan atas segala aspek yang mengancam kesehatan manusia, terutama bahaya pada kulit dan selaput lendir. Meskipun berbahaya, penggunaan disinfektan dapat digunakan untuk sterilisasi ruangan (Cahyo, 2020).

Mobile robot adalah jenis robot yang memiliki ciri khas berupa aktuator berbentuk roda yang berfungsi sebagai penggerak utamanya. Berdasarkan

sistem penggeraknya, robot beroda diklasifikasikan menurut jumlah roda yang digunakan. Salah satunya adalah robot beroda yang sistem gerak differential drivenya terdiri dari dua buah roda yang dipasangkan pada bagian kiri dan kanan robot. Sistem ini memungkinkan robot berputar di tempat dengan memutar motor ke arah yang berlawanan (Eko, 2017).

Internet of Things adalah salah satu teknologi yang dapat mengontrol perangkat elektronik yang menggunakan Internet sebagai penghubung antara perangkat dan pengguna. Internet of Things mengubah objek tradisional ini menjadi perangkat pintar melalui penggunaan teknologi yang semakin berkembang, peralatan yang ada, teknologi komunikasi, jaringan sensor, protokol dan jaringan Internet. (Faiz, 2018).

Dengan metode ini memungkinkan seseorang bisa mengendalikan suatu alat/objek baik fisik maupun virtual, dalam hal ini adalah robot disinfektan, dari jarak jauh menggunakan koneksi jaringan internet. Hal ini semakin memudahkan kita untuk memonitoring dan mengendalikan robot disinfektan di manapun kita berada selama kita terhubung dengan koneksi internet.

Pada penelitian yang kami lakukan akan fokus pada Rancang Bangun Mobile Robot Disinfektan Menggunakan UVC (Ultraviolet C) Berbasis IoT (Internet of Things) dengan alasan untuk meningkatkan kemampuan penggunaan robot disinfektan yang terhubung dengan jaringan internet.

1.2. Rumusan Masalah

Berdasarkan latar belakang penelitian, maka dapat dirumuskan beberapa permasalahan yang dapat dibahas dalam penelitian ini untuk membatasi apa yang akan dibahas dalam inti penelitian ini. Beberapa masalah tersebut sebagai berikut.

1. Bagaimana merancang pembuatan mobile robot disinfektan menggunakan UVC berbasis IOT?
2. Bagaimana perbandingan kecepatan respon mobile robot melalui internet dengan provider internet yang berbeda?

3. Berapa efektifitas jarak dan durasi waktu penyinaran UVC dalam melakukan sterilisasi ruangan?

1.3. Tujuan Penelitian

Berdasarkan rumusan masalah di atas, ada beberapa tujuan dari penelitian ini antara lain:

1. Merancang dan membuat mobile robot disinfektan menggunakan UVC berbasis IOT.
2. Mengetahui perbandingan kecepatan respon mobile robot melalui internet dengan provider internet yang berbeda.
3. Mengetahui efektifitas jarak dan durasi waktu penyinaran UVC dalam melakukan sterilisasi ruangan.

1.4. Batasan Masalah

Supaya penelitian ini dapat mengarah pada tujuan dan menghindari terlalu banyaknya permasalahan yang muncul, maka penulis memberikan batasan batasan masalah yang sesuai dengan judul penelitian ini. Adapun Batasan Masalah tersebut yaitu:

1. Perancangan mobile robot ini menggunakan ESP32 sebagai kontrol utamanya.
2. Menggunakan 2 buah Lampu UVC 8W.
3. Pengujian intensitas cahaya UVC menggunakan sensor ML8511/GUVA sebagai pendekatan uji sterilisasi ruangan.
4. Tidak memperhitungkan material baik kekuatan bahan maupun sifat dari bahan yang digunakan.
5. Tidak memperhitungkan ukuran dari rancangan mobile robot disinfektan menggunakan UVC berbasis IOT.

1.5. Manfaat Penelitian

Manfaat yang diperoleh dari penelitian ini adalah sebagai berikut:

1. Diharapkan Dapat Dijadikan Sebagai Pedoman Pembuatan Alat Sterilisasi Ruang Publik Yang Lebih Modern Dengan System IOT.
2. Memberikan solusi terhadap masalah sterilisasi ruangan saat ini.

BAB II

TINJAUAN PUSTAKA

2.1. Mobile Robot

Mobile Robot adalah kombinasi dari berbagai komponen fisik (perangkat keras) dan komputasi (perangkat lunak). Dalam hal komponen perangkat keras, mobile robot dapat dianggap sebagai suatu kumpulan subsistem seperti:

- a. *Locomotion* : Bagaimana robot bergerak melalui lingkungannya
- b. *Sensing* : bagaimana robot mengukur sifat dirinya dan lingkungannya
- c. *Reasoning* : Bagaimana Robot memetakan pengukuran ini menjadi tindakan
- d. *Communication* : Bagaimana robot berkomunikasi dengan perangkat lain

Secara umum mobile robot dapat diartikan sebagai robot yang memiliki ruang kerja yang luas. Dikatakan luas, karena bagian dasar dari robot ini diletakkan alat untuk bergerak, seperti roda atau kaki. Selain itu mobile robot dapat dibedakan juga berdasarkan kendalinya, berikut beberapa sistem pengendalian mobile robot diantaranya :

- a. Autonomous Robot

Robot Autonomous adalah robot yang dapat melakukan tugas-tugas yang diinginkan dalam lingkungan yang tidak terstruktur tanpa bimbingan manusia terus menerus berdasarkan logika-logika yang diberikan manusia kepada robot. Salah satu contoh autonomous robot adalah robot Avider yang dapat menghindari penghalang secara otomatis.

- b. Teleoperated Robot

Robot ini dalam pengoperasian dikendalikan dari jarak jauh. Robot jenis ini dikendalikan oleh operator (manusia) dengan menggunakan remote control. Salah satu contoh dari teleoperated robot adalah robot

Radio Control bernama Earth Rider yang bisa dijalankan melewati air, udara sekaligus jalan darat.

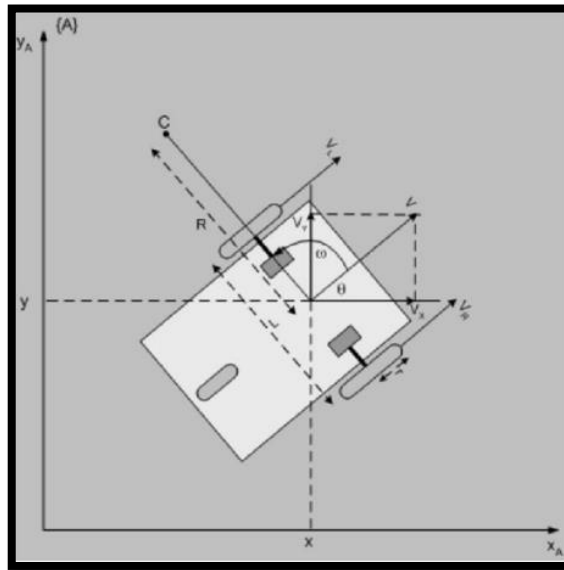
c. Semi Autonomous

Robot semi autonomous adalah robot yang pengendaliannya dapat secara otomatis ataupun pengendalian jarak jauh dengan menggunakan remote control. Salah satu contohnya yaitu robodog buatan Amerika Serikat yang didesain untuk membantu pekerjaan tentara.

2.1.1. Kinematika Mobil Robot Penggerak Diferensial

Salah satu jenis penggerak robot mobile yang umum digunakan terutama untuk dioperasikan dalam ruangan adalah robot mobil adalah dengan sistem penggerak diferensial (differential drive). Alasan utamanya karena relatif lebih fleksibel dalam melakukan manuver serta kemudahan dalam pengontrolannya.

Secara teknis, robot jenis ini pada dasarnya memiliki dua roda utama yang masing-masing digerakan oleh penggerak tersendiri (umumnya berupa motor DC magnet permanent dengan gear-pereduksi yang berfungsi untuk memperkuat torsi motor), selain itu robot ini dilengkapi juga dengan satu atau dua buah roda castor yang ditempatkan dibagian belakang robot yang berfungsi sebagai penyeimbang. **Gambar 2.1** memperlihatkan arsitektur robot dilihat dari bagian atas: Jika kedua roda penggerak tersebut berputar dengan kecepatan yang sama maka robot tersebut akan bergerak dengan arah yang lurus, sedangkan jika kecepatan salah satu roda lebih lambat maka robot akan bergerak membentuk kurva dengan arah lintasan menuju salah satu roda yang bergerak lebih lambat.



Gambar 2.1 Skema Orientasi dan Posisi Mobile Robot Dalam Sistem Koordinat Cartesian

Untuk panjang jari-jari roda r , serta kecepatan rotasi masing-masing roda kanan dan roda kiri berturut-turut ω_r dan ω_l , maka kecepatan linear roda kanan dan roda kiri dapat dicari dengan menggunakan persamaan berikut:

$$V_r(t) = r\omega_r(t) \quad (2.1)$$

$$V_l(t) = r\omega_l(t) \quad (2.2)$$

Ketika robot melakukan gerak memutar (berotasi) sesaat dengan panjang jari-jari R yang diukur dari pusat rotasi (ICC) dan titik pusat kedua roda seperti pada **Gambar 2.2** (Erni, 2009). Maka kecepatan rotasi disetiap titik robot tersebut akan selalu sama (robot adalah sistem mekanis yang rigid), sehingga persamaan berikut ini berlaku untuk menghitung kecepatan rotasi dari robot tersebut.

$$V_R = \omega(R+d) \quad (2.3)$$

$$V_L = \omega(R-d) \quad (2.4)$$

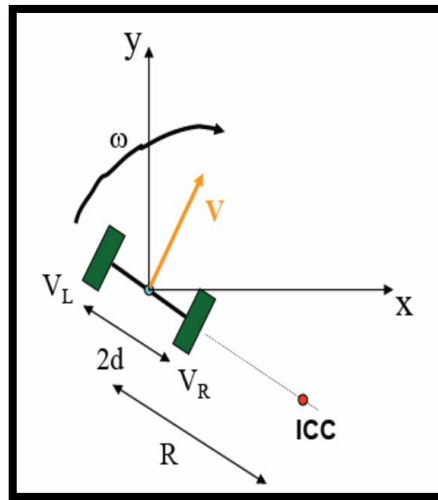
kemudian,

$$\omega = (V_R - V_L) / 2d \quad (2.5)$$

$$R = d(V_R + V_L) / (V_R - V_L) \quad (2.6)$$

Sehingga kecepatan robot :

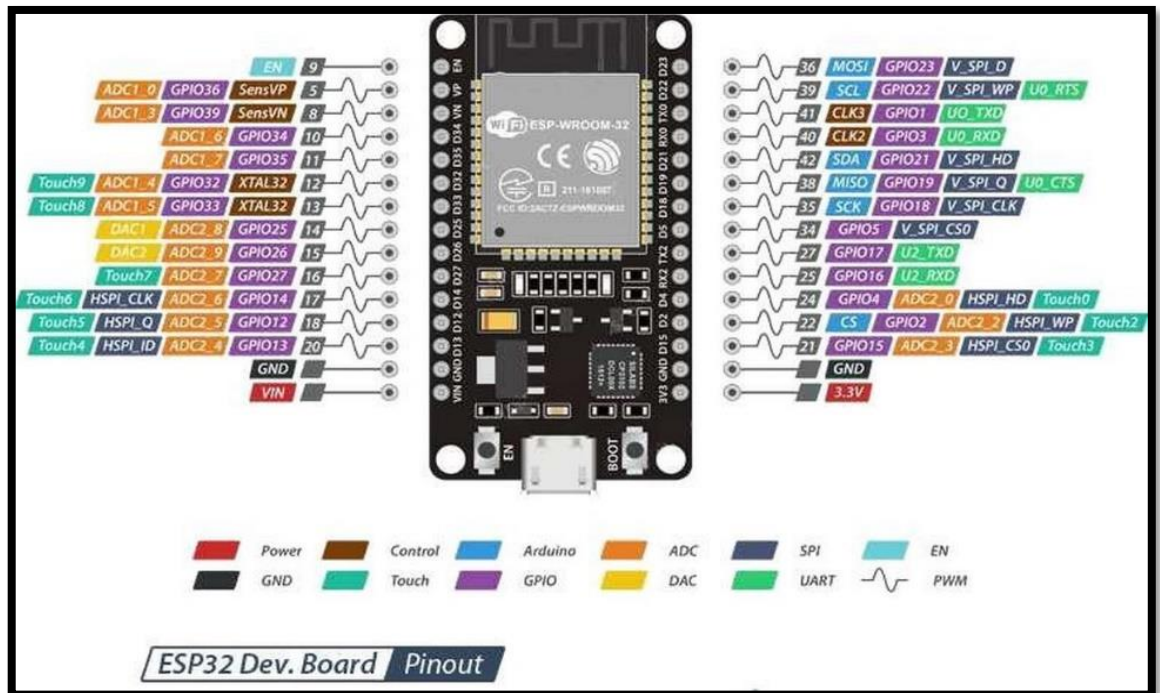
$$V = \omega R = (V_R + V_L)/2 \quad (2.7)$$



Gambar 2.2 Ilustrasi pergerakan robot berdasar titik pusat rotasi

2.1.2. Mikrokontroler ESP32

ESP 32 adalah rangkaian sistem berdaya rendah untuk mikrokontroler chip yang mengintegrasikan *WiFi* dan *Bluetooth* mode ganda. Mikrokontroler ini sudah memiliki modul *WiFi* pada chipnya, yang membuatnya sangat berguna untuk membuat sistem aplikasi *Internet of Things*. Seri ESP 32 mencakup sakelar antena internal, balun RF, penguat daya, penguat penerima kebisingan rendah, filter, dan modul manajemen daya menggunakan mikroprosesor Tensilica Xtensa LX6 dalam varian inti ganda dan inti tunggal. Diproduksi oleh TSMC yang berbasis di Shanghai, China. Menggunakan proses 40nm. Ini adalah penerus dari mikrokontroler ESP 8266.



Gambar 2.3 Pin-out ESP32

Terdapat 36 pin GPIO (**Gambar 2.3**) yang bisa difungsikan sebagai berikut:

- *Analog to Digital Converter* (ADC): 16 kanal SAR ADC 12 bit. Rentang ADC bisa diatur di dalam program, apakah 0-1 V, 0-1.4 V, 0-2V atau 0-4V.
- *Digital to Analog Converter* (DAC): terdapat DAC 8 bit yang bisa menghasilkan tegangan analog.
- *Pulse Width Modulation* (PWM): 16 kanal PWM yang bisa digunakan untuk mengendalikan LED atau motor.
- *Touch Sensor*: 10 GPIO memiliki kemampuan pengindra kapasitif yang dapat digunakan sebagai 10 tombol buttonpad.
- UAR: 2 pin antarmuka UART. Salah satunya digunakan untuk mendownload program secara serial.
- I2C, SPI, I2S: Terdapat dua antarmuka I2C dan 4 antarmuka SPI untuk mengakses sensor dan perangkat ditambah lagi 2 antarmuka I2S.

2.2. Internet of Things (IoT)

IoT merupakan segala aktifitas yang pelakunya saling berinteraksi dan dilakukan dengan memanfaatkan internet. Dalam penggunaannya *Internet of Things* banyak ditemui dalam berbagai aktifitas, contohnya banyaknya transportasi online, *e-commerce*, pemesanan tiket secara online, *live streaming*, *e-learning* dan lain-lain bahkan sampai alat-alat untuk membantu dibidang tertentu seperti *remote temperature sensor*, *GPS tracking*, dan sebagainya yang menggunakan internet atau jaringan sebagai media untuk melakukannya.

2.2.1. Konsep dan Cara Kerja Internet of Things

Cara kerja dari IoT yaitu setiap benda harus memiliki sebuah alamat Internet Protocol (IP). Alamat IP adalah sebuah identitas dalam jaringan yang membuat benda tersebut bisa diperintahkan dari benda lain dalam jaringan yang sama. Alamat IP dalam benda-benda tersebut akan dikoneksikan ke jaringan internet.

Saat ini koneksi internet sudah sangat mudah didapatkan. Dengan demikian pengguna dapat memantau benda bahkan memberi perintah kepada benda lain dengan koneksi internet. Setelah modul *wireless* yang dilengkapi sistem IoT memiliki alamat IP dan terkoneksi dengan internet, pada modul tersebut juga dipasang sebuah sensor. Sensor pada module memungkinkan modul tersebut memperoleh informasi yang dibutuhkan. Setelah memperoleh informasi, modul tersebut dapat mengolah informasi itu sendiri, bahkan berkomunikasi dengan benda-benda lain yang memiliki alamat IP dan terkoneksi dengan internet juga.

2.2.2. Rekomendasi Perangkat Internet of Things

Untuk membuat sebuah sistem IoT, diperlukan sebuah perantara untuk menghubungkan pengguna dengan mikrokontroller. Perantara yang digunakan adalah API atau *Application Programming Interface*.

API dapat memudahkan *programmer* dalam mengendalikan mikrokontroller IoT yang dimiliki. Beberapa API yang bisadigunakan yaitu: Agnosthings, Evothings, Telegram, Blynk, dan lain sebagainya. Dari sekian banyak API, yang paling mudah untuk para pemula adalah Telegram.

2.2.3. VPS (Virtual Private Server)

Untuk membangun sistem IoT dibutuhkan server untuk menjalankan API, salah satu jenis server yang digunakan adalah VPS. VPS (Virtual Private Server) adalah teknologi virtualisasi server. Server fisik dibagi menjadi beberapa server pribadi virtual, dengan masing-masing VPS bertindak seperti server mandiri yang sebenarnya. Setiap VPS memiliki pengaturan sendiri seperti akses root penuh, sistem operasi, dan skrip inisialisasi termasuk sumber daya server seperti CPU dan RAM yang berdiri sendiri, pengguna, pemrosesan, dan sistem file. Tidak seperti shared hosting, yang menggunakan sumber daya server untuk saling memengaruhi, proses yang berjalan di VPS tidak memengaruhi VPS lain di satu server. (Ricky et al., 2010)

VPS memungkinkan Anda untuk menjalankan beberapa sistem operasi secara bersamaan pada satu sistem server fisik. Ini dapat dilakukan tanpa mempartisi ulang partisi dan me-reboot. VPS yang ditentukan menjalankan sistem operasi sesuai kebutuhan. Dengan cara ini, pengguna dapat mem-boot sistem operasi (misalnya, Linux) dan menjalankan sistem operasi lain. Sistem operasi yang berjalan pada sistem operasi host disebut sistem operasi tamu.. (Eckart, Chris. 2008)

2.3. UVC (Ultraviolet C)

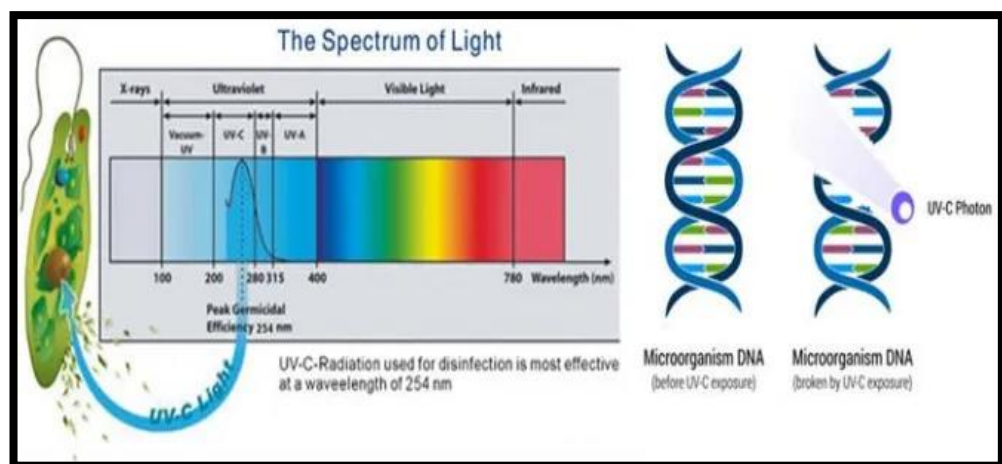
Radiasi ultraviolet telah dipelajari sebagai kemungkinan pembersih patogen lingkungan, termasuk desinfeksi permukaan, instrumen, dan udara. Sinar ultraviolet menghancurkan ikatan molekul dalam DNA pada panjang gelombang tertentu untuk memainkan efek bakterisida. Panjang gelombang spesifik sinar UVC adalah antara 200 dan 270 nanometer (biasanya 254 nanometer), dan panjang gelombang ini berada di bagian kuman dari spektrum elektromagnetik (200 hingga 320 nanometer). Studi tentang efek radiasi ultraviolet harus mempertimbangkan interaksi antara beberapa parameter yang berbeda, secara tidak langsung, waktu paparan, posisi lampu relatif terhadap permukaan yang diterangi, dan hambatan antara sumber cahaya dan permukaan target. Kekuatan cahaya yang dipancarkan, kecepatan dan aliran pergerakan udara. Semua ini mempengaruhi efek keseluruhan radiasi UVC pada permukaan.



Gambar 2.5 Lampu UVC

Sumber cahaya ultraviolet buatan umumnya berasal dari lampu fluorescent khusus, seperti lampu merkuri bertekanan rendah dan lampu merkuri bertekanan sedang. Lampu merkuri bertekanan sedang dapat menghasilkan keluaran radiasi ultraviolet yang lebih tinggi daripada lampu merkuri bertekanan rendah. Namun, lampu merkuri bertekanan rendah mengkonsumsi

listrik lebih efisien daripada lampu merkuri bertekanan sedang. Panjang gelombang maksimum radiasi yang dihasilkan oleh lampu merkuri bertekanan rendah adalah 253,7 nm, yang berakibat fatal bagi mikroorganisme, protozoa, virus, dan ganggang. Pada saat yang sama, panjang gelombang radiasi yang dipancarkan oleh lampu merkuri tekanan sedang adalah 180 hingga 1370 nm. Sterilisasi UV adalah proses fisik di mana energi elektromagnetik ditransfer dari sumber (lampu) ke bahan seluler tubuh (protein dan asam nukleat). Sinar ultraviolet merusak DNA mikroorganisme dengan membentuk dimer timin. Dimer ini mencegah mikroorganisme dari menyalin dan mereplikasi DNA yang pada akhirnya menyebabkan kematian sel. Mekanisme kerusakan DNA UV berdasarkan Alcamo (1984) ditunjukkan pada gambar berikut.



Gambar 2.5 Spektrum lampu UVC dan cara kerja UVC terhadap DNA

Penggunaan dosis radiasi ultraviolet telah terbukti efektif melawan bakteri dan virus di air tanah tanpa menyebabkan pembentukan produk samping yang beracun (Metcalf, Eddy. 2003). Tingkat inaktivasi mikroba tergantung pada dosis UV yang digunakan. Dosis UV adalah produk dari intensitas UV dan waktu paparan, seperti yang ditunjukkan pada **Persamaan 2.8** (Metcalf, Eddy. 2003)

$$D = I \cdot t \quad (2.8)$$

dimana :

D = Dosis UV ($\text{mW} \cdot \text{s}/\text{cm}^2 = \text{mJ}/\text{cm}^2 = \text{J}/\text{m}^2$)

I = Intensitas ultraviolet ($\text{mW}/\text{cm}^2 = \text{W}/\text{m}^2$)

t = Waktu paparan (sekon)

Berikut beberapa data tentang dosis yang dapat menonaktifkan virus SARS Coronavirus berdasarkan data dari beberapa sumber

Tabel 2.1 Efektivitas Dosis Ultraviolet terhadap Coronavirus

| Jenis Virus | Dosis (J/m^2) | Efektivitas | Sumber |
|--|---------------------------------|-------------|-----------------|
| SARS-CoV-2 (Italy-INMI1) | 12.3 | 90% | Bianco 2020 |
| SARS Coronavirus (Frankfurt 1) | 16.4 | 90% | Eickmann 2020 |
| SARS Coronavirus (CoV-P9) | 40.0 | 90% | Duan 2003 |
| SARS-CoV-2 (SARS-CoV-2/Hu/DP/Kng/19-027) | 41.7 | 90% | Inagaki 2020 |
| SARS Coronavirus (Hanoi) | 133.9 | 90% | Kariwa 2004 |
| SARS-CoV-2 | 237.1 | 99.99% | Anna et al 2021 |

Dari **Table 2.1** dapat diketahui bahwa dosis $237.1 \text{ J}/\text{m}^2$ dapat menonaktifkan coronavirus tipe SARS-CoV-2 dengan efektivitas sebesar 99.99%. Dengan demikian berdasarkan data-data dosis UV terhadap efektivitas menonaktifkan virus corona menjadi topik penelitian dengan judul Rancang Bangun Mobile Robot Disinfektan Menggunakan UVC (Ultraviolet C) Berbasis IoT (Internet of Things).

BAB III

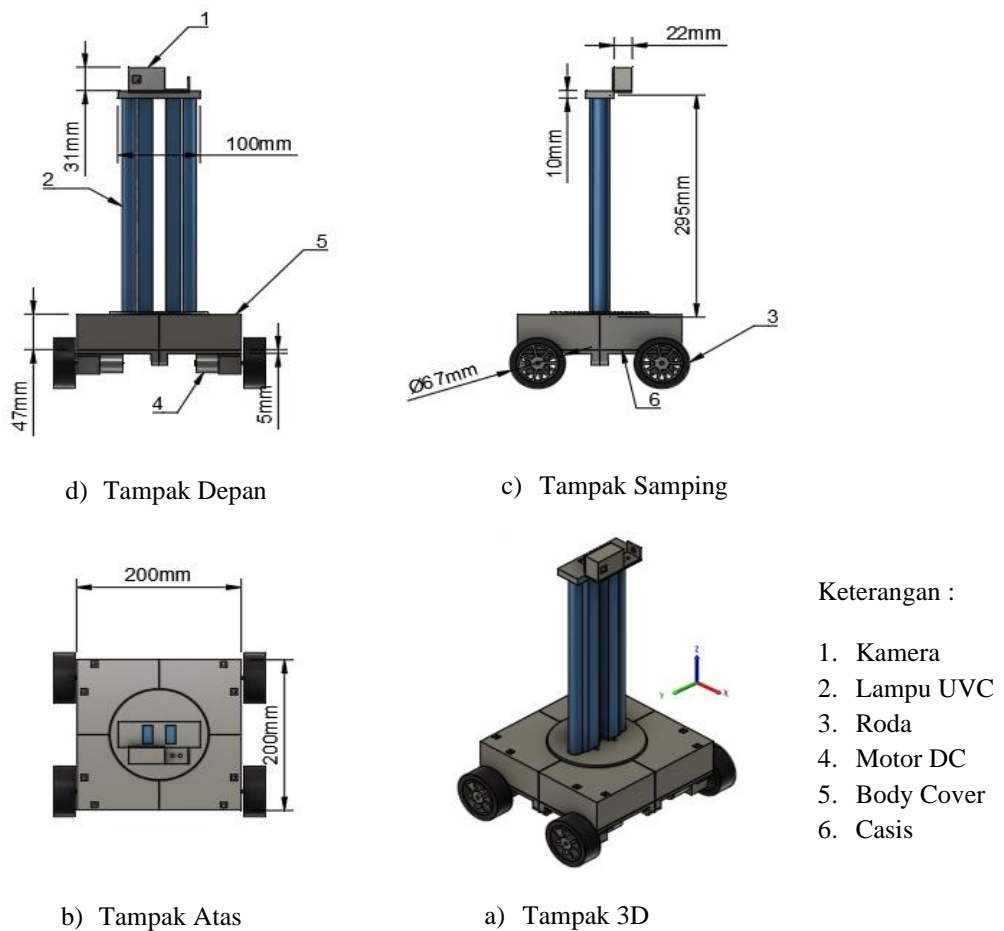
METODOLOGI PENELITIAN

3.1. Waktu dan Tempat Penelitian

Waktu penelitian direncanakan selama 6 bulan dimulai dari bulan Juli 2021 sampai dengan Desember 2021 bertempat di Laboratorium Robot Fakultas Teknik Universitas Hasanuddin, Gowa, Sulawesi Selatan.

3.2. Desain Mobile Robot

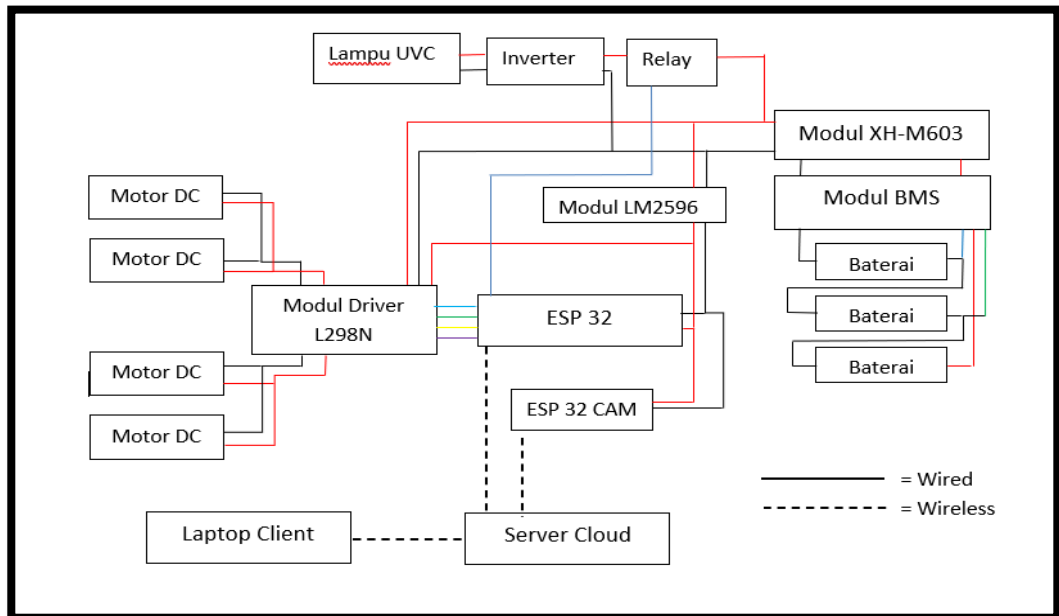
Desain Mobile robot memiliki empat buah motor DC sebagai penggerak robot, 2 Lampu UVC sebagai alat disinfektan. Posisi kedua lampu saling bertolak belakang, hal ini bertujuan agar jangkauan penyinaran lebih efektif. Skema desain mobile robot dapat dilihat pada **Gambar 3.1**.



Gambar 3.1 Skema Desain Mobile Robot Disinfektan UVC

3.3. Desain Perangkat Elektronik

Perangkat Elektronik pada Mobile robotic Disinfektan UVC yang akan dibuat terdiri dari ESP 32, ESP 32CAM, Modul BMS, Driver L298N, Modul XH-M603, Modul LM2596, Inverter dan Relay. Skema rangkaian ditunjukkan pada **Gambar 3.2**.



Gambar 3.2 Skema Rangkaian Elektronik Mobile Robotic

Sistem perangkat elektronik ini merupakan bagian yang mengatur sistem kendali dan kelistrikan mobile robot. Server bertindak sebagai penyimpanan data, penerima, pengirim data dan juga sebagai pemrosesan data yang diterima dari robot (ESP-32) maupun dari laptop (client) melalui internet. ESP 32 berfungsi sebagai pusat kontrol pada mobile robot seperti mengendalikan jalannya motor dc melalui modul driver L298N dan mengontrol nyala pada lampu UVC.

3.4. Desain Perangkat Lunak (*Software*)

Pembuatan API akan dilakukan pada server cloud (VPS) berbasis website dengan menggunakan bahasa pemrograman PHP dan database menggunakan MariaDB. Untuk ESP-32 dan ESP 32CAM akan menggunakan Bahasa pemrograman C/C++.

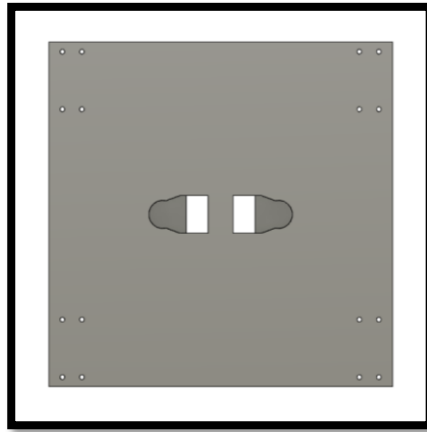
3.5. Alat dan Bahan

Alat dan bahan yang digunakan dalam perancangan Mobile Robot ini adalah

:

1. Sasis Robot

Sasis ini digunakan sebagai dasar penopang berbagai komponen komponen pendukung robot.



Gambar 3.3 Sasis dasar robot

2. Motor DC tipe JGA25-370 100RPM

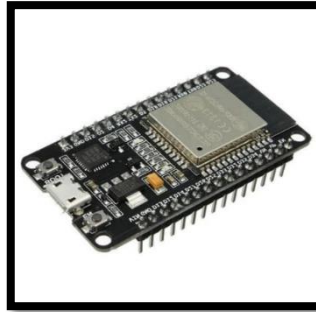
Motor DC berfungsi sebagai penggerak robot.



Gambar 3.4 Motor DC

3. ESP-32

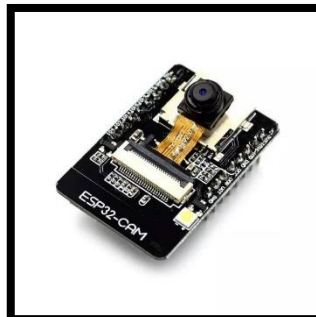
ESP-32 berfungsi sebagai pengontrol komponen komponen elektronik mobile robot.



Gambar 3.5 ESP-32

4. ESP-32CAM

ESP-32CAM berfungsi sebagai kamera yang akan menampilkan lingkungan sekitar robot.



Gambar 3.6 ESP-32CAM

5. Lampu UVC 8W

Lampu UVC berfungsi sebagai alat mensterilkan ruangan



Gambar 3.7 Lampu UVC

6. Baterai Li-ion 18650 3.7V

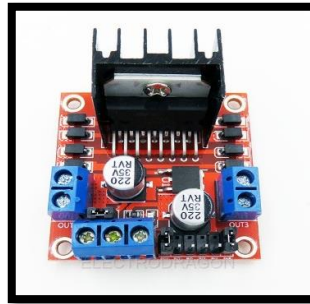
Baterai Li-ion berfungsi sebagai sumber energi listrik mobile robot



Gambar 3.8 Baterai Li-ion 18650

7. Modul H Bridge Driver Motor L298N

Modul Driver Motor L298N berfungsi sebagai pengontrol gerak motor DC robot.



Gambar 3.9 Motor Driver L298N

8. Modul XH-M603

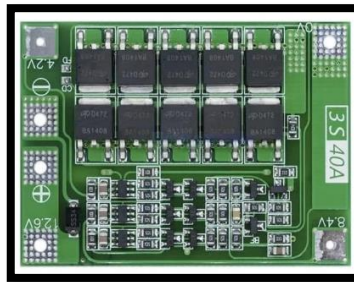
Modul XH-M603 berfungsi sebagai modul kontrol pengisian daya baterai Li-ion



Gambar 3.10 Modul XH-M603

9. Modul BMS 3S 12.6V 40A

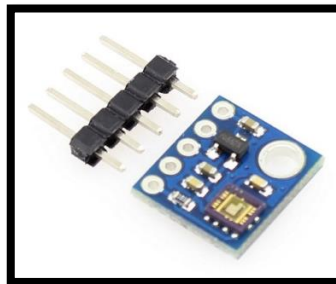
Modul BMS berfungsi sebagai penyeimbang, pemantauan dan proteksi pada baterai Li-Ion 3.7v



Gambar 3.11 Modul BMS

10. Modul sensor ML8511

Modul sensor ML8511 berfungsi sebagai untuk mendeteksi intensitas sinar UVC yang dihasilkan dari UVC mobile robot.



Gambar 3.12 Sensor ML8511

11. Inverter 40W DC-AC 12V to 220V

Inverter berfungsi sebagai mengubah tegangan 12v DC menjadi 220v AC agar Lampu UVC dapat bekerja.



Gambar 3.13 Inverter 40W

12. Modul LM2596

Modul LM2596 Berfungsi sebagai pengubah tegangan dari 12v DC menjadi 5v DC agar modul ESP 32 dan ESP 32 CAM dapat bekerja.



Gambar 3.14 Modul LM2596

13. Modul Relay

Modul relay berfungsi sebagai pemutus atau penyambung aliran listrik pada inverter lampu UVC



Gambar 3.15 Modul Relay

14. Modem

Modem berfungsi sebagai koneksi data mobile robot ke internet.



Gambar 3.16 Modem

3.6. Metode pengambilan data

Setelah dilakukan proses perakitan pada mobile robot dan pemrograman pada website, proses pengambilan data dilakukan secara ekperimental dengan dua sesi.

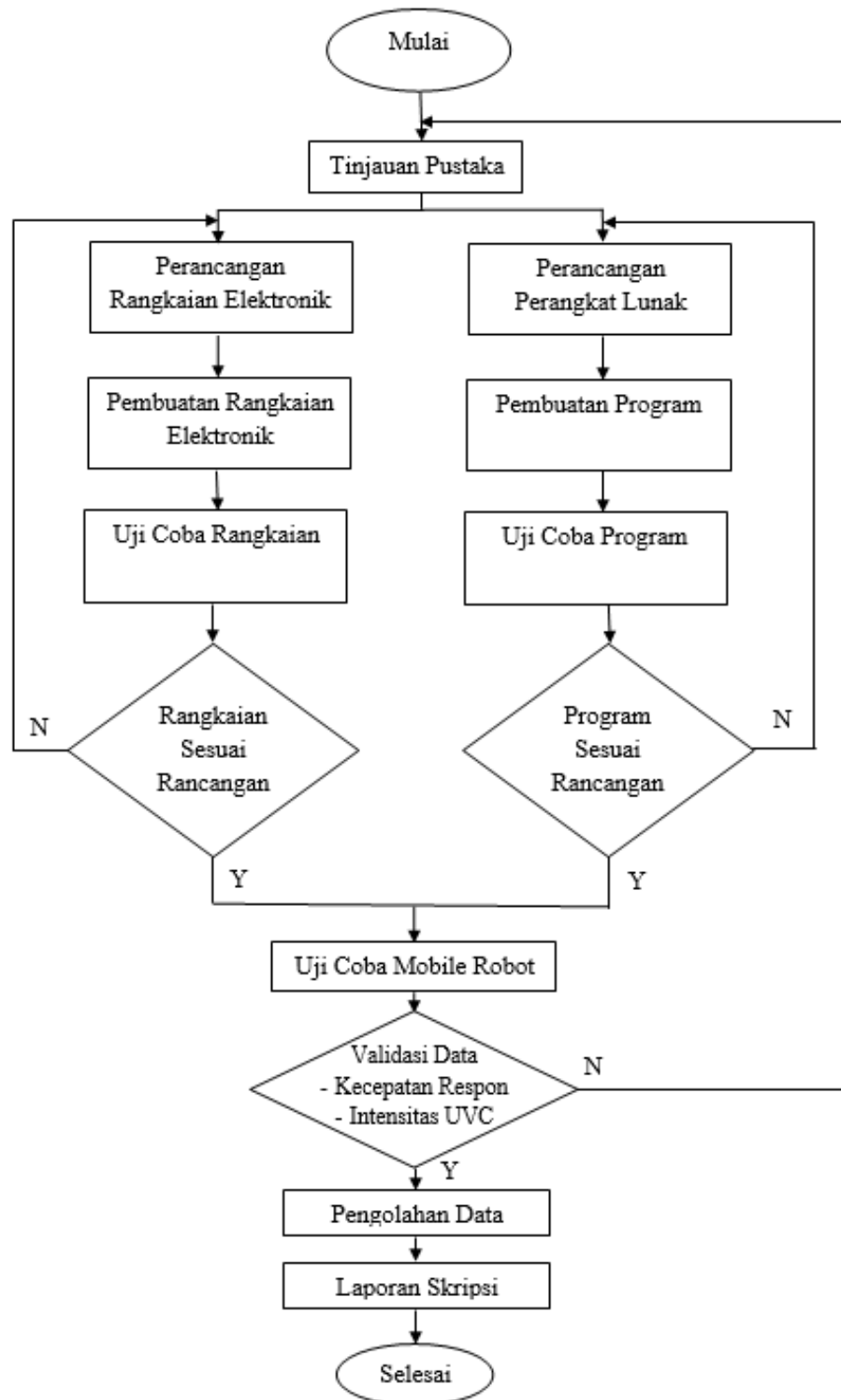
Sesi pertama melakukan proses uji kecepatan akses dari sisi client ke mobile robot melalui internet, Adapun langkah pengambilan datanya sebagai berikut:

Langkah pertama yaitu menyalakan mobile robot, pastikan mobile robot sudah terhubung dengan koneksi, kemudian akses website kontrol dan tekan tombol perintah untuk mengontrol mobile robot. Proses ini akan otomatis mengambil sampel data, ulangi proses diatas dan ganti provider koneksi apabila sampel data respon time sudah melebihi dari 100 data. Setelah data respon time telah didapatkan matikan mobile robot.

Sesi kedua melakukan proses uji intensitas radiasi UVC pada mobile robot, Adapun Langkah pengamblan datanya adalah sebagai berikut:

Langkah pertama nyalakan mobile robot dan lampu UVC, kemudian ukur intensitas menggunakan sensor dengan jarak tiap 0,05m hingga 0,5m pada arah depan, kiri, kanan dan belakang. Setelah data intensitas telah didapatkan matikan mobile robot.

3.7. Flow Chart Penelitian



Gambar 3.17 Diagram Alir Penelitian Mobile Robot

BAB IV

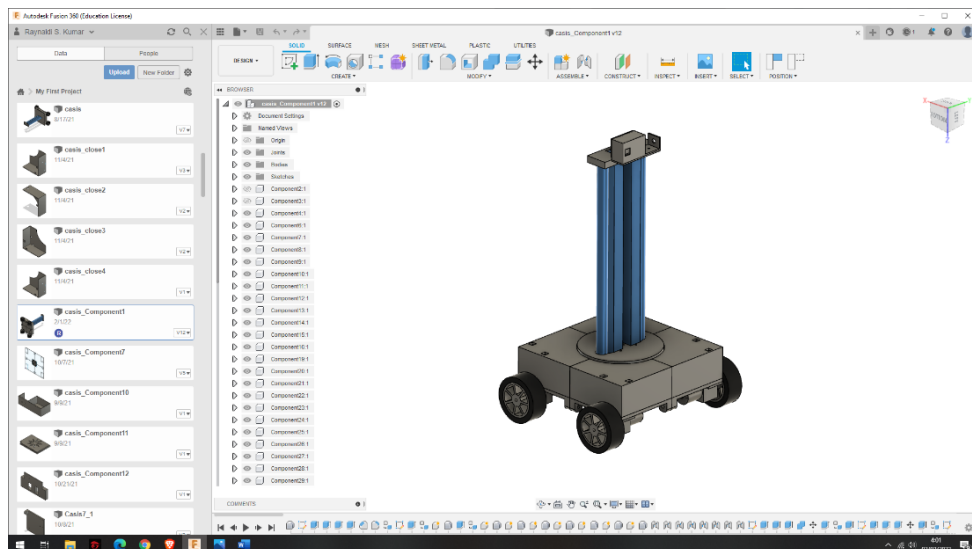
HASIL DAN PEMBAHASAN

4.1. Desain dan Pembuatan *Mobile Robot* Disinfektan *UVC*

Secara umum, ada beberapa langkah yang dilakukan dalam penelitian ini salah satunya adalah mendesain pembuatan *Mobile robot* disinfektan *UVC* yang akan dijelaskan di bawah ini.

4.1.1. Mendesain *Mobile Robot* Disinfektan *UVC*

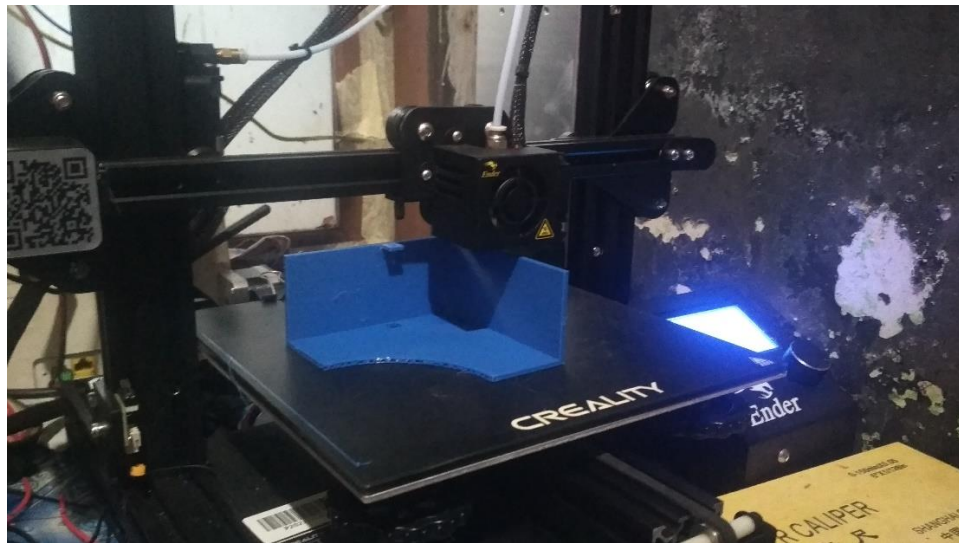
Tahap mendesain *Mobile robot* disinfektan *UVC* merupakan tahapan dimana *Mobile robot* dirancang dalam bentuk tiga dimensi dengan kelengkapan ukuran yang diskalakan dan menggunakan program perangkat lunak Fusino 360. Hal ini mempermudah dalam proses pembuatan kontruksi *Mobile robot* disinfektan *UVC* lihat **Gambar 4.1**



Gambar 4.1 Tampilan perangkat lunak Fusion 360

4.1.2. Pembuatan *Body* Keseluruhan *Mobile Robot*

Pembuatan keseluruhan *body mobile robot* menggunakan teknologi Printer 3D (**Gambar 4.2**) dengan menggunakan material berbahan dasar plastik PLA (*Polylactic acid*) dicetak sesuai dengan desain yang dibuat pada *software Fusion 360*.

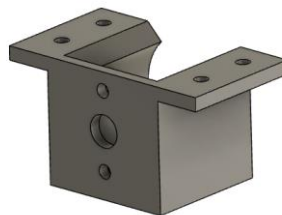


Gambar 4.2 Metode pencetakan salah satu *body mobile robot* menggunakan *Printer 3D*

Pada tahap ini *body mobile robot* dicetak perbagian-bagian kemudian setelah semua selesai tercetak, *body mobile robot* dirakit menggunakan baut dan mur. Berikut bagian-bagian *body mobile robot* dengan fungsinya.

1. *Motor DC housing*

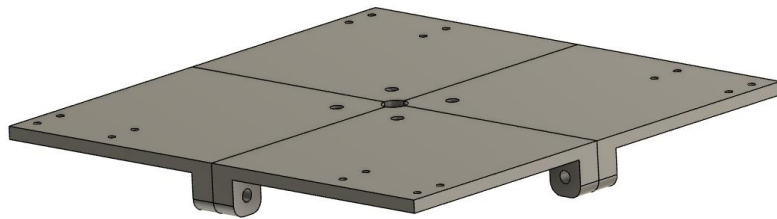
Pada *body* bagian ini merupakan tempat penopang *motor DC* yang akan dihubungkan pada casing utama menggunakan baut dan mur.



Gambar 4.3 *Motor DC housing*

2. *Casis mobile robot*

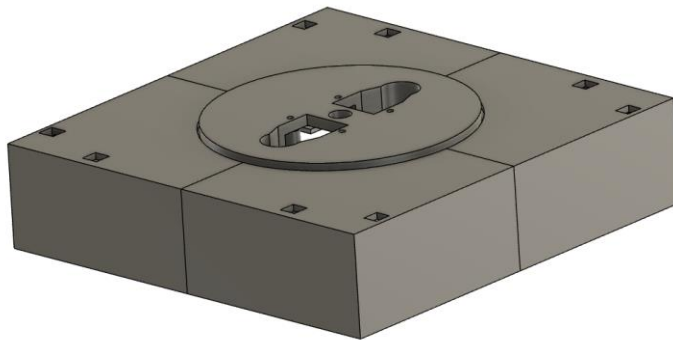
Pada casing ini berguna sebagai penopang utama berbagai komponen seperti komponen elektronik, lampu *UVC*, motor dc, body cover.



Gambar 4.4 Casis mobile robot

3. *Body Cover*

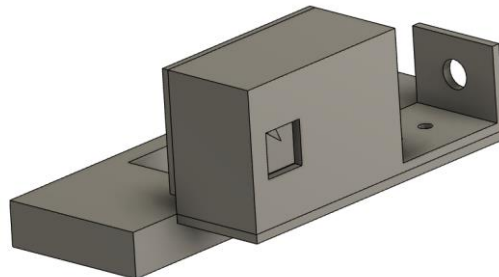
Setelah semua komponen kontrol dan baterai di pasang pada casis *mobile robot*, langkah selanjutnya menutup bagian casis dengan *body cover*. *Body cover* juga berfungsi sebagai penopang lampu *UVC* dan kamera.



Gambar 4.5 *Body cover mobile robot*

4. *Camera Housing*

Camera Housing berfungsi sebagai tempat kamera *esp-32* yang akan diletakkan di atas lampu *UVC*, pada bagian ini didesain khusus dengan tambahan antena agar kamera dapat menampilkan gambar secara optimal melalui *wireless*.



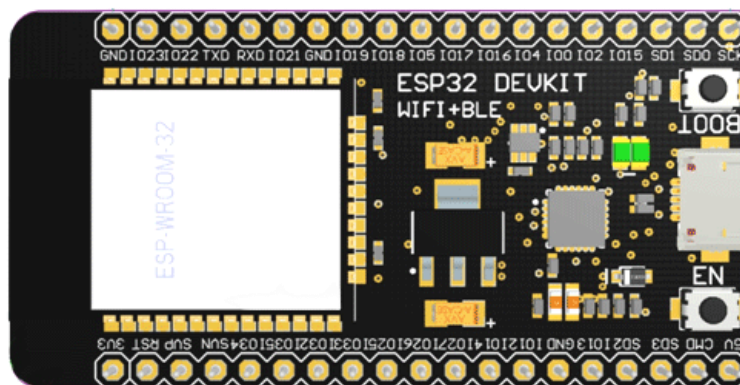
Gambar 4.6 *Camera Housing*

4.1.3. Komponen Elektronik

Mobile robot disinfektan *UVC* menggunakan beberapa komponen elektronik yang memiliki fungsi tersendiri setiap komponennya, salah satu contoh modul L298N yang berfungsi sebagai pengatur arah putaran motor sesuai perintah input. Berikut dijelaskan komponen elektronik yang akan digunakan pada *mobile robot* disinfektan *UVC*.

1. Modul ESP-32 Wroom

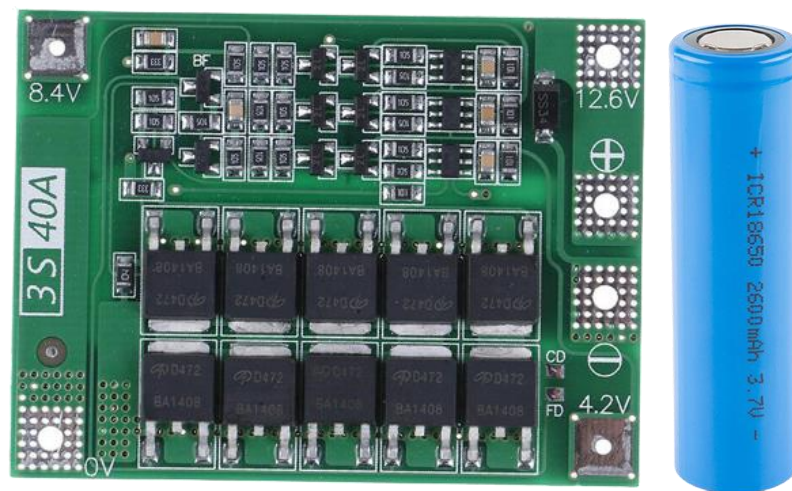
Modul ESP-32 merupakan salah satu mikrocontroller yang mempunyai kelengkapan WiFi dan Bluetooth sebagai komunikasi data melalui gelombang radio, ESP-32 digunakan sebagai pengontrol komponen lainnya yang membutuhkan input perintah, seperti *module drive* L298N dan *modul relay*. Selain itu ESP-32 juga sebagai device komunikasi antara *mobile robot* dengan *server cloud (website)*. Adapun Spesifikasi ESP-32 yaitu, Frekuensi *Clock* sampai dengan 240 MHz, SRAM 520 kB, *flash* memori 4 MB, 11b/g/n *WiFi transceiver*, *bluetooth* 4.2/BLE, 48 pin GPIO, 15 pin *channel ADC* (Analog to Digital Converter), 25 pin PWM (Pulse Width Modulation, 2 pin *channel DAC* (Digital to Analog Converter).



Gambar 4.7 ESP-32 Wroom

2. Modul BMS 3S 40A dan baterai 18650

Mobile robot disinfektan *UVC* menggunakan baterai 18650 3.7V 2600mAh sebagai sumber energi listrik, baterai 18650 dirangkai seri 3 sehingga mempunyai tegangan 11.1v agar dapat memenuhi standar tegangan pada inverter 40w yang akan menyalakan lampu *UVC*, selain itu modul BMS digunakan agar pengisian energi listrik pada baterai seimbang.



Gambar 4.8 Modul BMS (kiri) dan Baterai 18650 (kanan)

3. Kamera ESP-32

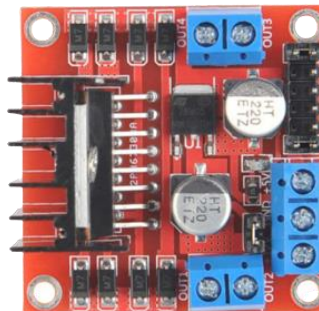
Fungsi dari kamera ini agar dapat melihat kondisi lingkungan sekitar robot, kamera ESP-32 dipilih karena memiliki kelebihan yaitu dapat mengirimkan data melalui WiFi, sehingga kamera ini dapat diakses di manapun melalui internet. Tampilan dari kamera ini akan dikirim ke *server cloud (website)* menggunakan *tunneling ngrok*. Berikut spesifikasi dari kamera ESP-32 yaitu, Low-Power Dual-Core 32bit CPU, Frekuensi sampai dengan 240 MHz, 520KB SRAM, mendukung Interfaces UART/SPI/I2C/PWM/ADC/DAC, koneksi WiFi 802.11b/g/n, OV2640 Camera 2MP, mendukung kartu memori TF.



Gambar 4.9 Kamera ESP-32

4. Modul Motor Driver L298N

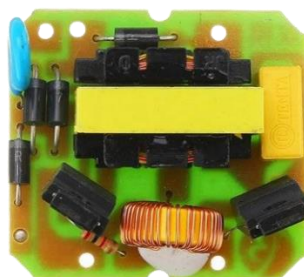
Motor driver L298N digunakan untuk mengontrol arah putaran, kecepatan putar, serta dapat digunakan sebagai rem dinamis putaran dengan masukan input perintah dari mikrokontroler ESP-32.



Gambar 4.10 Modul motor driver L298N

5. Inverter DC-AC 40w

Inverter berfungsi sebagai pengubah tegangan DC menjadi AC 220V, karena sumber tegangan menggunakan baterai inverter dibutuhkan agar dapat menyalakan lampu UVC yang bekerja pada tegangan AC 220V.



Gambar 4.11 Inverter DC-AC 40w

6. Modul Relay

Pada *mobile robot* modul relay ini berfungsi untuk memutus, menghubungkan dan menguatkan aliran listrik pada inverter yang terhubung langsung pada lampu UVC, dengan kata lain modul relay mengendalikan nyala dan matinya lampu UVC dengan perintah input kontrol dari ESP-32.



Gambar 4.12 Modul Relay

7. Lampu UVC 8W

Lampu UVC berguna untuk sterilisasi ruangan, *mobile robot* ini menggunakan 2 buah lampu UVC dengan penempatan saling bertolak belakang, hal ini bertujuan agar penyinaran UVC menjadi optimal. Adapun spesifikasi dari lampu UVC yaitu, wavelength 189nm – 245nm, tegangan kerja 220V, frekuensi 50Hz/60Hz, daya 8W.



Gambar 4.13 Lampu UVC

8. Modul LM2596

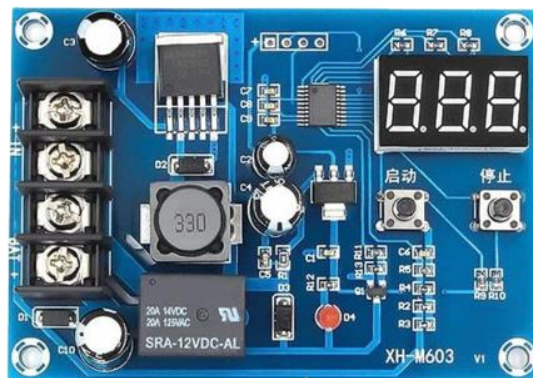
Modul LM2596 berfungsi sebagai pengubah tegangan, pada *mobile robot* tegangan baterai 11.1v diubah menjadi 5v, ini bertujuan agar komponen yang mempunyai tegangan kerja 5v seperti ESP-32, modul relay, kamera ESP-32 dapat bekerja dengan normal.



Gambar 4.14 Modul LM2596

9. Modul XH-M603

Modul XH-M603 ini berfungsi sebagai pemutus daya agar pengisian ulang pada baterai aman. Pada Modul XH-M603 terdapat konfigurasi batas maksimal tegangan yang terisi pada baterai, sehingga pada saat kondisi tegangan mencapai batas maksimal modul ini dapat memutus arus secara langsung.



Gambar 4.15 Modul XH-M603

10. Motor DC JGA25-370 100RPM

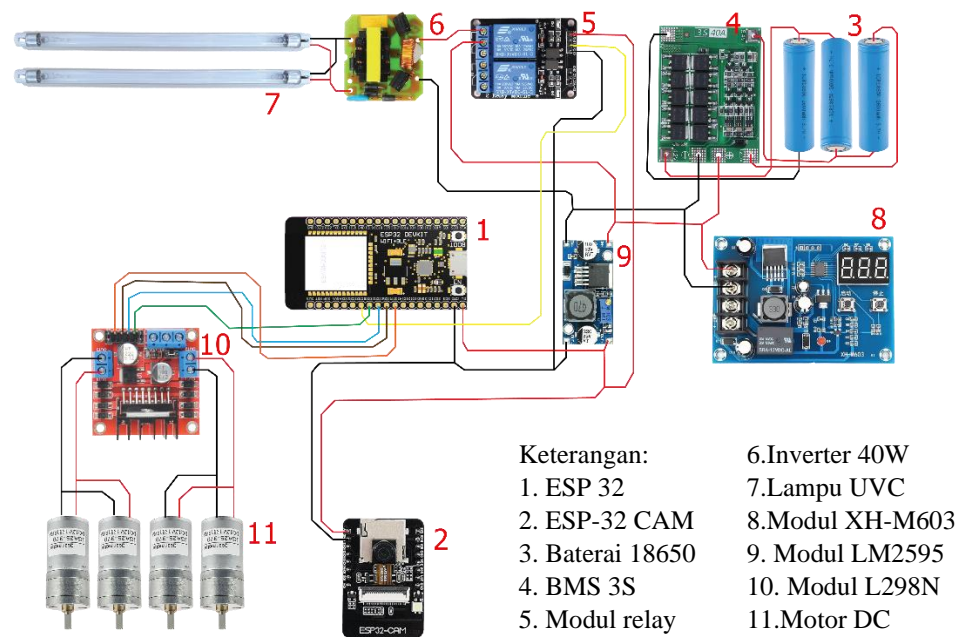
Motor DC berfungsi sebagai penggerak utama mobile robot, mobile robot mempunyai 4 buah motor DC yang terhubung dengan driver L298N. Adapun spesifikasi dari motor DC JGA25-370 100RPM yaitu, tegangan kerja 6-18 volt, berat 84 gram, arus maksimal 1,3 ampere, torsi 2,1 kg.cm.



Gambar 4.16 Motor DC JGA25-370

4.1.4. Skema Rangkaian Elektronik

Setelah semua komponen telah ditentukan tahap selanjutnya adalah menghubungkan atau merangkai semua komponen komponen menjadi satu kesatuan yang dapat berfungsi sesuai dengan yang direncanakan.

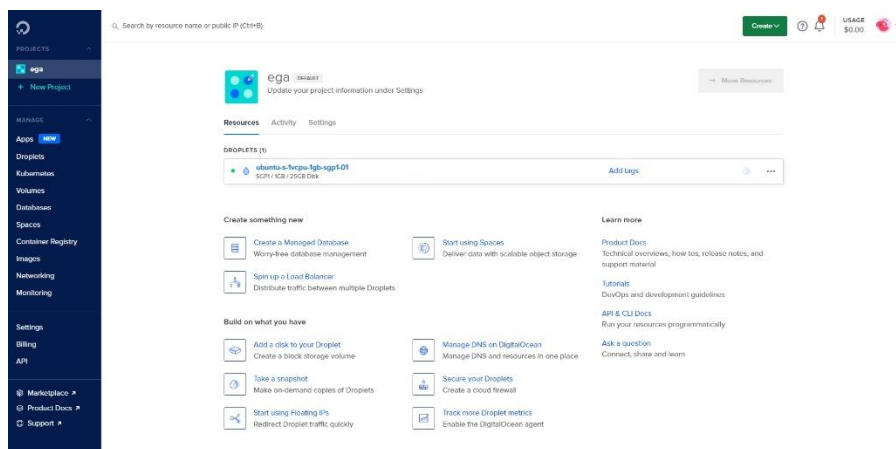


Gambar 4.17 Skema Rangkaian Elektronik *Mobile Robot*

Pada **Gambar 4.17** *mobile robot* disinfektan UVC memiliki 4 buah motor DC yang terhubung pada modul L298N, 2 buah lampu UVC yang terhubung pada inverter, 3 buah baterai yang memiliki total energi listrik 11.1 Volt dengan kapasitas 2600mAh. Penggunaan ESP-32 sebagai kontrol utama yang mengontrol modul driver L298N yang memiliki 4 pin input, selain itu ESP-32 juga mengontrol modul relay agar dapat menghidupkan atau mematikan lampu UVC yang terhubung pada inverter. Selain itu modul LM295 yang terhubung pada baterai 11.1 Volt memberikan output tegangan 5v ke komponen seperti, ESP-32, ESP-32 CAM, motor driver L298N dan relay.

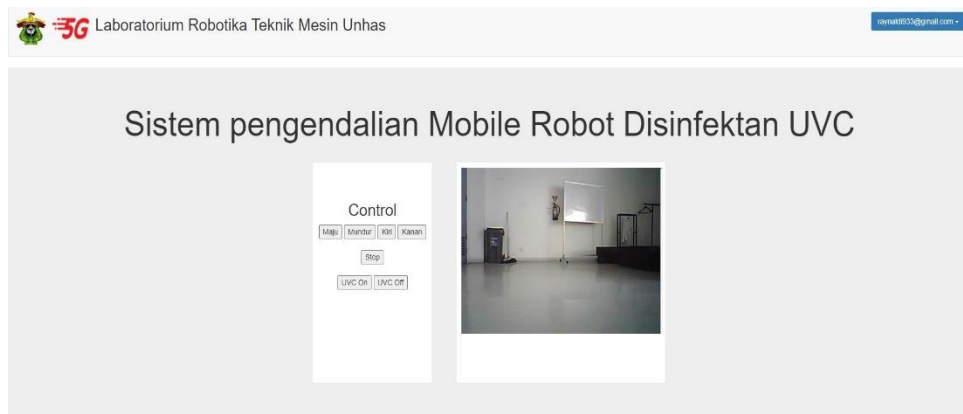
4.1.5. Pembuatan Website Dan Server

Setelah *mobile robot* terakit sesuai pada skema rangkaian, langkah selanjutnya adalah membuat server untuk penyimpanan data website serta database untuk kontrol *mobile robot* via internet. Server yang dipakai adalah server cloud yang dihosting pada Digitalocean. Adapun spesifikasi dari server cloud yaitu, server lokasi singapura, 25gb ssd, 1gb ram,1 vcpu dan OS ubuntu.



Gambar 4.18 Tampilan dashboard Digitalocean

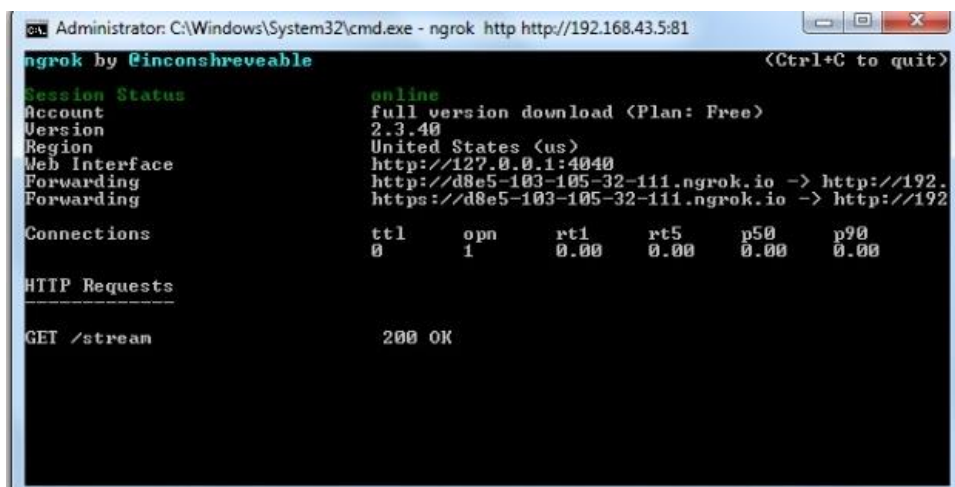
Setelah proses deploy server pada digitalocean selanjutnya membuat aplikasi API secara native, untuk pembuatannya MariaDB digunakan sebagai pengolahan database serta PHP versi 7.2 sebagai aplikasi API. API sendiri berfungsi sebagai penghubung komunikasi antara *mobile robot* dan website, protokol yang dipakai untuk komunikasi antara *mobile robot* dan website adalah HTTP (Hypertext Transfer Protocol).



Gambar 4.19 Tampilan website kontrol *mobile robot*

Pada website terdapat 3 halaman utama yaitu halaman home, login dan halaman kontrol. Untuk halaman kontrol dapat kita lihat pada **Gambar 4.19**, halaman kontrol menunjukkan beberapa tombol kontrol seperti tombol maju, mundur, kiri, kanan, stop, UVC off dan UVC on. Selain itu pada halaman kontrol juga terdapat tampilan monitoring kamera ESP32 yang terpasang pada *mobile robot* UVC.

Untuk kamera ESP-32 diperlukan tunneling agar sambungan lokal IP address pada kamera ESP-32 bisa terhubung lewat internet, salah satu layanan yang dipakai untuk tunneling ini adalah Ngrok (**Gambar 4.20**). Ngrok membuat saluran dari ip lokal dari kamera ESP-32 ke jalur URL khusus yang dapat diakses melalui internet.



Gambar 4.20 Tunneling EPS-32 CAM melalui Ngrok

4.1.6. Pembuatan Kode Program pada ESP-32

Setelah website telah selesai dibuat dan rangkaian elektronik dari *mobile robot* disinfektan UVC telah selesai, tahap selanjutnya yaitu penulisan kode program menggunakan software Arduino IDE (**Gambar 4.21**) dengan menghubungkan url website yang telah dibuat tadi sebagai pusat kontrol dari *mobile robot* dengan metode http get.



```
web | Arduino 1.8.13
File Edit Sketch Tools Help

web $

#include <WiFi.h>
#include <HTTPClient.h>

const char* ssid = "ayeye";
const char* password = "passwordkuini";

//Your Domain name with URL path or IP address with path
String serverName = "http://ssrc.my.id/api/motorstatus.php";

// the following variables are unsigned longs because the time, measured in
// milliseconds, will quickly become a bigger number than can be stored in an int.
unsigned long lastTime = 0;
// Timer set to 10 minutes (600000)
//unsigned long timerDelay = 600000;
// Sec timer to 5 seconds (5000)
unsigned long timerDelay = 100;

int m1 = 33;
int m2 = 25;
int m3 = 26;
int m4 = 27;
int uvc = 32;

void setup() {
  Serial.begin(115200);

  pinMode(m1, OUTPUT);
  pinMode(m2, OUTPUT);
  pinMode(m3, OUTPUT);
  pinMode(m4, OUTPUT);
  pinMode(uvc, OUTPUT);

  WiFi.begin(ssid, password);
  Serial.println("Connecting");
  while(WiFi.status() != WL_CONNECTED) {
    delay(50);
    Serial.print(".");
  }
  Serial.println("");
  Serial.print("Connected to WiFi network with IP Address: ");
  Serial.println(WiFi.localIP());

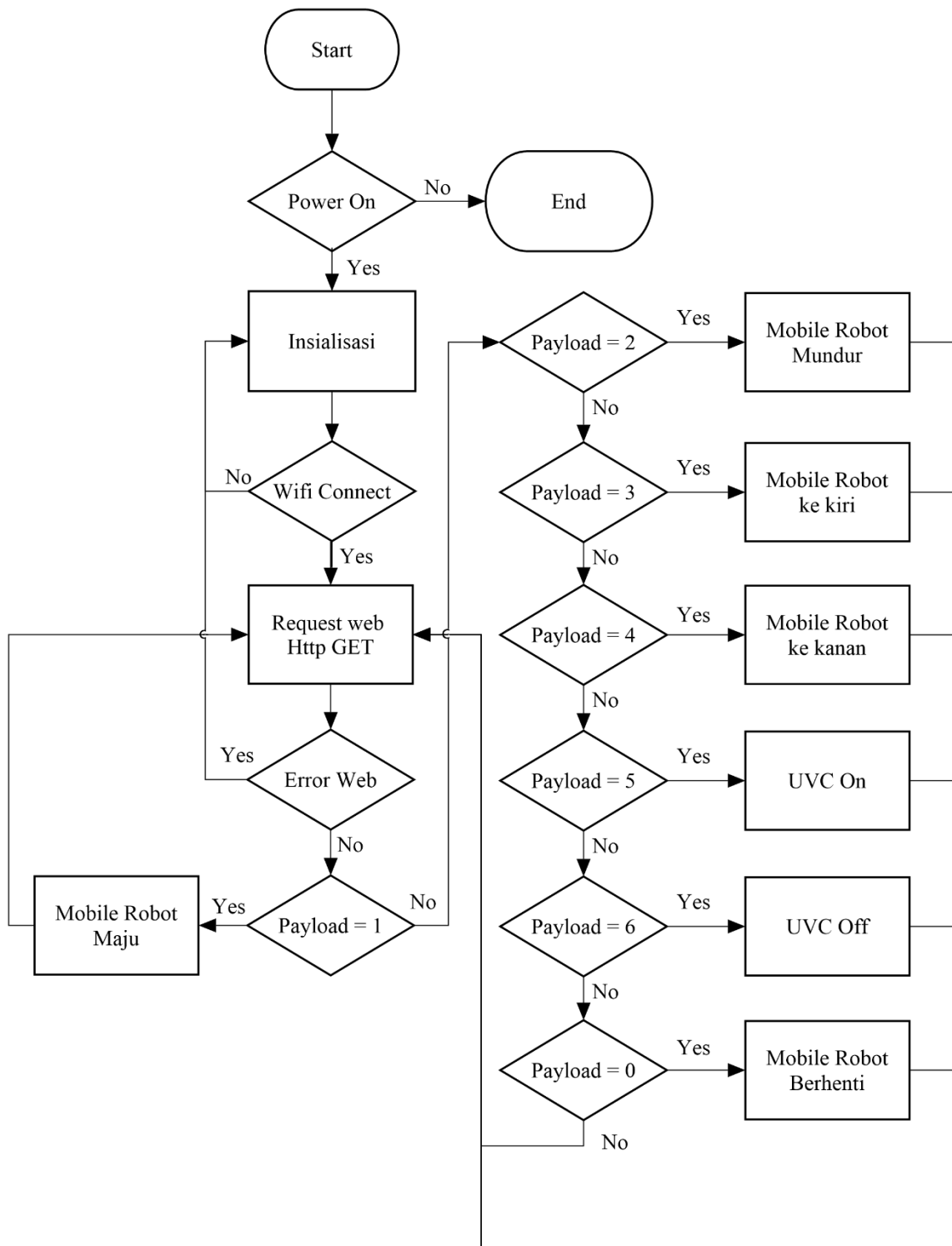
  Serial.println("Timer set to 5 seconds (timerDelay variable), it will take 5 seconds before publishing the first reading.");
}

const char* root_ca = \
```

Gambar 4.21 Tampilan software Arduino IDE

Secara garis besar dasar program ini merupakan basic example code dari httpclient yang kemudian dikembangkan dengan menambahkan perintah data yang diterima untuk menjalankan driver motor L298N dan perintah untuk mengendalikan kondisi relay pada lampu UVC dengan menggunakan logic statement.

Adapun langkah-langkah dari program *mobile robot* bisa dilihat pada flowchart dibawah ini.



Gambar 4.22 Diagram Alir Program *Mobile Robot*

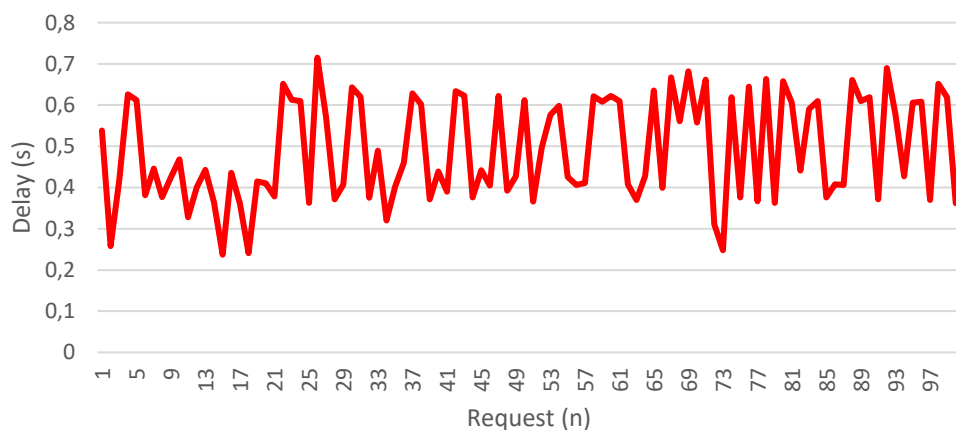
4.2. Uji Coba Mobile Robot Disinfektan UVC

Uji coba dilakukan untuk mendapatkan data berupa delay respon antara *mobile robot* dengan website yang menggunakan beberapa provider sebagai acuan perbandingan dan intensitas lampu UVC dari *mobile robot*.

4.2.1. Respon Time (Delay) Mobile Robot

Delay merupakan waktu yang dibutuhkan untuk mengirimkan data dari sumber ke tujuan. Pengujian respon time (delay) dilakukan untuk mengetahui waktu yang dibutuhkan *mobile robot* untuk request data pada website. Pengujian ini dilakukan sebanyak 100 request dengan metode http get dari sisi *mobile robot* (client) ke sisi website (server) dengan menggunakan jaringan 4G dari berbagai provider. Berikut hasil dari pengujian respon data delay dari beberapa provider.

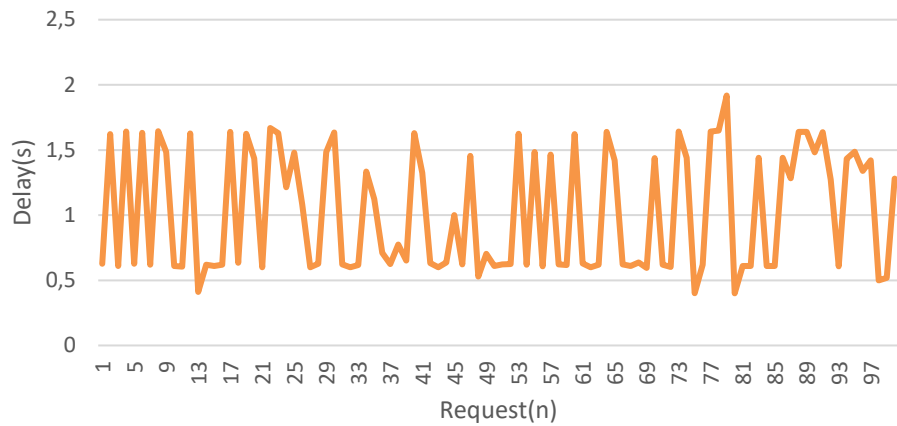
1. Data Delay Provider Telkomsel



Gambar 4.23 Respon time (delay) provider Telkomsel

Pada **Gambar 4.23** menunjukkan respon time (delay) dalam satuan s(second) dari 100 request yang berasal dari *mobile robot* ke website dengan menggunakan provider Telkomsel, untuk nilai delay terendah yaitu 0,237s pada request (n) ke 15, kemudian untuk delay tertinggi terdapat pada request (n) ke 26 dengan nilai 0,715s, kemudian untuk nilai rata-rata dari delay Telkomsel yaitu 0,491s.

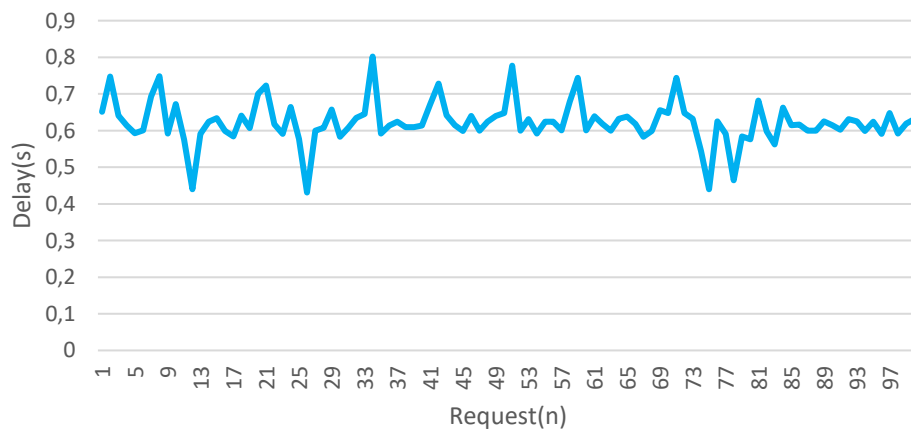
2. Data Delay Provider Tri



Gambar 4.24 Respon time (delay) provider Tri

Pada **Gambar 4.24** menunjukkan respon time (delay) dalam satuan s(second) dari 100 request yang berasal dari *mobile robot* ke website dengan menggunakan provider Tri, untuk nilai delay terendah yaitu 0,410s pada request (n) ke 13, kemudian untuk delay tertinggi terdapat pada request (n) ke 79 dengan nilai 1,918s, kemudian untuk nilai rata-rata dari delay Tri yaitu 1,022s.

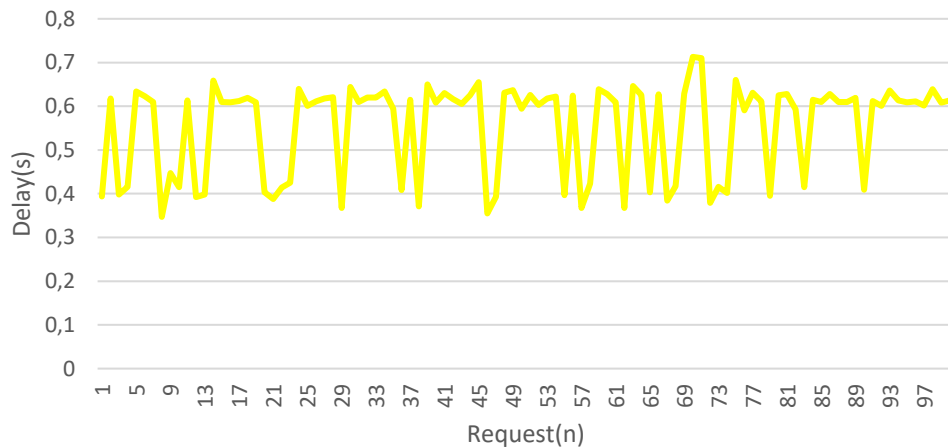
3. Data Delay Provider XL



Gambar 4.25 Respon time (delay) provider XL

Pada **Gambar 4.25** menunjukkan respon time (delay) dalam satuan s(second) dari 100 request yang berasal dari *mobile robot* ke website dengan menggunakan provider XL, untuk nilai delay terendah yaitu 0,431s pada request (n) ke 26, kemudian untuk delay tertinggi terdapat pada request (n) ke 34 dengan nilai 0,802s, kemudian untuk nilai rata-rata dari delay XL yaitu 0,622s.

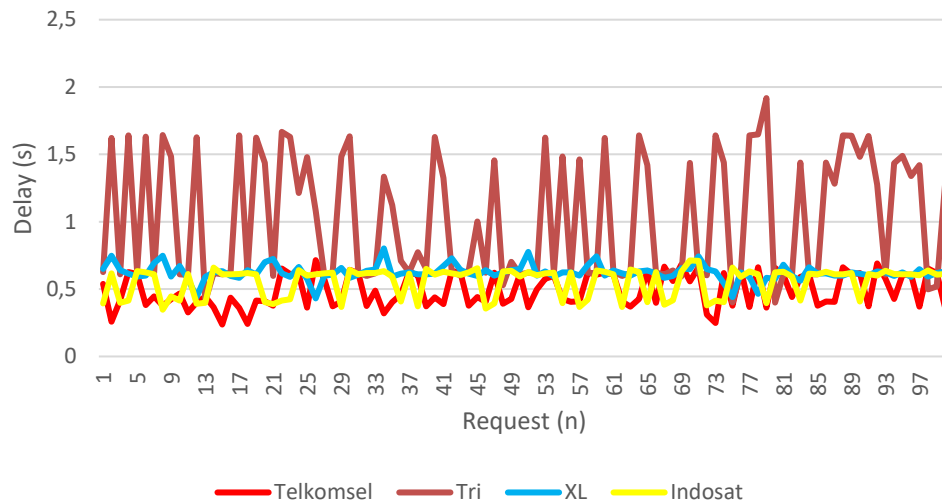
4. Data Delay Provider Indosat



Gambar 4.26 Respon time (delay) provider Indosat

Pada **Gambar 4.26** menunjukkan respon time (delay) dalam satuan s(second) dari 100 request yang berasal dari *mobile robot* ke website dengan menggunakan provider Indosat, untuk nilai delay terendah yaitu 0,347s pada request (n) ke 8, kemudian untuk delay tertinggi terdapat pada request (n) ke 70 dengan nilai 0,713s, kemudian untuk nilai rata-rata dari delay Indosat yaitu 0,554s.

5. Perbandingan Delay Tiap Provider



Gambar 4.27 Respon time (delay) tiap provider

Pada **Gambar 4.27** menunjukkan data respon time (delay) dari berbagai provider, untuk delay tertinggi dimiliki oleh provider tri dengan rata-rata delay 1,022s sedangkan untuk respon delay terendah dimiliki oleh provider telkomsel dengan rata-rata delay 0,491s.

Dari data diatas faktor yang mempengaruhi delay dari tiap provider seperti, kualitas layanan internet, kondisi cuaca, waktu pengujian dan letak geografis. (hasyim, 2016)

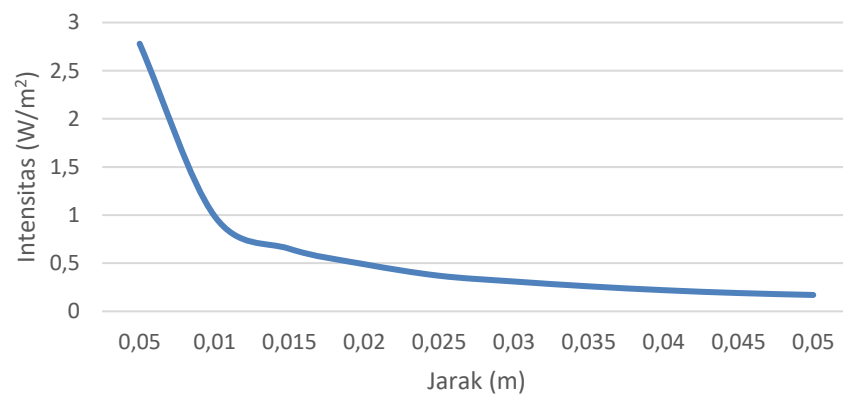
Delay antara request web dengan *mobile robot* sangat mempengaruhi jalannya *mobile robot* sehingga pengguna sangat sulit menjalankan perintah apabila delay yang dihasilkan tinggi, seperti contoh apabila ingin menekan tombol stop saat mobil robot sedang berjalan maju, apabila delay terlalu tinggi maka robot akan terus berjalan sehingga dapat menabrak. (yahya, 2019)

4.2.2. Intensitas Lampu UVC *Mobile Robot*

Intensitas merupakan daya yang ditransfer persatuan luas, adapun satuan yang dipakai dari intensitas lampu UVC yaitu W/m^2 . Pada

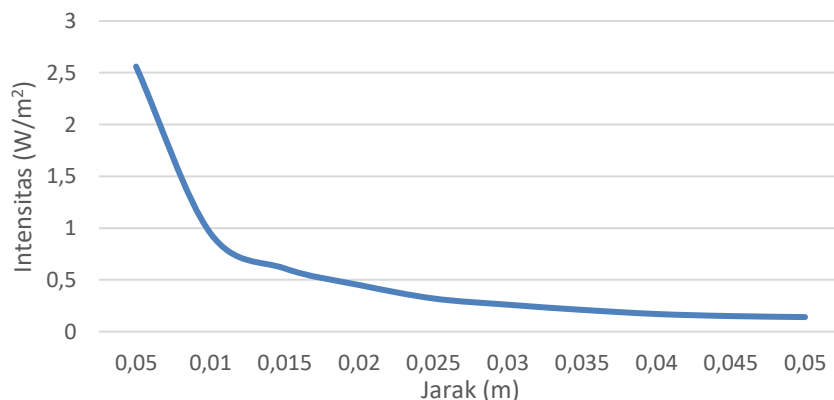
pengujian *mobile robot*, intensitas UVC diukur dengan variasi jarak 0,5m hingga 0,05m dan arah pancaran dari lampu UVC sendiri. Berikut hasil dari pengujian intensitas dari *mobile robot* disinfektan UVC.

1. Intensitas UVC pada arah depan *mobile robot*



Gambar 4.28 Jarak terhadap intensitas lampu UVC pada arah depan
Pada **Gambar 4.28** menunjukkan besaran intensitas lampu UVC pada arah depan yang dipengaruhi oleh jarak, adapun nilai intensitas UVC tertinggi terdapat pada jarak 0,05m dari arah depan *mobile robot* yaitu sebesar 2,78W/m² sedangkan pada jarak 0,5m intensitas UVC menjadi 0,17W/m².

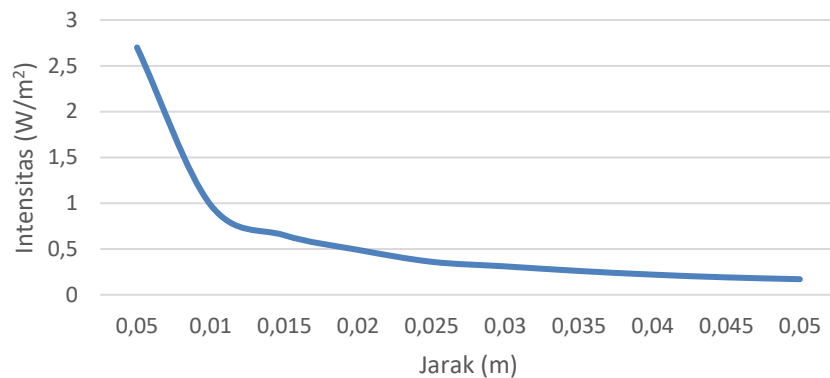
2. Intensitas UVC pada arah kiri *mobile robot*



Gambar 4.29 Jarak terhadap intensitas lampu UVC pada arah kiri

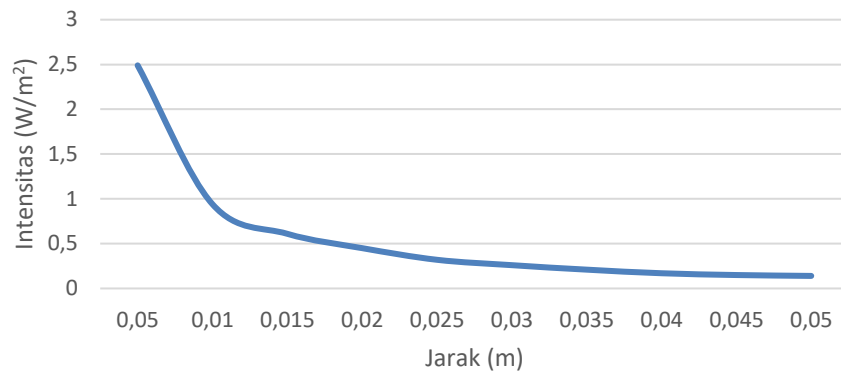
Pada **Gambar 4.29** menunjukkan besaran intensitas lampu UVC pada arah kiri yang dipengaruhi oleh jarak, adapun nilai intensitas UVC tertinggi terdapat pada jarak 0,05m dari arah kiri *mobile robot* yaitu sebesar 2,56W/m² sedangkan pada jarak 0,5m intensitas UVC menjadi 0,14W/m².

3. Intensitas UVC pada arah belakang mobile robot



Gambar 4.30 Jarak terhadap intensitas lampu UVC pada arah belakang Pada **Gambar 4.30** menunjukkan besaran intensitas lampu UVC pada arah belakang yang dipengaruhi oleh jarak, adapun nilai intensitas UVC tertinggi terdapat pada jarak 0,05m dari arah belakang *mobile robot* yaitu sebesar 2,7W/m² sedangkan pada jarak 0,5m intensitas UVC menjadi 0,17W/m².

4. Intensitas UVC pada arah kanan mobile robot

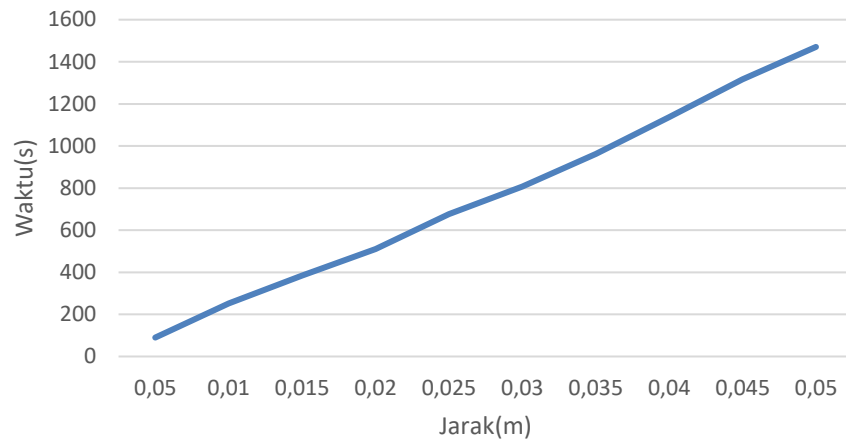


Gambar 4.31 Jarak terhadap intensitas lampu UVC pada arah kanan
Pada **Gambar 4.31** menunjukkan besaran intensitas lampu UVC pada arah kanan yang dipengaruhi oleh jarak, adapun nilai intensitas UVC tertinggi terdapat pada jarak 0,05m dari arah kanan *mobile robot* yaitu sebesar 2,49 W/m^2 sedangkan pada jarak 0,5m intensitas UVC menjadi 0,14 W/m^2 .

4.2.3. Perbandingan Jarak Dengan Durasi waktu Penyinaran Pada Dosis 250J/m²

Berdasarkan data intensitas diatas, maka dapat diperoleh durasi waktu penyinaran oleh lampu UVC dari dosis yang ditentukan dibagi dengan data intensitas. Dosis 250J/m² didapat dari referensi yang terbukti dapat menonaktifkan virus Covid19 pada permukaan. Berikut merupakan perbandingan jarak dengan durasi waktu penyinaran oleh UVC pada dosis 250J/m².

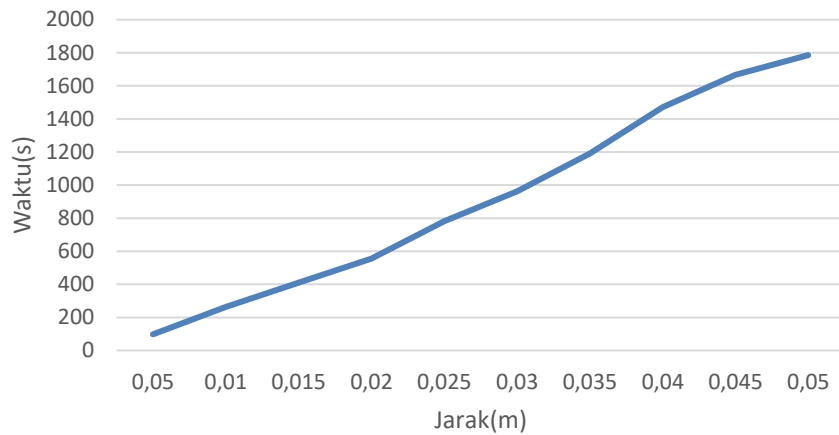
1. Durasi waktu penyinaran UVC pada arah depan



Gambar 4.32 Jarak terhadap durasi waktu penyinaran lampu UVC pada arah depan

Pada **Gambar 4.32** menunjukkan durasi waktu penyinaran lampu UVC pada arah depan *mobile robot* yang dipengaruhi oleh jarak, adapun durasi waktu penyinaran terlama terdapat pada jarak 0,5m dengan waktu 1470 detik dan untuk waktu penyinaran tercepat terdapat pada jarak 0,05m dengan waktu 89 detik. Arah penyinaran pada posisi depan mendapatkan nilai terbaik dari arah yang lainnya, hal ini disebabkan posisi dari lampu UVC.

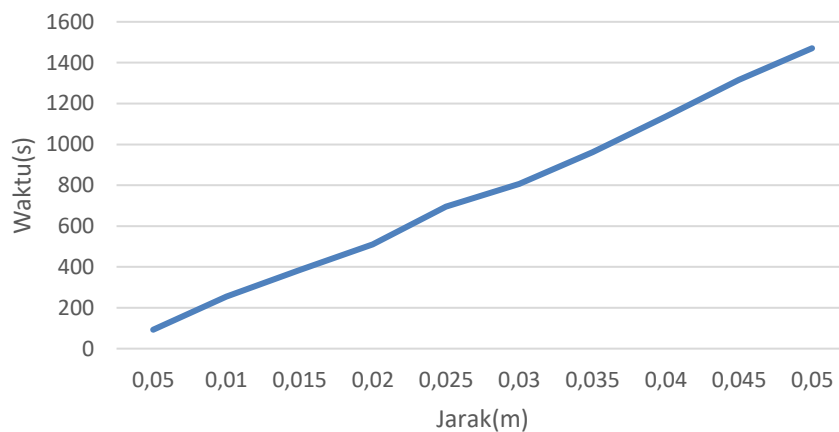
2. Durasi waktu penyinaran UVC pada arah kiri



Gambar 4.33 Jarak terhadap durasi waktu penyinaran lampu UVC pada arah kiri

Pada **Gambar 4.33** menunjukkan durasi waktu penyinaran lampu UVC pada arah kiri *mobile robot* yang dipengaruhi oleh jarak, adapun durasi waktu penyinaran terlama terdapat pada jarak 0,5m dengan waktu 1785 detik dan untuk waktu penyinaran tercepat terdapat pada jarak 0,05m dengan waktu 97 detik.

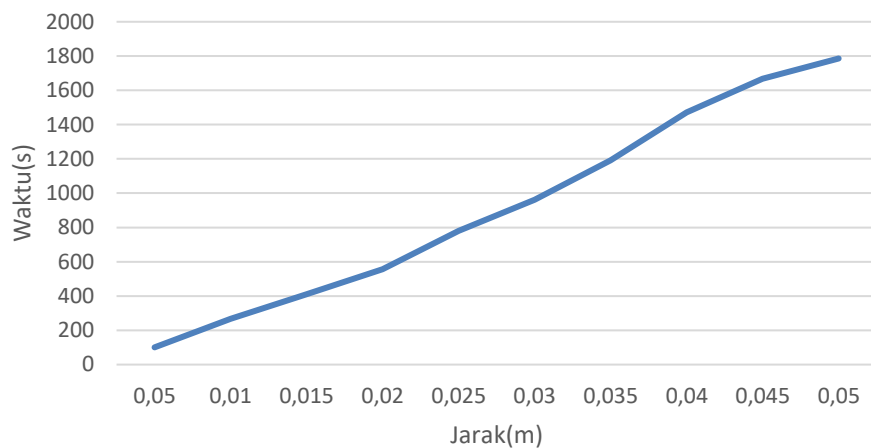
3. Durasi waktu penyinaran UVC pada arah belakang



Gambar 4.34 Jarak terhadap durasi waktu penyinaran lampu UVC pada arah belakang

Pada **Gambar 4.34** menunjukkan durasi waktu penyinaran lampu UVC pada arah belakang *mobile robot* yang dipengaruhi oleh jarak, adapun durasi waktu penyinaran terlama terdapat pada jarak 0,5m dengan waktu 1470 detik dan untuk waktu penyinaran tercepat terdapat pada jarak 0,05m dengan waktu 92 detik.

4. Durasi waktu penyinaran UVC pada arah kanan



Gambar 4.35 Jarak terhadap durasi waktu penyinaran lampu UVC pada arah kanan

Pada **Gambar 4.35** menunjukkan durasi waktu penyinaran lampu UVC pada arah kanan *mobile robot* yang dipengaruhi oleh jarak, adapun durasi waktu penyinaran terlama terdapat pada jarak 0,5m dengan waktu 1785 detik dan untuk waktu penyinaran tercepat terdapat pada jarak 0,05m dengan waktu 100 detik.

Berdasarkan data grafik hubungan jarak terhadap durasi waktu penyinaran dapat disimpulkan bahwa semakin dekat jarak penyinaran lampu UVC maka semakin cepat waktu penyinaran yang dibutuhkan.

4.2.4. Hasil Percobaan Kertas Dosis sinar UVC pada dosis 500J/m²

Selain menggunakan sensor untuk mendeteksi besaran intensitas, kertas dosis dipakai untuk mengetahui berapa lama durasi waktu

penyinaran lampu UVC pada *mobile robot* dengan indikator perubahan warna pada kertas uji. Adapun jarak yang diukur menggunakan dosis adalah 0,25m dan 0,5m.



Gambar 4.36 Uji dosis sinar UVC

Dari hasil pengujian, untuk jarak 0,25m dibutuhkan durasi waktu 32 menit untuk berubah warna pada indikator dosis $500\text{J}/\text{m}^2$ sedangkan untuk jarak 0,5m dibutuhkan durasi waktu 67 menit untuk berubah warna pada indikator dosis $500\text{J}/\text{m}^2$.

BAB V

PENUTUP

5.1. Kesimpulan

Dari pembahasan dapat ditarik kesimpulan sebagai berikut.

1. Hasil rancangan mobile robot disinfektan UVC mempunyai 2 buah lampu UVC sebagai alat sterilisasi ruangan. 1 mikrokontroller ESP-32 yang terhubung langsung pada server cloud untuk pengendalian melalui internet, serta 1 buah kamera ESP-32 sebagai monitoring jarak jauh keadaan sekitar mobile robot.
2. Hasil pengujian respon time (delay) pada mobile menggunakan berbagai provider robot menunjukkan bahwa provider Telkomsel memiliki delay terendah dengan rata rata nilai delay sebesar 0,491s sedangkan untuk provider Indosat, XL dan Tri memiliki rata rata nilai delay masing-masing sebesar 0,554s, 0,622s dan 1,022s.
3. Pengujian intensitas dari lampu UVC pada arah depan mendapatkan hasil terbaik berada pada jarak 0,05m dengan nilai $2,78\text{W}/\text{m}^2$ sedangkan untuk jarak 0,5m nilai intensitas sebesar $0,17\text{W}/\text{m}^2$. Kemudian berdasarkan nilai dosis $250\text{j}/\text{m}^2$ didapat hasil terbaik durasi waktu penyinaran pada arah depan mobile robot dengan waktu 89 detik pada jarak 0,05m dan waktu 1470 detik pada jarak 0,5m.

5.2. Saran

1. Pada penelitian ini menggunakan UVC 8W sebagai alat sterilisasi dinilai kurang efektif karena jarak penyinaran terlalu kecil, oleh karena itu penggunaan lampu UVC dengan daya yang lebih besar dapat memberikan efek yang lebih baik.
2. Untuk penelitian selanjutnya mungkin bisa menggunakan teknologi 5G yang bisa menekan delay pada mobil robot.
3. Diharapkan pengembangan kedepannya penelitian mobile robot ini bisa menggunakan sistem autonomous.

DAFTAR PUSTAKA

- Aprylia. 2020. Smart House Berbasis Web Server Menggunakan Esp 32 Sebagai Door Lock Menggunakan Face Lock. Medan: Universitas Sumatera Utara
- Aswar P., Iswandi. 2014. Teknologi Robot. Makassar: UIN Alauddin Makassar
- Ayman DA, Marwa AA. Mobile Robot Monitoring System based on IoT. Journal of Xi'an University of Architecture & Technology. 2020;12:5438-5448.
- Begum M., Hocking AD, Miskelly D. Inactivation of Food Spoilage fungi by Ultraviolet (UVC) Irradiation. International Journal of Food Mikrobiology.2009;129:74-77
- Cahyo, MO. Kurniawan, Y. PEMANFAATAN UV-C CHAMBER SEBAGAI DISINFEKTAN ALAT PELINDUNG DIRI UNTUK PENCEGAHAN PENYEBARAN VIRUS CORONA, Jurnal Abdi Insani Universitas Mataram, 7(2020) 87-92
- Eko S. Mobile Robot Dengan Pengontrolan Perintah Suara Android. Jurnal MANUTECH. 2017;14-88.
- Erni D. 2009. Rancang Bangun Program Visualisasi Pergerakan Differential Drive Mobile Robot. Surabaya: Politeknik Elektronika Negeri Surabaya
- Gidari A., S. Sabbatini. SARS-CoV-2 Survival on Surfaces and the Effect of UV-C Light. MDPI. 2021;13, 408.
- Giusti, MM, Wrolstad RE. Characterization and Measurement of Anthocyanin by UV-Visible Spectroscopy. John Willey and Sons. Inc. London. 2010.
- Grieco L., A. Rizzo, S. Colucci, S. Sicari, G. Piro, D.D. Paola, G. Boggia, IoTaided robotics applications: Technological implications, target domains and open issues, Comput. Commun. 54 (2014) 32–47
- Hasyim D. 2016. ANALISIS PERBANDINGAN UNJUK KERJA VIDEO CALL VOIP SERVER KAMAILIO DAN VOIP SERVER TRIXBOX. Surabaya: STIKOM
- Hidaka Y, Kubota K. Study on Sterilization of Grain Surface using UV radiation. Japan Agricultural Research Quarterly. 2006;40:157- 161.
- Muhammad F. 2018. Perancangan Pengaplikasian Internet Of Things Pada Robot Pemotong Rumput Berbasis Arduino Mega 2560 Dengan Esp8266. Palembang: Universitas Sriwijaya
- Pandu, A. 2017. RANCANG BANGUN ROBOT PENGAMBIL OBJEK BAWAH AIR. Surabaya: ITS

- Ricky E., Andy R. 2010 Virtual Private Server (VPS) Sebagai Alternatif Pengganti Dedicated Server. Surabaya: ITATS
- Sarinaningsih. 2018. Pengaruh Intensitas, Lama Waktu Penyinaran Dan Posisi Sumber Sinar Ultraviolet Terhadap Reduksi Jumlah Bakteri E.Coli Pada Air Sumur. Mataram: Universitas Mataram
- Wang Z., W. Qiang. A Handbook of 2019-nCoV Pneumonia Control and Prevention. Hubei Science and Technologi Press. China. 2020
- Wilianto, dan Ade Kurniawan. 2018. Sejarah, Cara Kerja dan Manfaat *Internet of Things*. Paper. Batam: Program Studi Teknik Informatika Universitas Universal.
- Wladyslaw J. 2020. COVID-19 Coronavirus Ultraviolet Susceptibility. Purple Inc. New York. 2020.
- Yahya, A. Mukmin. 2019. RANCANG BANGUN SISTEM BANGUNAN CERDAS BERBASIS INTERNET OF THINGS. Makassar: Politeknik Negeri Ujung Pandang
- Yeni R. 2019. Efektivitas Penggunaan Satu Dan Dua Tabung Ultraviolet Terhadap Penurunan Angka Kuman Udara Di Laboratorium Bakteriologi Jurusan Analis Kesehatan Poltekkes Kemenkes Yogyakarta. Yogyakarta: Politeknik Kesehatan Kementerian Kesehatan

Lampiran 1 Program pada ESP-32, ESP32-CAM, dan Website

1. Program ESP-32

```
#include <WiFi.h>
#include <HTTPClient.h>

const char* ssid = "ayeaye";
const char* password = "sagiri";

//Your Domain name with URL path or IP address with path
String serverName = "http://ssrc.web.id/api/motorstatus.php";

// the following variables are unsigned longs because the time,
measured in
unsigned long lastTime = 0;
// Timer set to 10 minutes (600000)
//unsigned long timerDelay = 600000;
// Set timer to 5 seconds (5000)
unsigned long timerDelay = 100;
int m1 = 33;
int m2 = 25;
int m3 = 26;
int m4 = 27;
int uvc = 32;

void setup() {
  Serial.begin(115200);

  pinMode(m1, OUTPUT);
  pinMode(m2, OUTPUT);
  pinMode(m3, OUTPUT);
  pinMode(m4, OUTPUT);
  pinMode(uvc, OUTPUT);

  WiFi.begin(ssid, password);
  Serial.println("Connecting");
  while(WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.print("Connected to WiFi network with IP Address: ");
  Serial.println(WiFi.localIP());

  Serial.println("Timer set to 5 seconds (timerDelay variable), it
will take 5 seconds before publishing the first reading.");
}

void loop() {

  //Send an HTTP POST request every 10 minutes
  if ((millis() - lastTime) > timerDelay) {
    //Check WiFi connection status
```

```

if(WiFi.status()== WL_CONNECTED){
  HTTPClient http;

  String serverPath = serverName ;

  // Your Domain name with URL path or IP address with path
  http.begin(serverPath.c_str());

  // Send HTTP GET request
  int httpResponseCode = http.GET();

  if (httpResponseCode>0) {
    Serial.print("HTTP Response code: ");
    Serial.println(httpResponseCode);
    String payload = http.getString();
    Serial.println(payload);

    if (payload == "1") {
      maju();
    }else if (payload == "2") {
      mundur();
    }
    else if (payload == "3") {
      kiri();
    }
    else if (payload == "4") {
      kanan();
    }
    else if (payload == "5") {
      uvcon();
    }
    else if (payload == "6") {
      uvcoff();
    }
    else if (payload == "0") {
      berhenti();
    }
  }
  else {
    Serial.print("Error code: ");
    Serial.println(httpResponseCode);
    offstopmr();
  }
  // Free resources
  http.end();
}
else {
  Serial.println("WiFi Disconnected");
}
lastTime = millis();
}
}

void maju()
{

```

```

    digitalWrite(m1, HIGH);
    digitalWrite(m2, LOW);
    digitalWrite(m3, HIGH);
    digitalWrite(m4, LOW);
}

void mundur()
{
    digitalWrite(m1, LOW);
    digitalWrite(m2, HIGH);
    digitalWrite(m3, LOW);
    digitalWrite(m4, HIGH);
}

void kiri()
{
    digitalWrite(m1, LOW);
    digitalWrite(m2, HIGH);
    digitalWrite(m3, HIGH);
    digitalWrite(m4, LOW);
}

void kanan()
{
    digitalWrite(m1, HIGH);
    digitalWrite(m2, LOW);
    digitalWrite(m3, LOW);
    digitalWrite(m4, HIGH);
}

void berhenti()
{
    digitalWrite(m1, LOW);
    digitalWrite(m2, LOW);
    digitalWrite(m3, LOW);
    digitalWrite(m4, LOW);
}

void uvcon()
{
    digitalWrite(uvc, HIGH);
}

void uvcoff()
{
    digitalWrite(uvc, LOW);
}

void offstopmr()
{
    digitalWrite(m1, LOW);
    digitalWrite(m2, LOW);
    digitalWrite(m3, LOW);
    digitalWrite(m4, LOW);
    digitalWrite(uvc, LOW);
}

```

2. Program ESP-32 CAM

```
#include "esp_camera.h"
#include <WiFi.h>

//
// WARNING!!! Make sure that you have either selected ESP32 Wrover
Module,
//           or another board which has PSRAM enabled
//

// Select camera model
// #define CAMERA_MODEL_WROVER_KIT
// #define CAMERA_MODEL_ESP_EYE
// #define CAMERA_MODEL_M5STACK_PSRAM
// #define CAMERA_MODEL_M5STACK_WIDE
#define CAMERA_MODEL_AI_THINKER

#include "camera_pins.h"

const char* ssid = "ayeaye";
const char* password = "pahasagiri";

void startCameraServer();

void setup() {
  Serial.begin(115200);
  Serial.setDebugOutput(true);
  Serial.println();

  camera_config_t config;
  config.ledc_channel = LEDC_CHANNEL_0;
  config.ledc_timer = LEDC_TIMER_0;
  config.pin_d0 = Y2_GPIO_NUM;
  config.pin_d1 = Y3_GPIO_NUM;
  config.pin_d2 = Y4_GPIO_NUM;
  config.pin_d3 = Y5_GPIO_NUM;
  config.pin_d4 = Y6_GPIO_NUM;
  config.pin_d5 = Y7_GPIO_NUM;
  config.pin_d6 = Y8_GPIO_NUM;
  config.pin_d7 = Y9_GPIO_NUM;
  config.pin_xclk = XCLK_GPIO_NUM;
  config.pin_pclk = PCLK_GPIO_NUM;
  config.pin_vsync = VSYNC_GPIO_NUM;
  config.pin_href = HREF_GPIO_NUM;
  config.pin_sscb_sda = SIOD_GPIO_NUM;
  config.pin_sscb_scl = SIOC_GPIO_NUM;
  config.pin_pwdn = PWDN_GPIO_NUM;
  config.pin_reset = RESET_GPIO_NUM;
  config.xclk_freq_hz = 2000000;
  config.pixel_format = PIXFORMAT_JPEG;
  //init with high specs to pre-allocate larger buffers
  if(psramFound()){
    config.frame_size = FRAMESIZE_UXGA;
    config.jpeg_quality = 10;
    config.fb_count = 2;
  }
}
```

```

} else {
  config.frame_size = FRAMESIZE_SVGA;
  config.jpeg_quality = 12;
  config.fb_count = 1;
}

#if defined(CAMERA_MODEL_ESP_EYE)
  pinMode(13, INPUT_PULLUP);
  pinMode(14, INPUT_PULLUP);
#endif

// camera init
esp_err_t err = esp_camera_init(&config);
if (err != ESP_OK) {
  Serial.printf("Camera init failed with error 0x%x", err);
  return;
}

sensor_t * s = esp_camera_sensor_get();
//initial sensors are flipped vertically and colors are a bit
saturated
if (s->id.PID == OV3660_PID) {
  s->set_vflip(s, 1); //flip it back
  s->set_brightness(s, 1); //up the blightness just a bit
  s->set_saturation(s, -2); //lower the saturation
}
//drop down frame size for higher initial frame rate
s->set_framesize(s, FRAMESIZE_QVGA);

#if defined(CAMERA_MODEL_M5STACK_WIDE)
  s->set_vflip(s, 1);
  s->set_hmirror(s, 1);
#endif

WiFi.begin(ssid, password);

while (WiFi.status() != WL_CONNECTED) {
  delay(500);
  Serial.print(".");
}
Serial.println("");
Serial.println("WiFi connected");

startCameraServer();

Serial.print("Camera Ready! Use 'http://");
Serial.print(WiFi.localIP());
Serial.println("' to connect");
}

void loop() {
  // put your main code here, to run repeatedly:
  delay(10000);
}

```

3. Program Website

a. Index.php

```
<?php
define("BASEPATH", dirname(__FILE__));
include_once('header.php');
include_once('link.php');
include_once('footer.php');

if(isset($_GET['pesan'])){
    if($_GET['pesan']=="belumlogin"){
        echo '<script>alert("silahkan login")</script>';
    }
}
?>

```

b. Welcome.php

```
<!--
Into this file, we create a layout for welcome page.
-->

<?php
define("BASEPATH", dirname(__FILE__));
include_once('link.php');
include_once('header1.php');
require_once('datauser.php');
require_once('api/caminfo.php');

?>
<div class="jumbotron">
    <center>
        <h1>Sistem pengendalian Mobile Robot Disinfektan UVC</h1>
    </center>

    <center>
<div class="row">

    <div class="column" style="background-color:#fff; width: 31.5%;
height: 400px; ">
    <br>
    <br>

        <center><h2>Control</h2> </center>
    <center>

        <form action="" method="get">
            <button type="button" id="M1-on" class="button
button1" >Maju</button>
```



```

        <button type="button" id="M2-on" class="button
button3" >Mundur</button>
        <button type="button" id="M3-on" class="button
button1" >Kiri</button>
        <button type="button" id="M4-on" class="button
button3" >Kanan</button>
        <br>
        <br>
        <button type="button" id="off" class="button
button3" >Stop</button>
        <br>
        <br>
        <button type="button" id="uvcon" class="button
button3" >UVC On</button>
        <button type="button" id="uvcoff" class="button
button3" >UVC Off</button>
    </form>
</center>

</div>

<div class="column" style="background-color:#fff; width: 55%;
height: 400px;">
    <center>
        <iframe width=400px height=300px src="<?php echo $urlcam;?" >
    </center>

</div>

</div>

</div>

</div>

</center>

<script>
    document.getElementById('M1-
on').addEventListener('mouseup', function() {
        var url =
"http://127.0.0.1/api/updatemotor.php?id=1&status=1";
        $.getJSON(url, function(data) {
            console.log(data);
        });
    });

    document.getElementById('M1-
on').addEventListener('mousedown', function() {

```

```

        var url =
"http://127.0.0.1/api/updatemotor.php?id=1&status=0";
        $.getJSON(url, function(data) {
            console.log(data);
        });
    });

    document.getElementById('M2-on').addEventListener('click',
function() {
        var url =
"http://127.0.0.1/api/updatemotor.php?id=1&status=2";
        $.getJSON(url, function(data) {
            console.log(data);
        });
    });

    document.getElementById('M3-on').addEventListener('click',
function() {
        var url =
"http://127.0.0.1/api/updatemotor.php?id=1&status=3";
        $.getJSON(url, function(data) {
            console.log(data);
        });
    });

    document.getElementById('M4-on').addEventListener('click',
function() {
        var url =
"http://127.0.0.1/api/updatemotor.php?id=1&status=4";
        $.getJSON(url, function(data) {
            console.log(data);
        });
    });
    document.getElementById('off').addEventListener('click',
function() {
        var url =
"http://127.0.0.1/api/updatemotor.php?id=1&status=0";
        $.getJSON(url, function(data) {
            console.log(data);
        });
    });
    document.getElementById('uvcon').addEventListener('click',
function() {
        var url =
"http://127.0.0.1/api/updatemotor.php?id=1&status=5";
        $.getJSON(url, function(data) {
            console.log(data);
        });
    });

document.getElementById('uvcoff').addEventListener('click',
function() {
        var url =
"http://127.0.0.1/api/updatemotor.php?id=1&status=6";
        $.getJSON(url, function(data) {

```

```

        console.log(data);
    });
});
</script>
</div>

```

c. Header1.php

```

<!--
this is second header file which is visible after login.
-->

<?php
include_once('link.php');
session_start();
$email = $_SESSION['email'];
if($_SESSION['email']==""){
    header("location:index.php?pesan=belumlogin");
}
?>

<nav class="navbar navbar-default">
    <div class="container-fluid">
        <div class="navbar-header" style="height:80px; margin-
top:10px; " >
            <img src = '/img/unhas.png' height="60px" width="60px"
/> <img src = '/img/5g.png' height="60px" width="90px" />
        </div>
        <div class="dropdown navbar-right" id="right">
            <button class="btn btn-primary dropdown-toggle" type="button"
data-toggle="dropdown"><?php echo $email;?>
            <span class="caret"></span></button>
            <ul class="dropdown-menu">
                <li><a href="logout.php">Logout</a></li>
                <li><a href="sensor.php">Sensor UV</a></li>
                <li><a href="welcome.php">Home</a></li>
            </ul>
        </div>
    </div>
</nav>

```

d. Link.php

```

<!--
Here i put all the link like a css, js for project.
-->

<link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstra
p.min.css">
<link rel="stylesheet" type="text/css" href="css/style.css">
<link rel="stylesheet" type="text/css" href="css/account.css">
<script type="text/javascript" src="js/jquery-3.3.1.js"></script>

```

```
<script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.
min.js"></script>
```

e. Datauser.php

```
<?php
require_once('connection.php');

$id = $_SESSION['id'];
$username = $email = '';
$sql = "SELECT * FROM user WHERE id='$id'";
$result = mysqli_query($conn, $sql);
if(mysqli_num_rows($result) > 0)
{
    while($row = mysqli_fetch_assoc($result))
    {
        $username = $row["username"];
        $email = $row["email"];
    }
}

?>
```

f. Footer.php

```
<?php
defined("BASEPATH") or exit("No direct access allowed");
include_once('link.php');

?>

<div class="footer">
    <p>Teknik Mesin | Universitas Hasanuddin 2021</p>
</div>
```

g. Connection.php

```
<?php
defined("BASEPATH") or exit("No direct access allowed");

$conn = mysqli_connect("localhost", "root", "", "mrduvc");

if(!$conn)
{
    echo "Database connection failed...";
}

?>
```

h. Header.php

```
<!--
```

this is header file which is visible in registration and login page.

```
-->
<?php
defined("BASEPATH") or exit("No direct access allowed");
include_once('link.php');

?>

<nav class="navbar navbar-default">
  <div class="container-fluid">
    <div class="navbar-header">
      <a href="index.php" class="navbar-brand">Sistem
Kontrol Mobile Robot Disinfektan UVC</a>
    </div>
    <ul class="nav navbar-nav navbar-right">
      <li><a href="login.php">Login</a></li>
      <li><a href="registration.php">Registration</a></li>
    </ul>
  </div>
</nav>
```

i. Login_code.php

```
<!--
Here, we write code for login.
-->
<?php
define("BASEPATH", dirname(__FILE__));
require_once('connection.php');
$email = $password = $pwd = '';

$email = $_POST['email'];
$pwd = $_POST['password'];
$password = MD5($pwd);
$sql = "SELECT * FROM user WHERE email='$email' AND
password='$password'";
$result = mysqli_query($conn, $sql);
if(mysqli_num_rows($result) > 0)
{
  while($row = mysqli_fetch_assoc($result))
  {
    $id = $row["id"];
    $email = $row["email"];
    session_start();
    $_SESSION['id'] = $id;
    $_SESSION['email'] = $email;
  }
  header("Location: welcome.php");
}
else
{
  header("location:login.php?pesan=gagal");
}
```

```
}  
?>
```

j. Login.php

```
<!--  
Into this file, we create a layout for login page.  
-->
```

```
<?php  
define("BASEPATH", dirname(__FILE__));  
include_once('header.php');  
include_once('link.php');  
if(isset($_GET['pesan'])){  
    if($_GET['pesan']=="gagal"){  
        echo '<script>alert("username atau password  
salah")</script>';  
    }  
}  
?>  
  
<div id="frmRegistration">  
<form class="form-horizontal" method="POST"  
action="login_code.php">  
    <h1>User Login</h1>  
  
    <div class="form-group">  
        <label class="control-label col-sm-2"  
for="email">Email:</label>  
        <div class="col-sm-4">  
            <input type="email" class="form-control" name="email"  
id="email" placeholder="Enter email">  
        </div>  
    </div>  
    <div class="form-group">  
        <label class="control-label col-sm-2"  
for="pwd">Password:</label>  
        <div class="col-sm-4">  
            <input type="password" class="form-control" name="password"  
id="pwd" placeholder="Enter password">  
        </div>  
    </div>  
    <div class="form-group">  
        <div class="col-sm-offset-2 col-sm-10">  
            <button type="submit" name="login" class="btn btn-  
primary">Login</button>  
        </div>  
    </div>  
</form>  
</div>
```

k. api/caminfo.php

```
<?php  
require_once('connection.php');
```

```

$query = "SELECT * from cam"; // Select all data
in table "status"
$result = $conn->query($query);

$row = mysqli_fetch_assoc($result);
{
    $idcam = $row["id"];
    $urlcam = $row["url"];
}
?>

```

l. api/connection.php

```

<?php
defined("BASEPATH") or exit("No direct access allowed");

$conn = mysqli_connect("localhost","root","","mrduvc");

if(!$conn)
{
    echo "Database connection failed...";
}
?>

```

m. api/motorstatus.php

```

<?php
define("BASEPATH", dirname(__FILE__));
require_once('connection.php');

$query = "SELECT * from motor"; // Select all data
in table "status"
$result = $conn->query($query);

$row = mysqli_fetch_assoc($result);
{
    $id1 = $row["id"];
    $status1 = $row["status"];
}
mysqli_query($conn, "INSERT INTO datadelay (ok) VALUES ('1')");

echo $status1 ;

?>

```

n. api/offmotor.php

```

<?php

```

```

define("BASEPATH", dirname(__FILE__));
require_once('connection.php');

$response = array();

// Check if we got the field from the user
if ( isset($_GET['status'])) {

    $status= $_GET['status'];

    // Fire SQL query to update LED status data by id
    $result = mysqli_query($conn,"UPDATE motor SET status=
'$status' ");

    // Check for succesfull execution of query and no results
found
    if ($result) {
        // successfully updation of LED status (status)
        $response["success"] = 1;
        $response["message"] = "Motor Status successfully
updated.";

        // Show JSON response
        echo json_encode($response);
    } else {

    }
} else {
    // If required parameter is missing
    $response["success"] = 0;
    $response["message"] = "Parameter(s) are missing. Please check
the request";

    // Show JSON response
    echo json_encode($response);
}
?>

```

o. api/updatecam.php

```

<?php
define("BASEPATH", dirname(__FILE__));
require_once('connection.php');

header("Access-Control-Allow-Origin: *");
header("Content-Type: application/json; charset=UTF-8");
//Creating Array for JSON response
$response = array();

// Check if we got the field from the user
if (isset($_GET['id']) && isset($_GET['url'])) {

```



```

$id = $_GET['id'];
$url= $_GET['url'];

// Fire SQL query to update cam data by id
$result = mysqli_query($conn,"UPDATE cam SET url= '$url' WHERE
id = '$id'");

// Check for succesfull execution of query and no results
found
if ($result) {
    // successfully upadation of url (url)
    $response["success"] = 1;
    $response["message"] = "cam Data successfully updated.";

    // Show JSON response
    echo json_encode($response);
} else {

}
} else {
    // If required parameter is missing
    $response["success"] = 0;
    $response["message"] = "Parameter(s) are missing. Please check
the request";

    // Show JSON response
    echo json_encode($response);
}
?>

```

p. api/updatemotor.php

```

<?php

define("BASEPATH", dirname(__FILE__));
require_once('connection.php');

$response = array();

// Check if we got the field from the user
if (isset($_GET['id']) && isset($_GET['status'])) {

    $id = $_GET['id'];
    $status= $_GET['status'];

    // Fire SQL query to update LED status data by id
    $result = mysqli_query($conn,"UPDATE motor SET status=
'$status' WHERE id = '$id'");

```

```

    // Check for succesfull execution of query and no results
found
    if ($result) {
        // successfully updation of LED status (status)
        $response["success"] = 1;
        $response["message"] = "Motor Status successfully
updated.";

        // Show JSON response
        echo json_encode($response);
    } else {

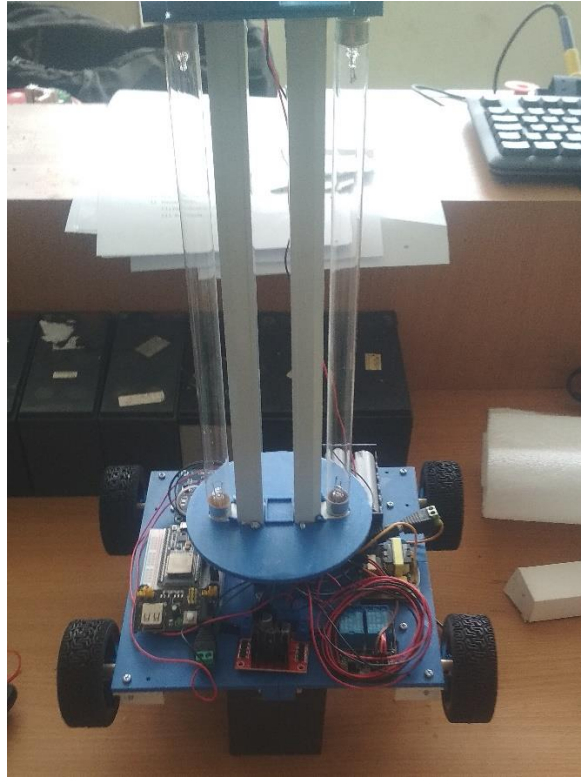
    }
} else {
    // If required parameter is missing
    $response["success"] = 0;
    $response["message"] = "Parameter(s) are missing. Please check
the request";

    // Show JSON response
    echo json_encode($response);
}
?>

```

Lampiran 2 Foto Kegiatan

1. Proses Perakitan dan pengujian Mobile Robot

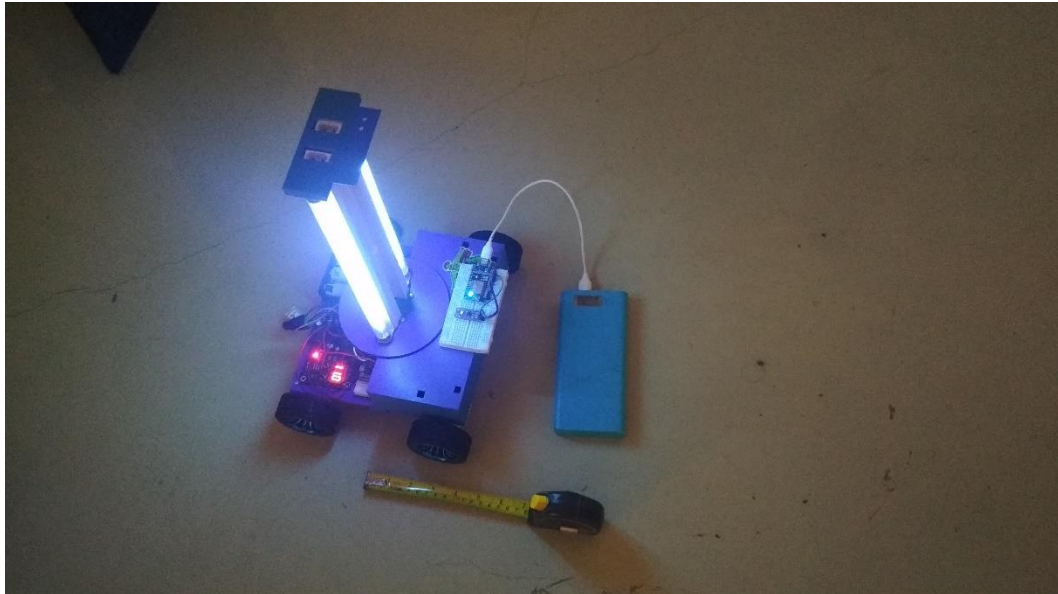


a.1. Perakitan Mobile Robot



b.1. Pengujian Mobile robot

2. Proses Pengambilan data



a.2 Pengambilan data Intensitas UVC

| | id | time | ok |
|--------------------------|------|-------------------------|----|
| <input type="checkbox"/> | 1862 | 2021-12-03 08:13:56.003 | 3 |
| <input type="checkbox"/> | 1863 | 2021-12-03 08:13:56.623 | 3 |
| <input type="checkbox"/> | 1864 | 2021-12-03 08:13:57.239 | 3 |
| <input type="checkbox"/> | 1865 | 2021-12-03 08:13:57.839 | 3 |
| <input type="checkbox"/> | 1866 | 2021-12-03 08:13:58.439 | 3 |
| <input type="checkbox"/> | 1867 | 2021-12-03 08:13:59.064 | 3 |
| <input type="checkbox"/> | 1868 | 2021-12-03 08:13:59.679 | 3 |
| <input type="checkbox"/> | 1869 | 2021-12-03 08:14:00.281 | 3 |
| <input type="checkbox"/> | 1870 | 2021-12-03 08:14:00.912 | 3 |
| <input type="checkbox"/> | 1871 | 2021-12-03 08:14:01.537 | 3 |
| <input type="checkbox"/> | 1872 | 2021-12-03 08:14:02.136 | 3 |
| <input type="checkbox"/> | 1873 | 2021-12-03 08:14:02.760 | 3 |
| <input type="checkbox"/> | 1874 | 2021-12-03 08:14:03.351 | 3 |
| <input type="checkbox"/> | 1875 | 2021-12-03 08:14:03.999 | 3 |

b.2. Pengambilan data delay tiap provider melalui datalogger MySQL