

BAB I

PENDAHULUAN

A. Latar Belakang

Sistem Operasi Android merupakan suatu perangkat lunak yang telah berkembang pesat pada saat ini, Oleh karena itu banyak software developer yang menjadikan android sebagai terobosan baru dalam bidang perangkat lunak atau sistem operasi pada smartphone, Seiring meningkatnya pengguna Smartphone Android sebagian masyarakat menginginkan sesuatu yang lain dari pengguna fungsi smartphone yang sekedar digunakan untuk SMS, Chat, Telepon, Internet dan Game.

Seiring dengan naiknya keinginan masyarakat dan teknologi yang semakin canggih, di buatlah suatu sistem remot control, sistem kontrol biasanya menuju kepada alat kontrol dengan jarak tertentu atau alat kontrol nirkabel. Remote control dapat memudahkan manusia untuk pengendalian suatu objek atau yang lainnya.

Mobil remote control menggunakan remot yang menggunakan sinyal analog yang diatur pada frekuensi tertentu. Pada remot biasanya memanfaatkan gelombang radio frekuensi sebagai media penghubungnya. Permasalahannya ketika remot mobil tersebut rusak atau hilang. Maka akan lebih mudah ketika mobil dikontrol dengan aplikasi pada smartphone android.

Ada beberapa tempat dan keadaan dimana manusia terbatas untuk masuk kedalam tempat tersebut dan mempunyai batasan penglihatan, batasan tersebut disebabkan oleh tempat tersebut berbahaya, tempat tersebut tidak bisa dijangkau manusia dan yang lain sebagainya. Kamera bisa menjadi solusi untuk melakukan penglihatan pada tempat-tempat tertentu dikarenakan batasan penglihatan manusia. Contohnya pada reruntuhan manusia tidak bisa langsung untuk mencari korban ditempat, ditakutkan terjadi runtuh susulan atau korban dibawah reruntuhan bisa terinjak, maka dari itu penulis ingin membuat realisasi sistem penglihatan menggunakan remote control mobil berbasis mikrokontroler ESP32 Cam dan android untuk mencari korban pada reruntuhan.

B. Rumusan Masalah

Berdasarkan latar belakang masalah yang ada dan fokus pekerjaan, maka dapat dikemukakan permasalahan pokok yang direalisasikan adalah :

1. Bagaimana membuat Aplikasi terhubung dengan mikrokontroler arduino dan Esp32 Cam.?
2. Bagaimana Aplikasi mengontrol mobil melalui Esp32 Cam.?

C. Batasan Masalah

Berikut ini ruang lingkup batasan masalah dari perancangan tugas akhir iyalah sebagai berikut:

1. RC mobil tidak dilengkapi oleh pelindung air bila ada keadaan tergenang air di medan jalan sehingga kemungkinan terjadi kerusakan.
2. Jika menggunakan jaringan Wifi, harus dalam satu jaringan HotSpot.

D. Tujuan Penelitian

1. Membangun pengendali mobil remot control menggunakan Esp 32 Cam berbasis IOT.
2. Mengaplikasiakn Esp 32 cam dalam perancangan mobil remot control.

E. Manfaat Penelitian

Manfaat dari penelitian ini adalah:

1. Pengguna dapat mengendalikan mobil remot kontrol melalui aplikasi smartphone adndroid.
2. Memberikan kemudahan dan kepraktisan seseorang dalam mencari sesuatu atau benda yang tertimpa musibah di sebuah reruntuhan.
3. Manfaat bagi pembaca.

Dengan penelitian ini diharapkan dapat menjadi referensi bagi para pembaca.

F. Sistematika Penulisan

Agar pembahasan lebih sistematis, maka tulisan ini dibuat dalam lima bab, yaitu:

BAB I PENDAHULUAN

Berisi tentang uraian latar belakang pemilihan judul Rancang Bangun Mobil Remot Control Berkamera Menggunakan ESP32 Cam Berkamera Berbasis

IOT, rumusan masalah, batasan masalah, tujuan penelitian dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Dalam bab ini berisikan tentang teori-teori yang berkaitan dengan *mobil remot control Berkamera* , Bahasa pemrograman python, library, PyCharm IDE.

BAB III METODE PENELITIAN

Dalam bab ini menjelaskan mengenai tahapan-tahapan yang dilalui dalam penyelesaian penelitian ini, yaitu tempat penelitian, waktu penelitian, metode pengumpulan data, alat, dan bahan penelitian, tahapan penelitian, metode pengujian serta gambaran desain sistem yang akan dirancang dan dibuat.

BAB IV HASIL PENELITIAN DAN PEMBAHASAN

Dalam bab ini menjelaskan mengenai rancangan sistem yang telah dibuat, metode dan teknik pengujian.

BAB V KESIMPULAN DAN SARAN

Dalam bab ini dipaparkan mengenai kesimpulan dan saran dari hasil.

BAB II

TUJUAN PENELITIAN

A. Penelitian Terdahulu

Pada Penelitian Yanolanda Susantry, (2018) yang berjudul “Mobil Remot Control Berbasis Arduino Dengan Sistem Kendali Menggunakan Android” Memiliki kelebihan yaitu menggunakan bloututh HC-05 dan bisa di kontrol menggunakan Android. Adapun Kelemahanya atau kekuranganya menggunakan mobil bloututh memiliki jangkauan jaraknya yang lebih sempit.

Pada Penelitian Ayyub Fadilah, A. (2017). ” Rancang Bangun Mobil Remote Conrol dengan Smartphone Android Menggunakan Sensor US-016 Berbasis Arduino Uno” Penelitian ini menghasilkan prototype model sebuah alat elektronik yang dikendalikan dengan smartphone android melalui koneksi bluetooth. perbedaan yang penulis rancang yaitu menggunakan esp32 cam.

Pada penelitian Andi Anugrah, (2011) yang berjudul, “kendali robot mobile dengan menggunakan stick” bertujuan untuk membuat perangkat keras, perangkat lunak dan mekanik untuk aktivasi pergerakan robot. Robot ini enggunakan stick sebagai media pengendali gerak.

B. Android

Android merupakan sebuah sistem operasi seluler yang didasarkan pada versi modifikasi dari kernel Linux dan perangkat sumber terbuka lainnya. Android dirancang untuk perangkat seluler terutama layar sentuh seperti [smartphone](#) dan tablet. Sistem operasi ini pertama kali diluncurkan pada bulan September 2008, di

mana Android dikembangkan oleh Open Handset Alliance yang disponsori secara komersial oleh Google.

Selain itu Android juga merupakan perangkat lunak gratis dengan sumber terbuka, dalam artian Google memperbolehkan pengguna untuk mengembangkan sistem operasi tersebut. Android juga memiliki toko aplikasi yang bernama Google Play Store. Tentunya bagi Anda yang menggunakan smartphone dengan sistem Android, dapat bebas mendownload aplikasi atau game yang terdapat pada [Google Play Store](#).

Saat ini Android memiliki beberapa versi yang telah dirilis, mulai dari Android 1.0 hingga yang terbaru Android 11. Menariknya dalam sistem operasi ini, terdapat beberapa versi yang menggunakan nama dessert sebagai penanda. Misalnya Android Cup Cake, Donut, Froyo, Jelly Bean, KitKat, Marshmallow, Oreo hingga Pie. Mungkin untuk ke depannya versi Android hanya menggunakan sistem penomoran saja, seperti halnya Android 10 dan Android 11.

C. Remote Control

Remote Control adalah sebuah alat elektronik yang digunakan untuk mengoperasikan sebuah alat dari jarak jauh. Istilah remote control juga sering disingkat menjadi "remote" saja. Remote juga sering kali mengacu pada istilah "controller, donker, doofer, zapper, click-buzz, box, flipper, zippity, clicker, atau changer". Pada umumnya, pengendali jarak jauh digunakan untuk memberikan perintah dari kejauhan kepada televisi atau barang-barang elektronik lainnya seperti system stereo dan pemutar DVD. Remote control untuk perangkat-perangkat ini biasanya berupa benda kecil nirkabel yang dipegang dalam tangan

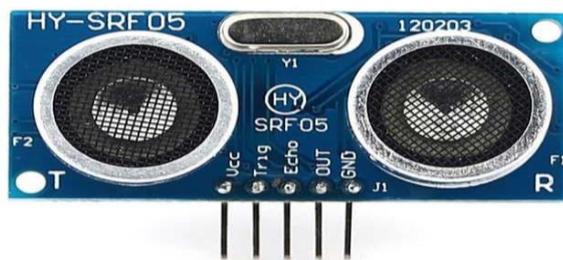
dengan sederetan tombol untuk menyesuaikan berbagai setting. Remote control terdiri dari dua bagian, yaitu bagian transmitter dan bagian receiver. Bagian transmitter berfungsi untuk mengirimkan data informasi perintah, sedangkan bagian receiver berfungsi untuk menerima data informasi untuk diteruskan ke mikrokontroler

D. Internet of Things

IoT (*Internet of Things*) dapat didefinisikan kemampuan berbagai device yang bisa saling terhubung dan saling bertukar data melalui jaringan internet. IoT merupakan sebuah teknologi yang memungkinkan adanya sebuah pengendalian, komunikasi, kerjasama dengan berbagai perangkat keras, data melalui jaringan internet. Sehingga bisa dikatakan bahwa *Internet of Things*(IoT) adalah ketika kita menyambungkan sesuatu (*things*) yang tidak dioperasikan oleh manusia, ke internet (Hardyanto, 2017).

Adanya IoT saat ini sudah sangat umum sehingga kita seringkali lupa bahwa kita dikelilingi oleh barang-barang konvensional yang terhubung ke internet. Salah satu contoh sederhana adalah fungsi ganda smartphone sebagai remote TV atau remote AC. Seiring berkembangnya teknologi, IoT akan mencakup lebih banyak sisi dari kehidupan kita. Tentunya, benda sehari-hari yang terhubung ke internet akan memudahkan kehidupan kita. IoT adalah hal yang menjadi pondasi *smart home* atau *smart living*. Tanpa IoT, benda-benda seperti *smart pet feeder*, *smart LED strip*, IP camera, dan sejenisnya tak akan dapat digunakan.

dipantulkan. Sensor ultrasonik HY-SRF05 sangat cocok diaplikasikan untuk minsys seperti Arduino, Raspberry Pi, dan lain-lain. Versi HC-SR05 ini adalah versi improvement dari HC-SR04, yaitu memiliki akurasi lebih baik khususnya dalam pengukuran jarak jauh.

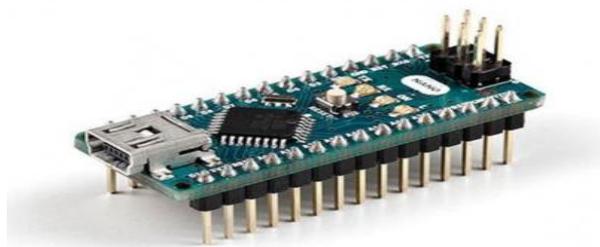


(Gambar 2.2 sensor Ultrasonic)

<https://www.google.com/search?q=+sensor+ultrasonic+hy-srf05>

F. Arduino Nano

Arduino Nano adalah papan mikrokontroler open source yang memiliki ukuran kecil dan mendukung penggunaan breadboard. Arduino Nano dilengkapi dengan 30 header I/O jantan, konektor USB Mini-B, dan 8 pin input analog. Arduino Nano dapat diprogram menggunakan lingkungan pengembangan terintegrasi Perangkat Lunak Arduino (IDE) yang dapat digunakan online maupun offline.



(Gambar 2.3 Arduino Nano)

<https://www.arduinoindonesia.id/2019/01/arduino-nano.html>

G. Driver Motor

Pengendalian kecepatan putar motor DC dapat dilakukan dengan mengatur besar tegangan terminal motor VTM. Metode lain yang biasa digunakan untuk mengendalikan kecepatan motor DC adalah dengan teknik modulasi lebar pulsa atau Pulse Width Modulation (PWM).



(Gambar 2.4. Modul Driver Motor DC)

(Sumber: <https://www.toleinnovator.com/2018/06/kontrol-motor-dc-with-arduino-dan-modul.html>)

H. ESP 32 CAM

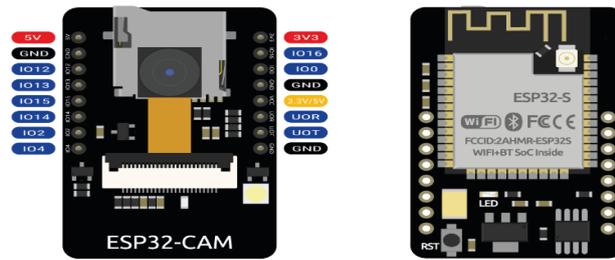
ESP32-CAM merupakan salah satu mikrokontroler yang memiliki fasilitas tambahan berupa bluetooth, wifi, kamera, bahkan sampai ke slot microSD. ESP32-CAM ini biasanya digunakan untuk project IoT (Internet of Things) yang membutuhkan fitur kamera. Modul ESP32CAM memiliki lebih sedikit pin I/O dibandingkan modul ESP32 produk sebelumnya, yaitu ESP32 Wroom. Hal ini dikarenakan sudah banyak pin yang digunakan secara internal untuk fungsi kamera dan fungsi slot kartu microSD. Selain itu, modul ESP32CAM juga tidak memiliki port USB khusus (mengirim program dari port USB komputer). Jadi untuk memprogram modul ini Anda harus menggunakan USB TTL atau kita

dapat menambahkan modul tambahan berupa downloader khusus untuk ESP32-CAM

SPESIFIKASI ESP32-CAM

ESP32-CAM Base Mikrokontrol dari Module ESP32-S , secara garis besar spesifikasinya sebagai berikut :

1. Low-Power Dual-Core 32bit CPU
2. Main Frequency up to 240 MHz
3. Built-in 520KB SRAM, External 4M PSRAM
4. Supports Interfaces such as UART/SPI/I2C/PWM/ADC/DAC
5. Support OV2640 and OV7670 Cameras, Built-in Flash
6. Support Image WiFi Upload
7. Support TF Card
8. Support Multiple Sleep Modes
9. Embedded Lwip and FreeRTOS
10. Mendukung Firmware Over the Air (FOTA)
11. Bluetooth 4.2 with BLE
12. 802.11b/g/n Wi-Fi
13. Support STA/AP/STA+AP Working Mode
14. Support Smart Config/AirKiss One-Click Distribution Network
15. Support Secondary Development
16. Built-in Flash LED
17. 9 GPIO ports



(Gambar 2.5. ESP32-CAM)

https://1.bp.blogspot.com/-5B4xv05vORY/YSYK_YjpDbI/AAAAAAAAAHYU/2SmTOe3TdIcMXD6jASYE TEvUijN_JJ0egCLcBGAsYHQ/s1095/ESP32-CAM-BACK-FRONT.png

I. Kabel Jumper

Kabel jumper merupakan kabel elektrik yang mempunyai pin konektor di setiap ujungnya dan memungkinkan untuk menghubungkan dua komponen yang melibatkan Arduino tanpa memerlukan solder. Intinya, kegunaan *kabel jumper* ini digunakan sebagai konduktor listrik untuk menyambungkan rangkaian listrik.

Kabel jumper biasanya digunakan pada breadboard atau alat prototyping lainnya supaya lebih mudah untuk mengutak-atik rangkaian. Konektor yang terdapat pada ujung kabel terdiri dari konektor jantan (male connector) dan konektor betina (female connector). Konektor female berfungsi untuk masuk dan konektor male berfungsi untuk ditusuk

1. Cara Kerja Kabel Jumper

Sederhananya, kabel jumper bekerja untuk menghantarkan arus listrik dari satu komponen ke komponen lainnya yang dihubungkan. Hal ini terjadi karena di ujung dan di dalam kabel terdapat konduktor listrik kecil yang memang fungsinya untuk menghantarkan listrik.

a. Kelebihan Kabel Jumper

- 1) . Mempunyai konektor di ujungnya yang sangat memudahkan dalam memasang maupun melepas kabel ke komponen
- 2) Harga yang terjangkau
- 3) 3. Mempunyai warna yang bervariasi yang memudahkan dalam membuat rangkaian



(Gambar 2.6. Kabel Jumper)

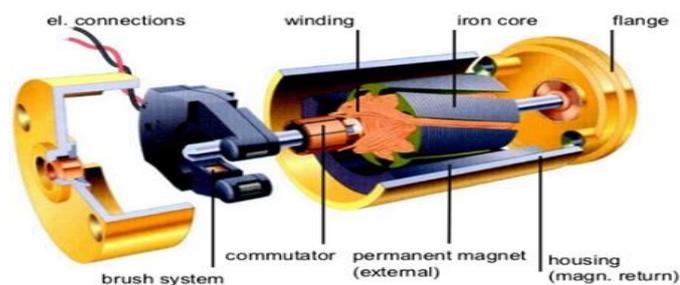
https://blogger.googleusercontent.com/img/b/R29vZ2xl/AVvXsEj0WwBXz4eA8hSjtPsSaFB5rT6FIj2NZbC1xKpTKa38czXK61tjwFuCH7xxdRUAdVzU8Uqx2_f9BHsmJzYNFdLtvG_JuLzjDNs95f0x2LE8T3P5SGU3-il_OntZb9eK-_LiZhd7Ro0V9AILDKJjHsqhZlhX6NJNiAlNFA-vqe-NjQIMU2vGDYcI8-Jg/s500/male%20to%20male.jpg

J. Motor DC

Motor DC adalah motor listrik yang memerlukan suplai tegangan arus searah pada kumparan medan untuk diubah menjadi energi gerak mekanik. Kumparan medan pada motor dc disebut stator (bagian yang tidak berputar) dan kumparan jangkar disebut rotor (bagian yang berputar). Motor arus searah,

sebagaimana namanya, menggunakan arus langsung yang tidak langsung/directunidirectional.

Motor DC adalah piranti elektronik yang mengubah energi listrik menjadi energi mekanik berupa gerak rotasi. Pada motor DC terdapat jangkar dengan satu atau lebih kumparan terpisah. Tiap kumparan berujung pada cincin belah (komutator). Dengan adanya insulator antara komutator, cincin belah dapat berperan sebagai saklar kutub ganda (double pole, double throw switch). Motor DC bekerja berdasarkan prinsip gaya Lorentz, yang menyatakan ketika sebuah konduktor beraliran arus diletakkan dalam medan magnet, maka sebuah gaya (yang dikenal dengan gaya Lorentz) akan tercipta secara ortogonal diantara arah medan magnet dan arah aliran arus. Motor DC tersusun dari dua bagian yaitu bagian diam (stator) dan bagian bergerak (rotor). Stator motor arus searah adalah badan motor atau kutub magnet (sikat-sikat), sedangkan yang termasuk rotor adalah jangkar lilitanya. Pada motor, kawat penghantar listrik yang bergerak tersebut pada dasarnya merupakan lilitan yang berbentuk persegi panjang yang disebut kumparan.



(Gambar 2.7. Simbol Motor DC)

(Sumber: <https://teknikelektronika.com/pengertian-1..motor-dc-prinsip-kerja-dc-motor/>)

K. Arduino IDE

Arduino *Development Environment* merupakan editor teks untuk menulis kode, sebuah area pesan, sebuah konsol, sebuah *toolbar* dengan tombol - tombol untuk fungsi yang umum dan beberapa menu. Arduino *Development Environment* terhubung ke arduino board untuk meng-*upload* program dan juga untuk berkomunikasi dengan arduino board. Perangkat lunak yang ditulis menggunakan Arduino *Development Environment* disebut *sketch*. *Sketch* ditulis pada editor teks. *Sketch* disimpan dengan file berekstensi *.ino*. Area pesan memberikan informasi dan pesan error ketika kita menyimpan atau membuka *sketch*. Konsol menampilkan output teks dari Arduino *Development Environment* dan juga menampilkan pesan error ketika kita mengkompilasi *sketch*. Pada sudut kanan bawah jendela Arduino *Development Environment* menunjukkan jenis board dan port serial yang sedang digunakan. Tombol *toolbar* digunakan untuk mengecek dan meng-*upload sketch*, membuat, membuka, atau menyimpan *sketch*, dan menampilkan serial monitor. Arduino *Development Environment* dapat dilihat pada gambar berikut.



(Gambar 2.8. Arduino IDE)

Berikut ini merupakan tombol-tombol *toolbar* serta fungsinya :

1.  *Verify*, berfungsi untuk mengecek error / kesalahan pada kode program.
2.  *Upload*, berfungsi untuk melakukan *compile* dan *upload*
3.  *New*, berfungsi untuk membuat *sketch* baru.
4.  *Open*, berfungsi untuk membuka *sketch* yang tersimpan.
5.  *Save*, berfungsi untuk menyimpan *sketch*
6.  *Serial Monitor*, berfungsi untuk membuka serial monitor. Serial monitor ini dapat digunakan untuk menampilkan nilai proses, nilai pembacaan, bahkan pesan error.

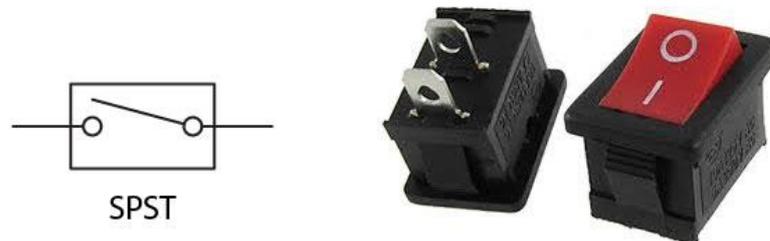
L. Saklar SPST

Saklar adalah sebuah komponen listrik yang berfungsi untuk memutus dan menghubungkan arus listrik. Entah berapa kali saya mengulang kata-kata ini. Yah pokoknya gitu. Aliran listrik akan mengalir apabila suatu kontak dihubungkan dengan kontak lainnya. Sebaliknya, aliran listrik akan terputus apabila hubungan tersebut dibuka atau dipisahkan.

Yang jelas komponen yang digunakan untuk memutus dan menghubungkan arus itu disebut saklar entah bagaimanapun bentuk dan bagaimana proses pemutus dan menghubungkan arus listrik tersebut.

Saklar SPST

Saklar SPST adalah saklar *Single-Pole Single-Throw* atau saklar *kutub tunggal dan gerakan tunggal*. Ini merupakan saklar yang paling sederhana. Hanya memiliki 2 terminal dimana 1 terminal digunakan untuk arus listrik masuk dan 1 terminal lagi untuk arus listrik keluar. Saklar ini memiliki gerakan tunggal didalamnya, ketika tidak diberikan aktuasi maka saklar dalam satu kondisi, dan ketika diberikan aktuasi akan berpindah pada satu kondisi lainnya. Ini adalah saklar yang umum digunakan dan sangat sederhana karena biasanya digunakan untuk mengontrol rangkaian tunggal.



(Gambar 2.9. Saklar SPST)

<https://henduino.github.io/library/ototronik/switch-auto/>

Gambar kiri menunjukkan simbol saklar SPST dan gambar kanan bentuk saklar umum yang sering ditemukan. Saklar bentuk seperti ini dikenal dengan Rocker Switch – SPST.

Fungsi saklar

tidak mungkin sebuah alat di buat hanya untuk gaya-gayaan, begitu juga untuk saklar. banyak fungsi serta manfaat yang didapat ketika kita menggunakan saklar.

Yaitu :

1. Saklar berfungsi untuk meringankan pekerjaan manusia. semua mesin memiliki fungsi sama yaitu meringankan pekerjaan manusia.
2. Mengurangi resiko sengatan listrik.
3. mengurangi resiko korseling listrik. penggunaan saklar untuk Instalasi listrik lebih dianjurkan untuk memutus dan menghubungkan arus dibandingkan menggunakan stop kontak dengan maksud mengurangi resiko korseling listrik.

M. Ban Roda Gearbox

Ban adalah material yang menutupi velg dan berfungsi untuk menyediakan bantalan kendaraan antara ban yang bersentuhan dengan permukaan jalan aspal. Ban merupakan suatu wadah yang isikan udara yang berfungsi menopang beban dari muatan kendaraan dan barang yang ada di kendaraan tersebut. Berikut adalah penjelasan tentang pengertian ban yang dimulai dari fungsi penggunaan ban.

Fungsi kegunaan dasar ban adalah memiliki daya cengkeram yang kuat, kemudahan untuk di kendalikan oleh pengendara dan bantalan ban untuk peredam getaran pada kendaran tersebut. Tujuannya adalah untuk menjaga keselamatan, kenyamanan dan ketahanan pada komponen [ban mobil](#) saat berkendara.

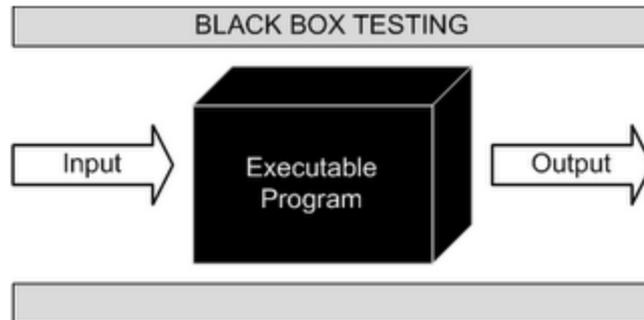
Roda-roda karet berikut kompatibel dengan poros dari *double gearbox* dan *single gearbox*. Masing-masing, dari kiri ke kanan, ber-diameter / lebar : 50 mm / 30 mm (70096, Off Road Tire Set), 58 mm / 15 mm (70145, Narrow Tire Set), 36 mm / 15 mm (70101, Truck Tire Set) dan 56 mm / 25 mm (70111, Sports Tire Set).



(Gambar 2.10. Ban Roda Gearbox)

<http://nextsys.web.id/edukasi/produk/lainnya/motor-dan-gearbox>

N. Black Box Testing



(Gambar 2.11. Black Box Testing)

(Sumber : <https://images.app.goo.gl/9U2JSCs4FRFYesUi9>)

Black box testing adalah pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Jadi dianalogikan seperti kita melihat suatu kotak hitam, kita hanya bisa melihat penampilan luarnya saja, tanpa tau ada apa dibalik bungkusan hitamnya. Sama seperti pengujian *black box*, mengevaluasi hanya dari tampilan luarnya (interfacenya), fungsionalitasnya tanpa mengetahui apa sesungguhnya yang terjadi dalam proses detilnya (hanya mengetahui *input* dan *output*). Dengan *black box testing* dapat diterapkan pada 3 jenis pengujian utama, yaitu *functional testing*, *non-functional testing*, dan *regression testing*.

1. Functional testing

Pengujian ini berfokus pada aspek yang paling kritis dari *software* (*smoke testing* atau *sanity testing*) pada integrasi antara komponen-komponen utama (*integration testing*), atau pada sistem secara keseluruhan (*system testing*).

2. Non-functional testing

Sementara berkebalikan dengan yang pertama, *non-functional testing* tidak dapat memeriksa “jika” *software* tersebut dapat melakukan tindakan tertentu, tetapi “bagaimana” melakukan tindakan itu.

3. Regression testing

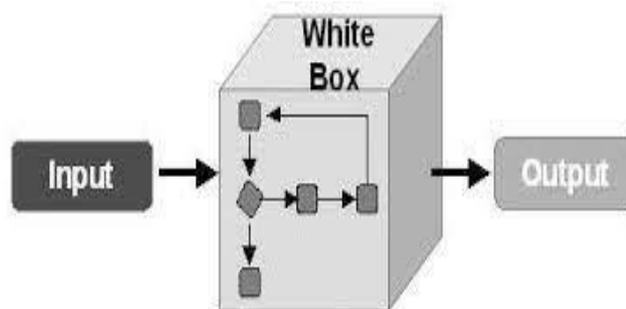
Fitur ini juga bisa digunakan memeriksa apakah versi terbaru dari *software* tersebut sudah menunjukkan regresi, atau penurunan kemampuan, dari satu versi ke versi berikutnya. *Regression testing* dapat diterapkan pada aspek fungsional *software* (misalnya, fitur tertentu tidak lagi berfungsi seperti yang diharapkan di versi terbaru), atau aspek *non-functional* (misalnya, memiliki kinerja yang sangat lambat di versi terbaru)

Black box testing atau dapat disebut juga *Behavioral Testing* adalah pengujian yang dilakukan untuk mengamati hasil input dan output dari perangkat lunak tanpa mengetahui struktur kode dari perangkat lunak. Pengujian ini dilakukan di akhir pembuatan perangkat lunak untuk mengetahui apakah perangkat lunak dapat berfungsi dengan baik. Untuk melakukan pengujian, penguji tidak harus memiliki kemampuan menulis kode program. Pengujian ini dapat dilakukan oleh siapa saja. Salah satu contoh pengujiannya ada pada tabel 2.8 berikut.

Tabel 2.1 Contoh Pengujian *Black Box*

No	Test Case	Hasil yang diharapkan	Hasil yang didapatkan	Keterangan
1.	Pengujian button submit	Sistem akan memproses form	Sistem menjalankan fungsi proses form	Berhasil
2.	Pengujian button cancel	Sistem akan keluar dari interface form dan kembali ke halaman utama modul yang bersangkutan	Sistem kembali ke halaman utama modul yang bersangkutan	Berhasil
3.	Pengujian button edit	Sistem akan masuk form edit data	Sistem masuk ke halaman yang menyajikan form edit data	Berhasil
4.	Pengujian button delete	Sistem akan menghapus	Sistem menghapus data yang dipilih	Berhasil

O. White Box Testing



(Gambar 2.12. White Box Testing)

Sumber : (raharja.ac.id,2020)

White Box Testing adalah salah satu cara untuk menguji suatu aplikasi atau perangkat lunak dengan melihat modul untuk memeriksa dan menganalisis kode program ada yang salah atau tidak. Jika modul ini dan telah diproduksi dalam

output yang tidak memenuhi persyaratan, kode akan dikompilasi ulang dan diperiksa lagi sampai Anda mencapai apa yang diharapkan pengujian yang didasarkan pada kontrol detail desain. Struktur kontrol desain program digunakan dengan cara prosedural untuk membagi tes ke dalam berbagai kasus uji. Sekilas dapat disimpulkan bahwa tes kotak putih adalah panduan untuk membuat program 100% benar (Akbar Asfihan.2019).

Istilah-istilah dalam pembuatan *flowgraph*

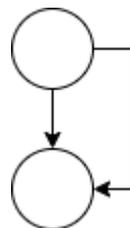
1. *Node* yaitu lingkaran pada *flowgraph* yang menggambarkan satu atau lebih perintah *prosedural*. Urutan proses dan keputusan dapat dipetakan dalam satu *node*.



2. *Edge* yaitu tanda panah yang menggambarkan aliran kontrol. Setiap *node* harus mempunyai tujuan *node*.



3. *Region* yaitu daerah yang dibatasi oleh *node* dan *edge*. Untuk menghitung region daerah diluar *flowgraph* juga harus dihitung.



4. *Predicate Node* yaitu kondisi yang terdapat pada node dan mempunyai karakteristik dua atau lebih *edge* darinya.



5. *Cyclomatic Complexity* yaitu metric perangkat lunak yang menyajikan ukuran kuantitatif dari kekompleksan logikal suatu program. *Cyclomatic Complexity* digunakan untuk mencari jumlah path dalam suatu *flowgraph*.
6. *Independent Path* yaitu jalur yang melintasi atau melalui dalam program dimana sekurang-kurangnya terdapat proses perintah yang baru atau kondisi yang baru. Dalam istilah *flowgraph*, independent path harus bergerak sekurang-kurangnya pada satu *edge*, dimana *edge* yang akan dilewati tidak dapat dilewati edge sebelum jalur tersebut didefinisikan.

Untuk menguji suatu sistem maka bagan alir program (*Flowchart*) yang didesain sebelumnya dipetakan ke dalam bentuk bagan alir kontrol (*flowgraph*) yang nantinya memudahkan untuk penentuan jumlah region, *Cyclomatic Complexity* (CC) dan Independent Path, jika jumlah region, *Cyclomatic Complexity* (CC) dan Independent Path sama besar maka sistem dinyatakan benar, tetapi jika sebaliknya maka sistem masih memiliki kesalahan, mungkin dari segi logika maupun dari sisi lainnya. *Cyclomatic Complelexity* (CC) dapat dihitung dengan menggunakan rumus

$$V(G) = E - N + 2$$

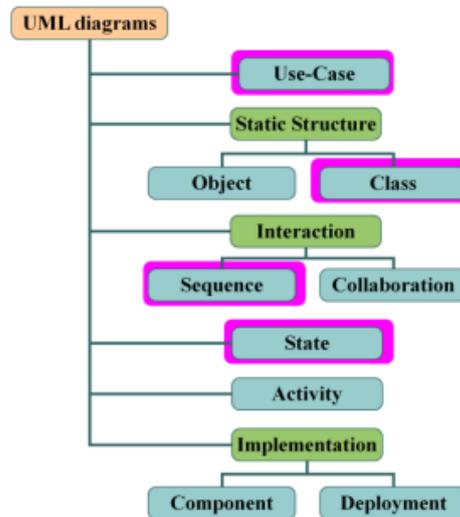
Dimana : E = Jumlah *edge* pada *flowgraph*

N = Jumlah *node* pada *flowgraph*

P. UML

Unified Modelling Language (UML) adalah suatu alat untuk memvisualisasikan dan mendokumentasikan hasil analisa dan desain yang berisi sintak dalam memodelkan sistem secara visual (Braun, et. al. 2001). Juga merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem software yang terkait dengan objek (Whitten, et. al. 2004). Sejarah UML sendiri terbagi dalam dua fase; sebelum dan sesudah munculnya UML. Dalam fase sebelum, UML sebenarnya sudah mulai diperkenalkan sejak tahun 1990an namun notasi yang dikembangkan oleh para ahli analisis dan desain berbeda-beda, sehingga dapat dikatakan belum memiliki standarisasi.

Saat ini sebagian besar para perancang sistem informasi dalam menggambarkan informasi dengan memanfaatkan UML diagram dengan tujuan utama untuk membantu tim proyek berkomunikasi, mengeksplorasi potensi desain, dan memvalidasi desain arsitektur perangkat lunak atau pembuat program. Secara filosofi UML diilhami oleh konsep yang telah ada yaitu konsep permodelan Object Oriented karena konsep ini menganalogikan sistem seperti kehidupan nyata yang didominasi oleh objek dan digambarkan atau dinotasikan dalam simbol yang cukup spesifik.



(Gambar 2.13. Diagram UML)

(Sumber : <http://www.uml.org/>).

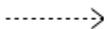
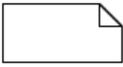
Dalam *UML* sendiri terdapat beberapa diagram yaitu :

1. *Use Case Diagram*

Use case diagram menggambarkan *fungsi* yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* mempresentasikan sebuah intraksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu, misalnya *login* ke sistem, *create* sebuah daftar belanja, dan sebagainya. Seorang/sebuah aktor adalah *entitas* manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu.

Tabel 2.1 Simbol *Use case Diagram*

No	Gambar	Nama	Keterangan
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .

No	Gambar	Nama	Keterangan
2		<i>Defendancy</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>)
3		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>ekspilisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>usecase</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemen
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

1. Class Diagram

Adapun simbol-simbol *Class Diagram* antara lain :

Tabel 2.2 Simbol *Class Diagram*

No	Gambar	Nama	Keterangan
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagai perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.

2. *Sequence Diagram*

Adapun simbol-simbol *Use case Diagram* antara lain :

Tabel 2.3 Simbol *Sequence Diagram*

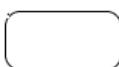
No	Gambar	Nama	Keterangan
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.

2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang yang terjadi.
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.

3. State Chart Diagram

Adapun simbol-simbol chart Diagram antara lain :

Tabel 2.4 Simbol *State Chart Diagram*

No	Gambar	Nama	Keterangan
1		<i>State</i>	Nilai atribut dan nilai link pada suatu waktu tertentu, yang dimiliki oleh suatu objek.
2		<i>Initial Pseudo State</i>	Bagaimana objek dibentuk atau diawali.
3		<i>Final State</i>	Bagaimana objek dibentuk dan dihancurkan.
4		<i>Transition</i>	Sebuah kejadian yang memicu sebuah state objek dengan cara memperbaharui satu atau lebih nilai atributnya.
5		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
6		<i>Node</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

4. Activiy Diagram

Adapun simbol-simbol *Activiy diagram* antara lain :

Tabel 2.5 Simbol *Activiy Diagram*

No	Gambar	Nama	Keterangan
1		<i>Actifity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.

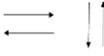
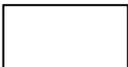
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi.
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang ada pada tahap tertentu berubah menjadi beberapa aliran.

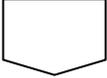
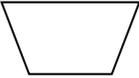
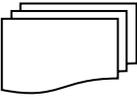
Q. Flowchart

(Rahmat Arifianto, 2014) *flowchart* adalah suatu bagan dengan simbol – simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu proses (instruksi) dengan proses lainnya suatu program.

Berikut ini adalah beberapa simbol yang digunakan dalam menggambar suatu *flowchart* :

Tabel 2.6 *Flowchart*

Simbol	Nama dan fungsi
	<i>Terminal point Symbol / simbol titik terminal</i> menunjukkan permulaan (<i>start</i>) atau akhir (<i>stop</i>) dari suatu proses.
	<i>Flow Direction Symbol / Simbol Arus</i> adalah simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain (<i>connecting line</i>). Simbol ini juga berfungsi untuk menunjukkan garis alir dari proses.
	<i>Processing Symbol / Simbol proses</i> digunakan untuk menunjukkan kegiatan yang dilakukan oleh komputer. Pada bidang industri (proses produksi barang), simbol ini menggambarkan kegiatan inspeksi atau yang biasa dikenal dengan simbol inspeksi.

	<p>Decision Symbol / simbol keputusan merupakan simbol yang digunakan untuk memilih proses atau keputusan berdasarkan kondisi yang ada. Simbol ini biasanya ditemui pada <i>flowchart program</i>.</p>
	<p>Input-output / Simbol keluar-Masuk menunjukkan proses input-output yang terjadi tanpa bergantung dari jenis peralatannya.</p>
	<p>Predefined Process / simbol proses Terdefinisi merupakan simbol yang digunakan untuk menunjukkan pelaksanaan suatu bagian prosedur (<i>sub-proses</i>). Dengan kata lain, prosedur yang terinformasi di sini detail dan akan dirinci di tempat lain.</p>
<p>Simbol</p>	<p>Nama dan fungsi</p>
	<p>Connector (on-page) Simbol ini fungsinya adalah untuk menyederhanakan hubungan antara simbol yang letaknya berjauhan atau rumit bila dihubungkan dengan garis dalam satu halaman</p>
	<p>Connector (off-page) sama seperti <i>on-page connector</i>, hanya saja simbol ini digunakan untuk menghubungkan simbol dalam halaman berbeda.</p>
	<p>Preparation Symbol / simbol persiapan merupakan simbol yang digunakan untuk mempersiapkan penyimpanan di dalam <i>storage</i>.</p>
	<p>Manual Input Symbol digunakan untuk menunjukkan input data secara manual menggunakan <i>online keyboard</i>.</p>
	<p>Manual Operation Symbol / simbol kegiatan manual digunakan untuk menunjukkan kegiatan/proses yang tidak dilakukan oleh komputer.</p>
	<p>Document Symbol jika anda menemukan simbol ini artinya input berasal dari dokumen dalam bentuk kertas, atau output yang perlu dicetak di atas kertas.</p>
	<p>Multiple Documents sama seperti <i>document symbol</i> hanya saja dokumen yang digunakan lebih dari satu dalam simbol ini.</p>
	<p>Display Symbol adalah simbol yang menyatakan penggunaan peralatan seperti monitor</p>



Delay Symbol sesuai dengan namanya digunakan untuk menunjukkan proses delay (menunggu) yang perlu dilakukan. Seperti menunggu surat untuk diarsipkan.

R. Kerangka Fikir

Mobil remote control menggunakan remot yang menggunakan sinyal analog yang diatur pada frekuensi tertentu. Pada remot biasanya memanfaatkan gelombang

radio frekuensi sebagai media penghubungnya. Permasalahannya ketika remot mobil tersebut rusak atau menggunakan Bluetooth sehingga jaraknya lebih terbatas. Maka akan lebih mudah ketika rc mobil dikontrol pada smartphone android karna kita hanya mengkonekkan IP ESP32Cam sehingga mudah di kontrol menggunakan smartphone android.



Dari permasalahan tersebut dibutuhkan suatu kontrol yang dapat mengendalikan mobil se cara jarak jauh menggunakan smartphone untuk mempermudah pengguna dalam mencari korban ditempat



Berdasarkan hal tersebut maka peneliti akan merancang sebuah alat menggunakan microcontroller esp32cam, modul l298n motor driver, dan motor dc.



Diharapkan dengna adanya alat ini dapat membantu pengguna dalam mencari korban di tempat kejadian dengan mudah tanpa kendala walaupun korban berada di reruntuhan

BAB III

METODE PENELITIAN

A. Lokasi dan Waktu Penelitian

1. Lokasi

Lokasi penelitian dilakukan di Laboratorium Informatika Fakultas Teknik kampus Universitas Muhammadiyah Parepare .

2. Waktu penelitian

Waktu yang dibutuhkan dalam pelaksanaan dalam penelitian ini adalah ± 2 bulan tahun 2023.

B. Jenis Penelitian

Untuk membantu kelancaran pengumpulan data, maka penulis menggunakan beberapa metode antara lain:

1. Penelitian lapangan

Yaitu penulis langsung melakukan penelitian pada objek. Data dan keterangan yang diperoleh dikumpulkan melalui pengamatan di lokasi penelitian. Dalam hal ini membutuhkan:

- a. Pengamatan (*Observation*), yaitu melakukan pengamatan secara langsung pada tempat yang terdapat reruntuhan.
- b. Dari hasil observasi dapat mengetahui masalah-masalah yang terjadi serta dilapangan yang terdapat reruntuhan.

- c. Kajian kepustakaan(*Literature Study*), merupakan pengumpulan data dengan cara mencari hasil penelitian sejenis yang menerapkan konsep *Internet of Things*(IoT) pada mekanisme kontrol.

C. Teknik Pengumpulan Data

Menganalisa data-data yang sebelumnya telah dikumpulkan dengan cara menganalisis cara kerja sistem yang akan dirancang, mengidentifikasi masalah, dan menganalisis kebutuhan sistem.

1. Perancangan program

Sebagai pedoman dalam penulisan program atau kode-kode agar berjalan sesuai rencana.

2. Uji coba program

Pengujian program dilakukan untuk memastikan bahwa program yang dibuat dapat berjalan dengan baik dengan cara membuat flowchart sistem kemudian akan digunakan untuk membuat desain sistem.

3. Evaluasi

Sistem yang telah selesai dibangun perlu adanya evaluasi untuk menentukan kelemahan yang terdapat pada program yang telah dibangun tadi, yang nantinya dapat digunakan sebagai acuan untuk memperbaiki program sehingga lebih sempurna.

D. Alat dan Bahan Penelitian

Dalam melakukan penelitian, penulis memerlukan alat dan bahan penelitian yang mendukung kegiatan penelitian tersebut antara lain:

1. Perangkat keras

Perangkat keras (*Hardware*) yang digunakan dalam pembuatan *prototype* sistem kontrol mobil control berkamera menggunakan esp32cam berbasis *internet of things* dapat dilihat pada tabel 3.1 berikut:

Tabel 3.1 Spesifikasi Perangkat Keras

No.	Spesifikasi	
1	Merk Leptop	Aser
2	Prosesor Leptop	Intel(R)Celeron(R) CPU 1000M @ 1.80GHz
3	RAM	2 GB
4	Mikrokontroler	Esp32cam
5	Jenis Alat	modul l298n motor driver, dan motor dc.
6	Perangkat Tambahan	<i>Breadboard, Kebel Jumper</i>

2. Perangkat Lunak

Perangkat lunak yang digunakan untuk merancang sistem kontrol dan monitoring *air conditioner* berbasis *internet of things* dapat dilihat pada tabel 3.2 berikut:

Tabel 3.2 Spesifikasi Perangkat Lunak

No.	Spesifikasi	
1	Sistem Operasi	Windows 7
2	Arduino IDE	1.8.16

3. Bahan

Adapun bahan yang digunakan selama penelitian yaitu:

1. *Esp32cam*
2. modul l298n motor driver

3. motor dc.
4. *Breadboard*
5. Kabel USB

E. Tahapan Penelitian

Tahap-tahap yang dilakukan penulis dalam perancangan sistem kontrol ini adalah sebagai berikut:

1. Pengumpulan data

Mengumpulkan data-data dan informasi untuk dijadikan acuan dalam membangun aplikasi yang dirancang.

2. Analisis sistem

Mengidentifikasi dan mengenali masalah yang ada, kemudian mencari alternatif-alternatif pemecahannya.

3. Desain sistem

Setelah masalah ditentukan dan analisis data sudah dilakukan maka perlu dilakukan pembangunan atau perancangan sistem tentang masalah yang sudah ditentukan.

4. Koding

Tahap penulisan kode program sistem.

5. Pengujian sistem

Setelah proses koding selesai, dilakukan pengujian sistem menggunakan metode *black box*.

6. Implementasi sistem

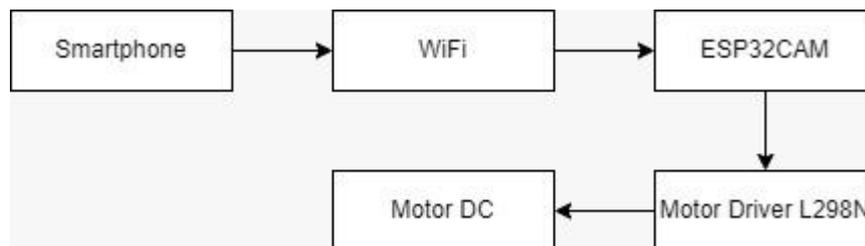
Tahap ini digunakan untuk penerapan dari pengujian sistem ke dalam kondisi sebenarnya agar dapat diketahui kelebihan dan kekurangannya.

F. Desain Sistem

Desain

Desain merupakan sebuah proses untuk mendefinisikan sesuatu yang akan di kerjakan dengan menggunakan teknik yang bervariasi serta di dalamnya melibatkan deskripsi mengenai arsitektur serta detail dan komponen dan juga keterbatasan yang akan di alami proses pengerjaannya.

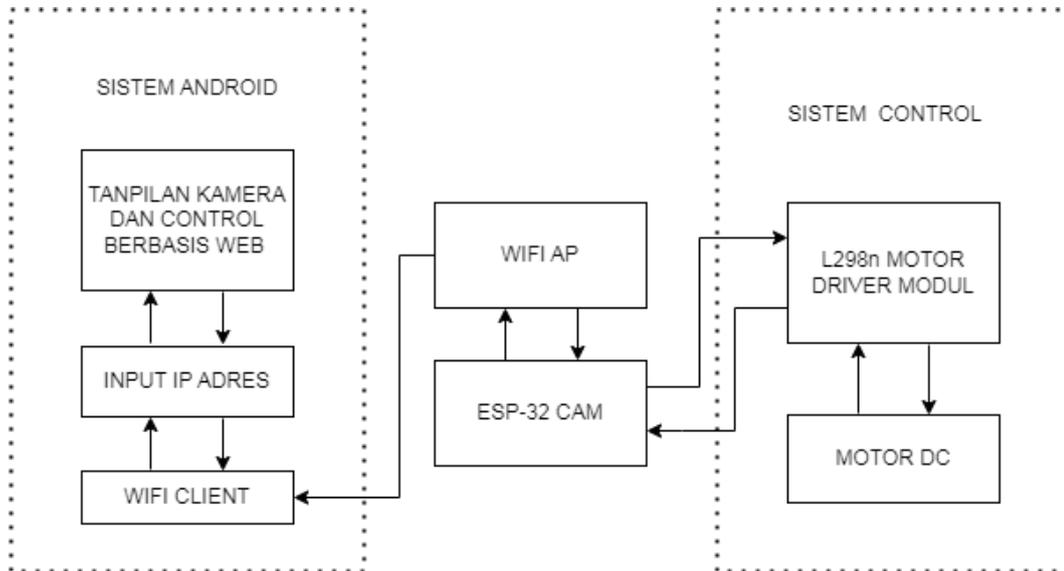
1. Balog Diagram



(Gambar 3.1. Diagram Balog)

Smartphone android yang berfungsi sebagai sistem remot control , dilanjutkan ke modul Wifi di sini sebagai alat penghubung komunikasi antara Smartphone android dengan ESP32Cam, kemudian perintah akan dilanjutkan pada ESP32Cam sebagai bagian utama untuk mengatur perintah input dan output,

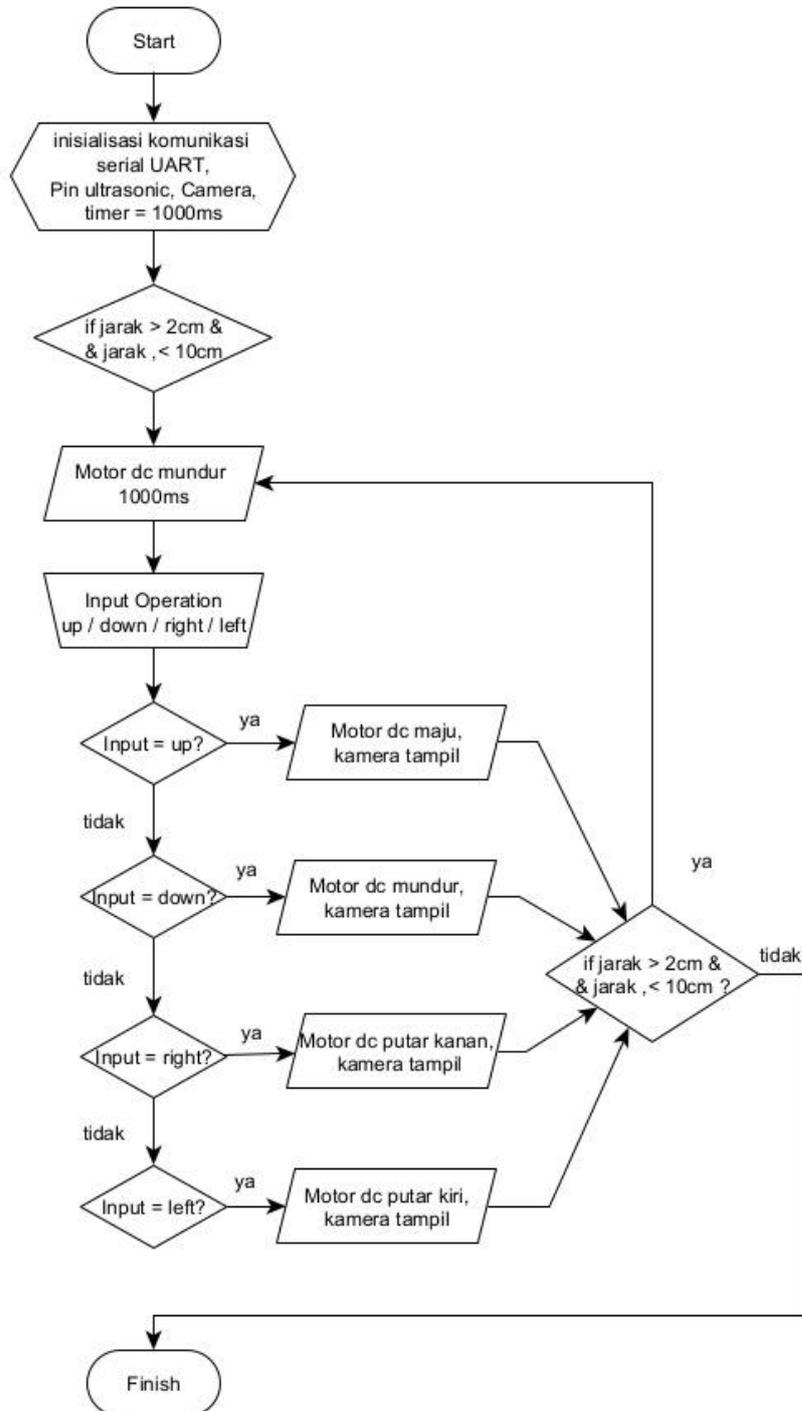
2. Rancangan Alat



(Gambar 3.2 Rancangan Alat)

Alat yang menjadi obyek penelitian adalah sebuah mobil Remote Control (RC) yang selama ini dikontrol melalui remote dengan gelombang radio, akan dikontrol menggunakan Android melalui wifi. Rangkaian elektronik akan digantikan dengan Arduino yang akan mengontrol gerakan motor DC. Kecepatan putaran dan arah putaran motor dikendalikan oleh android. Hubungan antara Android dengan mobil RC dirancang dengan skema peralatan seperti pada gambar di atas

3. Flowchart



(Gambar 3.3 Flowchart Aplikasi)

BAB IV

HASIL DAN PEMBAHASAN

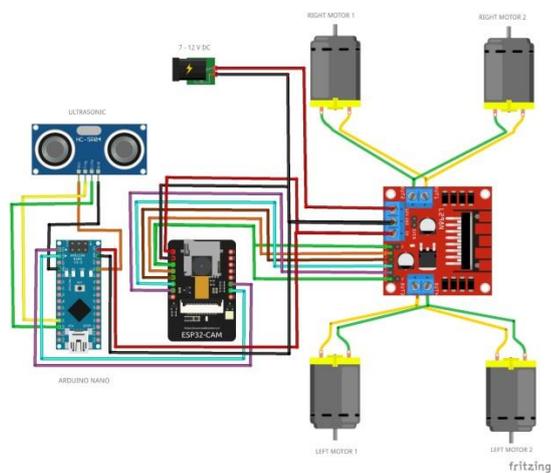
A. Rancangan Sistem

Alat yang dirancang pada penelitian ini terdiri dari 2 (dua) komponen utama, yaitu perangkat keras (*hardware*) dan perangkat lunak (*software*), untuk lebih jelasnya dapat dilihat sebagai berikut:

1. Perangkat Keras (Hardware)

Diagram pengkabelan

Sistem ini terdiri dari beberapa komponen dengan NodeMCU sebagai pengendali utama sistem. Frekuensi dihasilkan menggunakan PWM pada NodeMCU yang diteruskan ke sebuah speaker untuk menghasilkan gelombang ultrasonik, serta sebuah potensiometer yang berfungsi untuk mengatur nilai frekuensi.



Gambar 4.1 Diagram pengkabelan

2. Perangkat lunak (software)

Interface aplikasi

Aplikasi android ini bernama Mobil Remot Control, berfungsi menampilkan informasi yang diterima oleh ESP32Cam dan mampu bergerak kiri ke kanan atau maju mundur.



(Gambar 4.2 Tampilan aplikasi)

B. Pengujian Sistem

Pengujian sistem secara keseluruhan bertujuan untuk mengetahui berfungsi tidaknya alat. Adapun pengujian pada penelitian ini yaitu pengujian jangkauan esp32 cam dan pengujian frekuensi pada alat

1. Pengujian Jangkauan ESP32 cam

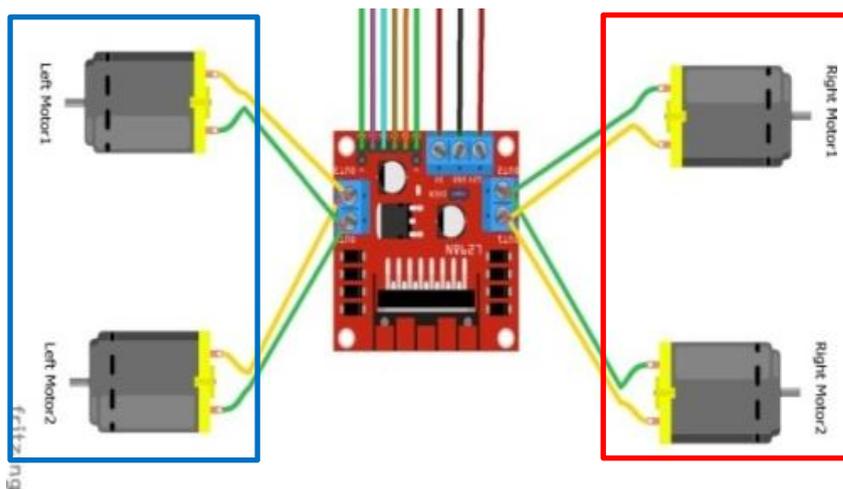
Tabel 4.1 tentang pengujian jangkauan ESP32 cam

Jarak Mobil	Respon Pada Alat	Waktu Pembacaan
1 meter	masih koneksi	1 Detik
2 meter	masih koneksi	1 Detik
3 meter	masih koneksi	1 Detik
4 meter	masih koneksi	1 Detik
5 meter	masih koneksi	1 Detik
6 meter	masih koneksi	1 Detik
7 meter	masih koneksi	1 Detik
8 meter	masih koneksi	1 Detik
9 meter	masih koneksi	1 Detik
10 meter	masih koneksi	1 Detik
11 meter	masih koneksi	1 Detik
12 meter	masih koneksi	1 Detik
13 meter	masih koneksi	1 Detik
14 meter	masih koneksi	1,5 Detik
15 meter	masih koneksi	1,5 Detik
16 meter	masih koneksi	1,5 Detik
19 meter	masih koneksi	1,5 Detik
20 meter	Koneksi terputus	0
21 meter	Koneksi terputus	0

Dari table di atas dapat disimpulkan bahwa mobil remote control ini dapat di kendalikan lancar dengan jarak maksimum 20-25 Meter, sedangkan di jarak 25-

30 meter koneksi sering terputus dan tidak lancar, sedangkan pada pengujian di atas 30 meter tidak dapat koneksi. Berdasarkan analisa penulis kemampuan maksimum modul wifi ini dapat bekerja dengan maksimal pada jarak 20 meter pada kondisi ruangan terbuka, tentunya ini juga tergantung pada sebuah wifi

2. Penjelasan kombinasi putaran roda



Gambar 4.2 Putaran Roda

a. Kombinasi roda belok kiri

kombinasi putaran roda pada saat belok kiri yaitu putaran roda sebelah kanan depan dan belakang (kotak berwarna merah) berputar maju, sedangkan pada roda sebelah kiri depan dan belakang berputar mundur (kotak berwarna biru)

b. Kombinasi roda belok kanan

kombinasi putaran roda pada saat belok kanan yaitu putaran roda sebelah kiri depan dan belakang (kotak berwarna biru) berputar maju, sedangkan pada roda sebelah kanan depan dan belakang berputar mundur (kotak berwarna merah)

c. Kombinasi roda maju

Kombinasi putaran roda pada saat maju yaitu putaran roda bagian depan dan belakang berputar maju

d. Kombinasi roda mundur

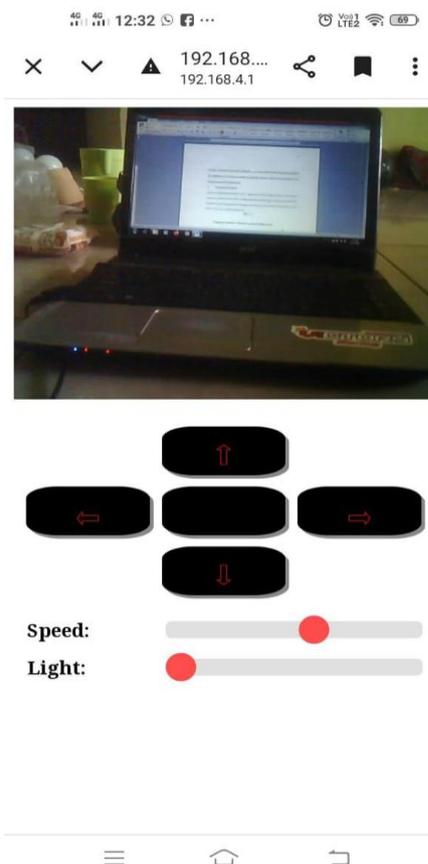
Kombinasi putaran roda pada saat mundur yaitu putaran roda bagian depan dan belakang berputar mundur

3. Pengujian aplikasi

Menguji keseluruhan memiliki tujuan guna melihat hasil kerja dari rangkian mobil rc dan aplikasi pada web browser secara keseluruhan. Perancangan sistem Mobil rc berbasis IoT dengan pengujian memberikan input sesuai fungsinya yaitu membantu tim relawan dalam mencari korban bencana alam gempa bumi dan menelusuri lorong yang tidak dapat di lalui oleh tim relawan dan sangat berbahaya jika di masuki paksa oleh tim relawan dengan jaringan Wifi Sebagai berikut langkah menguji robot mobil rc: 1. Hidupkan mobil rc melalui saklar pada robot. 2. Sambungkan ponsel ke dalam wifi ESP 32 Cam. 3. Masukkan IP dari ESP 32 Cam ke browser 4. Jalankan Mobil RC untuk menelusuri kolong dan tempat gelap. 5. Pada remote control yang berada di browser akan di tunjukan sebuah kontrol kanan, kiri, atas, bawah serta tombol on/off flash pada ESP 32 Cam 6. Gerak mobil diamati setiap perintah dari remote control. Dari hasil pengujian dilakukan prosedur diatas dapat disimpulkan Mobil RC berbasis IoT bekerja mengikuti perintah remote control serta berjalan pada tujuan yang sudah diharapkan.

4. Pengujian Kamera

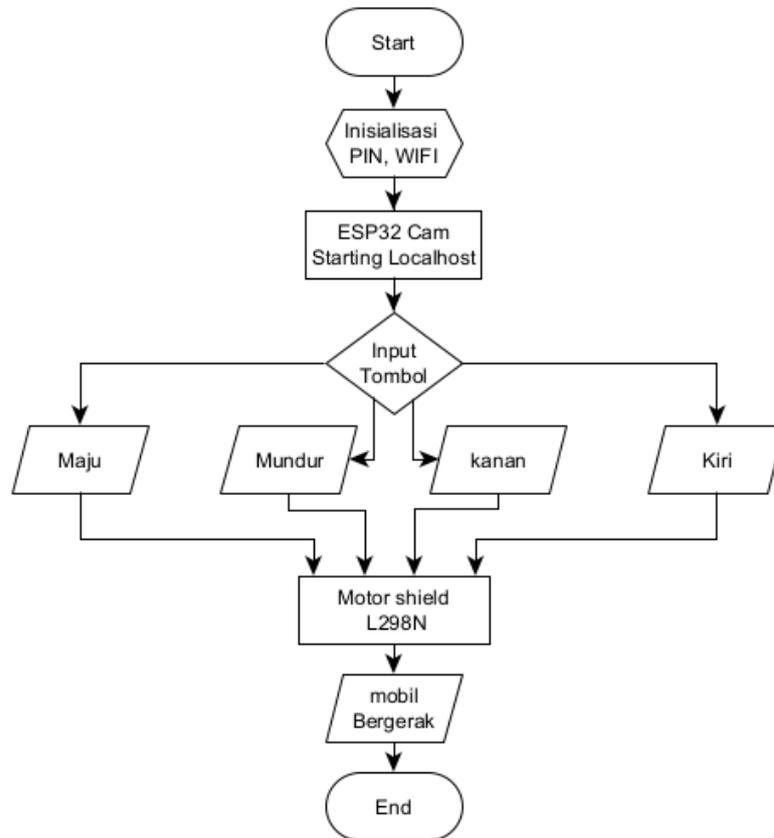
Untuk mengetahui kinerja ESP Cam 32 akan diuji terlebih dahulu dengan memasukan program coding kedalam ESP. Coding dibuat untuk ESP guna mengontrol Mobil RC. Algoritma pada program dibuat menggunakan bahasa java kemudian diupload ke board ESP 32 Cam dan melihat reaksinya.



(Gambar 4.3 Tampilan Kamera)

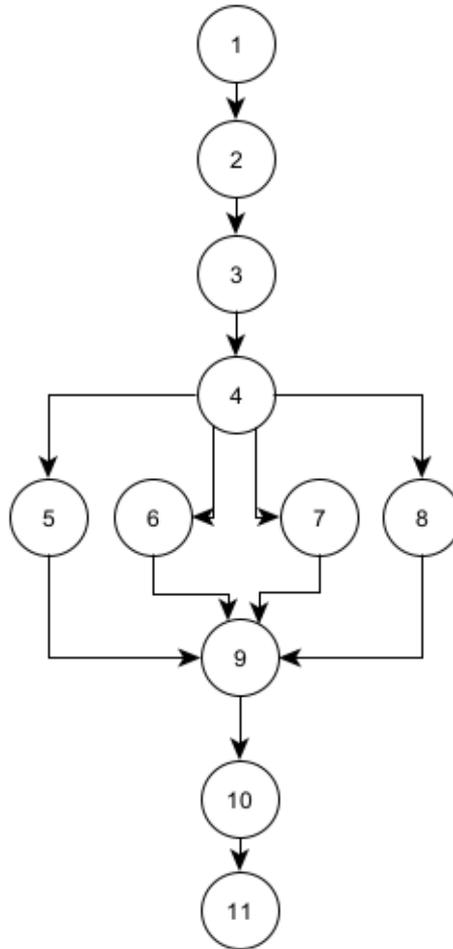
Pengujian whitebox dilakukan untuk melihat hasil

5. Whitebox Sistem



(Gambar 4.4 Whitebox Sistem)

1) Flowgraph Sistem



(Gambar 4.5 Flograf sistem)

1) Cyclomatic Complexity

Menghitung cyclomatic complexity $V(G)$ dari Edge dan Node:

Menggunakan rumus $V(G) = E - N + 2$

E (Edge) = 11

N (Node) = 10

$$\begin{aligned} \text{Penyelesaian: } V(G) &= E - N + 2 \\ &= 11 - 10 + 2 \\ &= 3 \end{aligned}$$

- 2) Berdasarkan perhitungan *cyclomatic complexity* dari *flowgraph* di atas mempunyai *region* = 3
- 3) *Independentpath* pada *flowgraph* di atas adalah:
- Path 1 = 1 – 2 – 3 – 4 – 5 – 9 – 10 – 11
- Path 2 = 1 – 2 – 3 – 4 – 6 – 9 – 10 – 11
- Path 3 = 1 – 2 – 3 – 4 – 7 – 9 – 10 – 11
- Path 4 = 1 – 2 – 3 – 4 – 8 – 9 – 10 – 11
- 4) Grafik matriks aktivasi *flowgraph*

Tabel 4.1 grafik matriks aktivasi *flowgraph* aplikasi

	1	2	3	4	5	6	7	8	9	10	11	E – 1
1		1										1 - 1 = 0
2			1									1 - 1 = 0
3				1								1 - 1 = 0
4					1	1	1	1				4 - 1 = 3
5									1			1 - 1 = 0
6									1			1 - 1 = 0
7									1			1 - 1 = 0
8									1			1 - 1 = 0
9										1		1 - 1 = 0
10											1	1 - 1 = 0
11												0
SUM (E) + 1												3 + 1 = 4

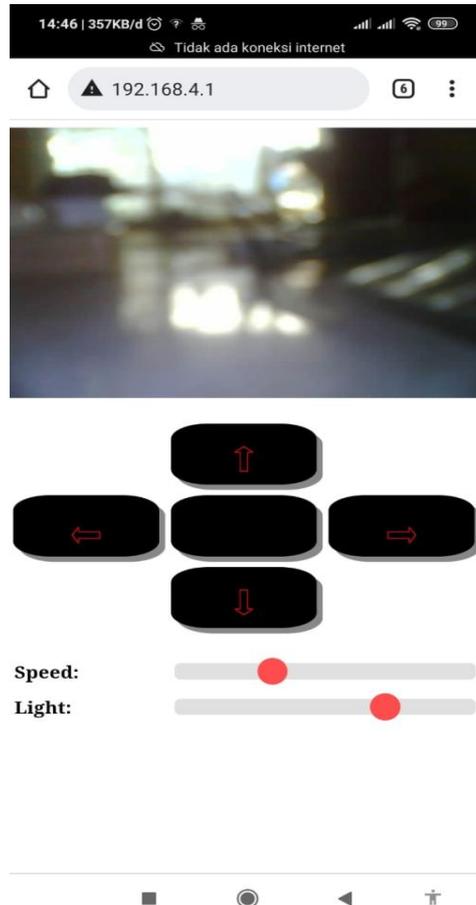
6. Pengujian Blackbox

a. Blackbox Aplikasi

Pengujian blackbox aplikasi bertujuan untuk mengetahui berfungsi atau tidaknya aplikasi yang telah dibuat. Pengujian dilakukan untuk melihat hasil deteksi yang dilakukan alat berdasarkan intruksi sensor yang digunakan pada sisi aplikasi.

1) Detail aplikasi

Halaman utama adalah tampilan aplikasi saat aplikasi dijalankan.

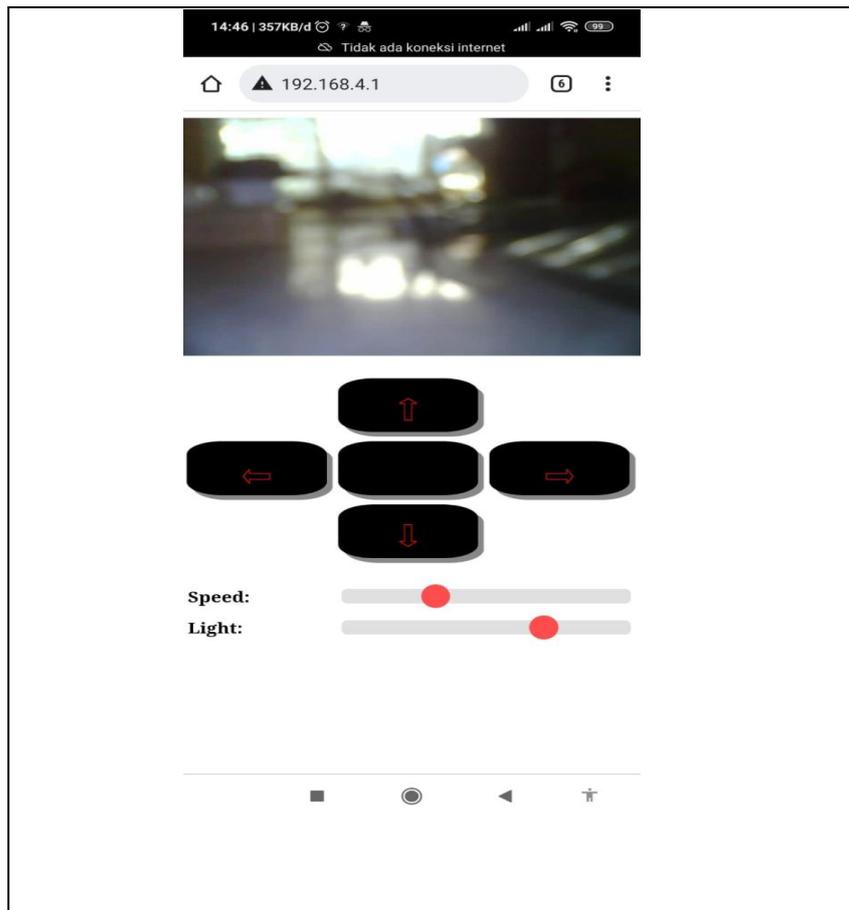


(Gambar 4.6 Halaman awal aplikasi)

- 2) Pengujian aplikasi Pengujian aplikasi dilakukan untuk mengetahui jalannya aplikasi,

Tabel 4.3 Blackbox tampilan awal

Text faktor	Hasil	Kesimpulan
Tampilan awal aplikasi	✓	Berhasil, Berhsil menampilkan aplikasi
Scranshot		



Tabel 4.4 Blackbox tampilan depan

Text faktor	Hasil	Kesimpulan
Tampilan depan aplikasi	✓	Berhasil, berhasil menampilkan tampilan depan alat
Foto		



Tabel 4.3 Blackbox tampilan belakang

Text faktor	Hasil	Kesimpulan
Tampilan mundur aplikasi	✓	Berhasil, Berhsil menampilkan alat saat mundur
Foto		



Tabel 4.3 Blackbox tampilan kiri

Text faktor	Hasil	Kesimpulan
Tampilan kiri aplikasi	✓	Berhasil, Berhasil menampilkan c belok kiri
Foto		



Tabel 4.5 Blackbox tampilan kanan

Text faktor	Hasil	Kesimpulan
Tampilan kanan aplikasi	✓	Berhasil, Berhasil menampilkan alat pada saat belok kanan
Foto		



Tabel 4.6 Blackbox tampilan lampu menyala

Text faktor	Hasil	Kesimpulan
Tampilan Depan	✓	Berhasil, Berhsil menampilkan pada saat lampu menyala
Foto		



Tabel 4.7 Blackbox tampilan Sensor Ultrasonic

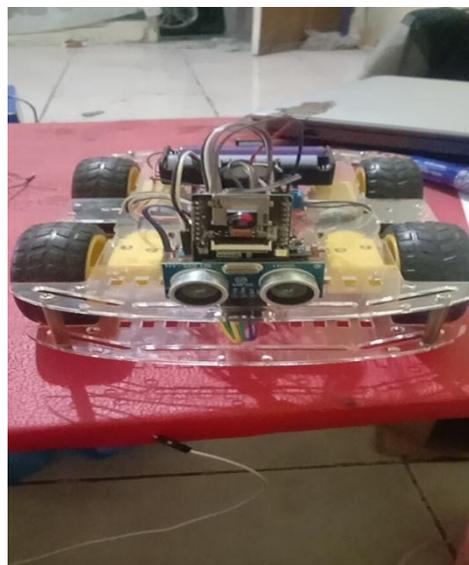
Text faktor	Hasil	Kesimpulan
Tampilan Sensor	✓	Berhasil, Berhsil menampilkan pada saat pembacaan Sensor Ultrasonic
Foto		



b. Blackbox Alat

Pengujian blackbox alat bertujuan untuk mengetahui berfungsi atau tidaknya alat yang telah dibuat. Pengujian dilakukan untuk melihat hasil atau jarak yang dilakukan alat berdasarkan kontrol yang digunakan.

1) Detail alat



(Gambar 4.7 Alat mobil remot control)

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan perancangan, pengamatan dan pengujian yang telah dilakukan maka dapat diambil kesimpulan sebagai berikut :

1. Telah berhasil dirancang sebuah sistem pengendali mobil remot control melalui android, dimana dapat mempermudah pengendalian mobil remot control.
2. Mobil remot control yang digunakan mampu menjadi penggerak baik untuk pergerakan kiri dan kanan maupun maju dan mundur.
3. ESP 32 Cam adalah sebuah perangkat pengendali yang sudah dilengkapi dengan camera sangat efisien jika digunakan dalam pembuatan rancang bangun.
4. Dengan dilengkapinya flash pada ESP 32 Cam mobil mampu bergerak dengan baik pada kondisi gelap.

B. Saran

Berdasarkan saran yang dapat diberikan dari hasil penelitian ini diantaranya

1. Disarankan untuk penelitian lebih lanjut mengenai mobil RC menggunakan roda tank.
2. Menambahkan sensor pendeteksi suhu manusia normal.

DAFTAR PUSTAKA

- Isrofi, A., Utama, S. N., & Putra, O. V. (2021). RANCANG BANGUN ROBOT PEMOTONG RUMPUT OTOMATIS MENGGUNAKAN WIRELESS KONTROLER MODUL ESP32-CAM BERBASIS INTERNET of THINGS (IoT). *Jurnal Teknoinfo*, 15(1), 45.
- Ramadan, D. N., Permana, A. G., & Hafidudin, H. (2017). Perancangan Dan Realisasi Mobil Remote Control Menggunakan Firebase. *Jurnal Elektro Dan Telekomunikasi Terapan*, 4(1), 505
- Suryadi, N., Marindani, E. D., W, F. T. P., Elektro, J. T., Teknik, F., & Tanjungpura, U. (2014). Rancang Bangun Robot Tank Yang Dilengkapi Kamera Terintegrasi Dengan Smartphone Android Via Wifi. 2(1), 1–8.
- Widiyanto, A., & Nuryanto, N. (2016). Rancang Bangun Mobil Remote Control Android dengan Arduino. *Creative Information Technology Journal*, 3(1), 50.
- Rosa Indriyani. (2014). PENGEMBANGAN ALAT PERAGA FISIKA MELALUI LIMBAH DAUR ULANG BERUPA MOBIL REMOTE CONTROL.
- Sugiharto, A. (2019). Rancang Bangun Robot Pengintai Dengan Kendali Androi. *Statistical Field Theor*, 53(9), 1689–1699.
- WATI, Ira; ACHMED, Sultan; AGUNG, Titus Y. Perancangan Mobil Remot Kontrol Arduino dengan Bluetooth Via Android. *Jurnal JE-UNISLA: Electronic Control, Telecommunication, Computer Information and Power System*, 2022, 7.1: 21-25.
- HARDI, Heri. PERANCANGAN PROTOTYPE MOBIL REMOTE CONTROL DENGAN SMARTPHONE ANDROID MENGGUNAKAN BLUETOOTH HC-05 BERBASIS ARDUINO UNO. *Jurnal Pengelolaan Laboratorium Sains dan Teknologi*, 2022, 2.1: 13-22.
- RAMADAN, Afwan; RUSDINAR, Angga; ROSA, Muhammad Ridho. Perancangan Kendali Kecepatan Mobil Listrik Dengan Metode Pid Berbasis Remot Kontrol. *eProceedings of Engineering*, 2022, 9.2.
- SINTARO, Sanriomi, et al. Implementasi Motor Driver Vnh2Sp30 Pada Mobil Remote Control Dengan Kendali Telepon Genggam Pintar. *Jurnal Teknologi dan Sistem Tertanam*, 2022, 3.1.
- SAPUTRA, Mohamad Wahyu. RANCANG BANGUN MOBIL REMOTE MENGGUNAKAN SENSOR ULTRASONIC BERBASIS ARDUINO NANO. 2022. PhD Thesis. Politeknik Negeri Sriwijaya.

YUSIKA, Andi; ROFIQ, Ahmad; RAMADHANI, Ade Tri. PERANCANGAN MOBIL REMOTE CONTROL MENGGUNAKAN ARDUINO UNO. Sebatik, 2019, 23.2: 541-546.