



TUGAS AKHIR - TE 141599

**RANCANG BANGUN KENDALI JARAK JAUH ROBOT SERVIS
PEMBERSIH DEBU BERBASIS INTERNET OF THINGS**

**ADRIE SENTOSA
NRP 2212 100 078**

**Dosen Pembimbing
Dr. Ir. Djoko Purwanto, M.Eng
Rudy Dikairono, ST., MT.**

**JURUSAN TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2016**



FINAL PROJECT - TE 141599

**DEVELOPMENT OF REMOTELY CONTROL FOR VACUUM CLEANER
SERVICE ROBOT BASED ON INTERNET OF THINGS**

ADRIE SENTOSA
NRP 2212 100 078

Supervisor
Dr. Ir. Djoko Purwanto, M.Eng
Rudy Dikairono, ST., MT.

DEPARTMENT OF ELECTRICAL ENGINEERING
Faculty of Industrial Technology
Sepuluh Nopember Institute of Technology
Surabaya 2016

**RANCANG BANGUN KENDALI JARAK JAUH ROBOT SERVIS
PEMBERSIH DEBU BERBASIS *INTERNET OF THINGS***

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik**

Pada

Bidang Studi Elektronika

Jurusan Teknik Elektro

Institut Teknologi Sepuluh Nopember

Menyetujui :

Dosen Pembimbing I,

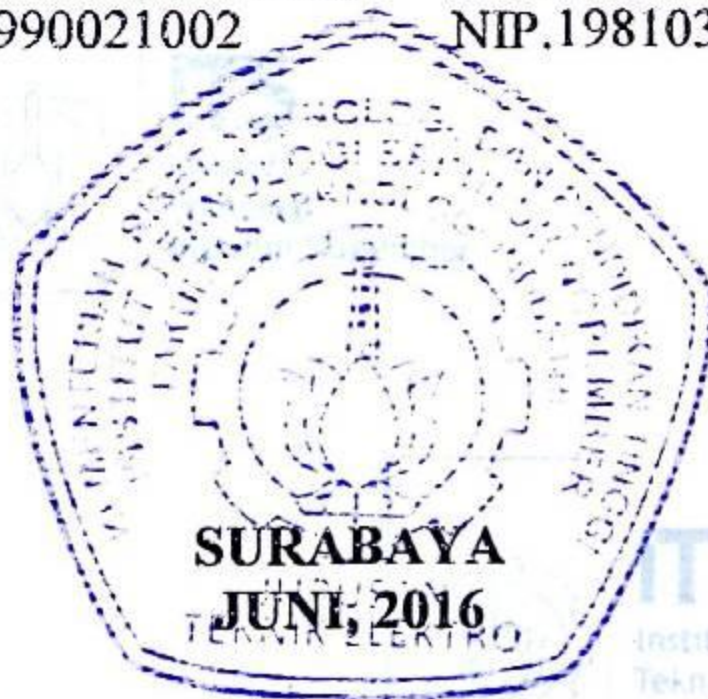


Dr. Ir. Djoko Purwanto, M. Eng
NIP.196512111990021002

Dosen Pembimbing II,



Rudy Wikairono, S.T., M.T.
NIP.198103252005011002



RANCANG BANGUN KENDALI JARAK JAUH ROBOT SERVIS PEMBERSIH DEBU BERBASIS *INTERNET OF THINGS*

Adrie Sentosa
2212100078

Dosen Pembimbing I : Dr. Ir. Djoko Purwanto, M.Eng
Dosen Pembimbing II : Rudy Dikairono, ST., MT.

Abstrak:

Robot servis otonom, khususnya robot pembersih debu otonom, yang melakukan pekerjaan secara mandiri ketika pengguna tidak berada di rumah merupakan impian sebagian besar masyarakat. Berbagai perusahaan dan institusi penelitian telah melakukan usaha yang baik dalam merancang robot servis pembersih debu otonom. Robot servis pembersih debu otonom yang dikembangkan saat ini, khususnya robot pembersih debu Chuwi iLife v5, dikendalikan dengan tombol ataupun remote control berbasis infrared. Hal ini menjadi permasalahan ketika pengguna tidak berada di lokasi robot servis sehingga pengguna tidak dapat memberikan perintah kepada robot servis secara langsung.

Maka dari itu, dirancanglah kendali jarak jauh robot servis berbasis *Internet of Things* yang memungkinkan robot servis untuk dikendalikan pada jarak jauh. Robot servis akan diintegrasikan dengan perangkat *smartphone* atau komputer berbasis internet untuk menggantikan fungsi *remote control* sehingga pengguna dapat melakukan perintah dimanapun mereka berada selama memiliki koneksi internet.

Hasil pengujian yang dilakukan pada pengujian tugas akhir ini adalah sistem dapat mengendalikan robot servis pembersih debu dengan tingkat kehandalan sebesar 100%. Dengan menggunakan spesifikasi sistem yang digunakan pada tugas akhir ini, robot servis dapat menjalankan seluruh perintah yang diberikan oleh pengguna.

Kata Kunci : Kontrol Jarak Jauh, Robot Servis Pembersih Debu, *Internet of Things*.

DEVELOPMENT OF REMOTELY CONTROL FOR VACUUM CLEANER SERVICE ROBOT BASED ON INTERNET OF THINGS

Adrie Sentosa
2212100078

Supervisor : Dr. Ir. Djoko Purwanto, M.Eng
Co-Supervisor : Rudy Dikairono, ST., MT.

Abstract:

Autonomous service robot, especially autonomous vacuum cleaner robot, that cleans users home while users are outdoors doing some activity is an old wish come true. Many companies and research institution have done well in developing autonomous vacuum cleaner service robot that controlled by button or infrared based remote control. The only lack of result with button or infrared based controller in vacuum cleaner service robot, especially Chuwi iLife v5, is the lack of application's range to control service robot. User should be in robot's location to give command with these kind of method.

Therefore it is designed a service robot based on internet of things as the solution to the problem that mention above. Service robot will be integrated with web-based device like android smartphone or computer to replace button or infrared-based remote control. With this kind of method, users could control service robot wherever they are through internet.

The results of tests performed in this project is the system could controlled vacuum cleaner service robot with the percentage of reliability is 100%. With system specification that used in this project, vacuum cleaner service robot could done all of command that given by user.

Keyword : Remotely Control, Vacuum Cleaner Service Robot, Internet of Things

DAFTAR ISI

ABSTRAK.....	ix
ABSTRACT.....	xi
KATA PENGANTAR	xiii
DAFTAR ISI	xiv
DAFTAR GAMBAR	xvii
DAFTAR TABEL	xix
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Perumusan Masalah	2
1.3. Tujuan Perancangan.....	2
1.4. Rumusan Masalah.....	2
1.5. Metodologi.....	2
1.6. Sistematika Penulisan	4
1.7. Relevansi	5
BAB II DASAR TEORI	7
2.1. Chuwi iLife V5	7
2.2. Sinyal Inframerah	8
2.3. IR333	9
2.4. Arduino Nano	10
2.5. <i>Internet of Things</i>	12
2.6. LAMP	13
2.6.1. Linux	13
2.6.2. Apache.....	13
2.6.3. My SQL.....	14
2.6.4. PHP.....	14
2.7. HTTP	14
2.8. Raspberry Pi.....	14
2.9. Modul Kamera Raspberry Pi	16
2.10. Komunikasi Serial	17
2.11. <i>Indoor Positioning System (IPS)</i>	17
BAB III PERANCANGAN SISTEM	19
3.1. Diagram Blok dan Flowchart Sistem	19
3.1.1. Diagram Blok	20
3.1.2. Flowchart.....	21

3.2.	Perancangan Perangkat Keras	22
3.2.1.	Robot Servis.....	22
3.2.2.	Arduino Nano	24
3.2.3.	Kamera.....	26
3.2.4.	Raspberry Pi.....	28
3.2.5.	Tempat Penyimpanan Debu	30
3.2.6.	Suplai Daya.....	34
3.2.7.	Monitoring Daya.....	36
3.3.	Perancangan Perangkat Lunak	36
3.3.1.	Perancangan Kendali Sinyal Inframerah.....	37
3.3.2.	Perancangan Web Server	38
3.3.3.	Pengambilan Citra.....	41
3.3.4.	Perancangan <i>User-Guided Control</i>	42
3.3.5.	Perancangan Antarmuka <i>Webpage</i>	43
3.3.6.	Port Forwarding	45
BAB IV	PENGUJIAN.....	47
4.1.	Pengujian Kendali Jarak Jauh	47
4.2.	Pengujian Responsivitas Kendali Jarak Jauh	48
4.3.	Pengujian <i>User-Guided Control</i>	49
4.3.1.	Pengujian <i>View-Left</i>	49
4.3.2.	Pengujian <i>Keep Forward</i>	50
4.3.3.	Pengujian <i>View-Right</i>	52
4.4.	Pengujian Antarmuka <i>Webpage</i>	53
4.4.1.	Pengujian pada Komputer.....	53
4.4.2.	Pengujian pada Smartphone.....	56
BAB V	PENUTUP.....	61
5.1.	Kesimpulan	61
5.2.	Saran	61
DAFTAR PUSTAKA		63
LAMPIRAN.....		65
BIODATA PENULIS.....		77

DAFTAR GAMBAR

Gambar 2.1	Robot pembersih debu Chuwi iLife V5	7
Gambar 2.2	Ilustrasi modulasi cahaya inframerah	9
Gambar 2.3	Ilustrasi komunikasi menggunakan media sinyal inframerah	9
Gambar 2.4	IR333	10
Gambar 2.5	Arduino Nano <i>Rear</i>	11
Gambar 2.6	Ilustrasi cara kerja <i>Internet of Things</i>	12
Gambar 2.7	Logo LAMP.....	13
Gambar 2.8	Raspberry Pi 2.....	15
Gambar 2.9	Modul kamera Raspberry Pi	16
Gambar 2.10	Ilustrasi cara kerja komunikasi serial.....	17
Gambar 2.11	Ilustrasi penggunaan RSSI pada IPS.....	18
Gambar 3.1	Ilustrasi kendali jarak jauh robot servis	19
Gambar 3.2	Diagram Blok Sistem.....	20
Gambar 3.3	Flowchart Sistem	21
Gambar 3.4	Foto Robot Servis	23
Gambar 3.5	Diagram Robot Servis.....	23
Gambar 3.6	Foto Arduino Nano	25
Gambar 3.7	Diagram arduino nano pada robot servis	25
Gambar 3.8	Foto instalasi modul kamera pada robot servis.....	27
Gambar 3.9	Diagram modul kamera pada robot servis	27
Gambar 3.10	Foto instalasi raspberry pi pada robot servis.....	29
Gambar 3.11	Diagram raspberry pi pada robot servis	29
Gambar 3.12	Foto tampak atas tempat penyimpanan debu	31
Gambar 3.13	Foto tampak bawah tempat penyimpanan debu	31
Gambar 3.14	Foto sebelum instalasi tempat penyimpanan debu.....	32
Gambar 3.15	Foto sesudah instalasi tempat penyimpanan debu	32
Gambar 3.16	Diagram tempat penyimpanan debu	33
Gambar 3.17	Foto instalasi suplai daya.....	34
Gambar 3.18	Diagram suplai daya pada robot servis	35
Gambar 3.19	Rangkaian monitoring daya	36
Gambar 3.20	Flowchart sistem kendali sinyal inframerah	37
Gambar 3.21	Flowchart web-server	39
Gambar 3.22	Diagram sekuensial client/server	40

Gambar 3.23	Frame Citra.....	42
Gambar 3.24	Rancangan antarmuka web-page	44
Gambar 3.25	Antarmuka web-page pada komputer	44
Gambar 3.26	Antarmuka web-page pada smartphone.....	45
Gambar 3.27	Antarmuka port forwarding pada router TP-Link.....	45
Gambar 4.1	Grafik hasil pengujian responsivitas kendali jarak jauh	49
Gambar 4.2	Tampilan antarmuka <i>web-page</i> pada Internet Explorer di komputer	54
Gambar 4.3	Tampilan antarmuka <i>web-page</i> pada Google Chrome di komputer	54
Gambar 4.4	Tampilan antarmuka <i>web-page</i> pada Mozilla Firefox di komputer	55
Gambar 4.5	Tampilan antarmuka <i>web-page</i> pada Opera di komputer	55
Gambar 4.6	Tampilan antarmuka <i>web-page</i> pada Default Browser di <i>smartphone</i>	57
Gambar 4.7	Tampilan antarmuka <i>web-page</i> pada Mozilla Firefox di <i>smartphone</i>	57
Gambar 4.8	Tampilan antarmuka <i>web-page</i> pada Google Chrome di <i>smartphone</i>	58
Gambar 4.9	Tampilan antarmuka <i>web-page</i> pada Opera di <i>smartphone</i>	58
Gambar 4.10	Tampilan antarmuka <i>web-page</i> pada UC Browser di <i>smartphone</i>	59
Gambar 4.11	Tampilan antarmuka <i>web-page</i> pada CM Browser di <i>smartphone</i>	59

DAFTAR TABEL

Tabel 2.1 Spesifikasi Chuwi iLife V5	7
Tabel 2.2 Metode Kerja Chuwi iLife V5	8
Tabel 2.3 Spesifikasi IR333	10
Tabel 2.4 Spesifikasi Arduino Nano.....	11
Tabel 2.5 Spesifikasi Raspberry Pi 2.....	15
Tabel 2.6 Spesifikasi modul kamera Raspberry Pi V1	16
Tabel 3.1 Kode sinyal inframerah	38
Tabel 3.2 Daftar perintah <i>user-guided control</i>	42
Tabel 4.1 Tabel pengujian kendali jarak jauh.....	47
Tabel 4.2 Tabel pengujian responsivitas kendali jarak jauh.....	48
Tabel 4.3 Tabel pengujian kendali <i>view-left</i>	50
Tabel 4.4 Tabel pengujian kendali <i>keep forward</i>	51
Tabel 4.5 Tabel pengujian kendali <i>view-right</i>	52
Tabel 4.6 Tabel kompatibilitas antarmuka <i>web-page</i> pada berbagai <i>browser</i> di komputer	53
Tabel 4.7 Tabel kompatibilitas antarmuka <i>web-page</i> pada berbagai <i>browser</i> di <i>smartphone</i>	56

BAB I

PENDAHULUAN

1.1. Latar Belakang

Dalam bidang pelayanan domestik, penggunaan tenaga manusia untuk melakukan tugas yang membutuhkan waktu dalam proses penyelesaiannya dan bersifat repetitif perlahan dikurangi dan digantikan dengan sistem yang dapat berjalan secara mandiri. Contoh dari hal yang membutuhkan waktu dalam proses penyelesaiannya dan bersifat repetitif adalah pekerjaan rumah tangga.

Baru – baru ini penggunaan robot servis otonom, khususnya robot pembersih debu otonom, sangat diminati karena mempermudah pengguna dalam menyelesaikan pekerjaan rumah tangga. Robot servis jenis ini juga dapat mencegah dan mengurangi kesalahan dalam pengerjaan pekerjaan rumah tangga akibat terjadinya *human error* yang disebabkan oleh tingkat akumulasi kelelahan pada pengguna.

Seiring berkembangnya teknologi robotika, penambahan fungsi kendali baik secara langsung maupun tidak langsung pada peralatan elektronika merupakan salah satu faktor yang penting. Fungsi sistem kendali pada aplikasi elektronika, khususnya robot servis, memudahkan pengguna untuk memberikan perintah spesifik kepada robot. Salah satu contoh dari penggunaan fungsi kendali ini adalah penggunaan tombol dan *remote-control* pada peralatan elektronika. Namun, penggunaan fungsi kendali yang sekarang ada di pasaran memiliki keterbatasan dalam jarak penggunaannya. Penggunaan tombol sebagai fungsi kendali mengharuskan pengguna untuk berada di lokasi robot servis, sedangkan penggunaan *remote-control* berbasis infra merah mengharuskan pengguna untuk berada di sekitar lokasi robot servis dengan jarak kurang dari 10m untuk mengendalikan robot.

Maka dari itu, dirancang dan dibangun sebuah sistem pengendalian jarak jauh robot servis berbasis *internet of things* (IoT). Dengan sistem pengendali jarak jauh berbasis *internet of things*, diharapkan dapat memudahkan pengguna dalam mengendalikan robot servis, dimana

pengguna dapat mengendalikan robot servis dimana saja mereka berada selama pengguna memiliki akses pada jaringan internet.

1.2. Perumusan Masalah

Permasalahan yang dibahas dalam tugas akhir ini adalah:

1. Bagaimana mengendalikan robot servis pembersih debu melalui internet?
2. Bagaimana memonitoring kondisi lingkungan disekitar robot servis pembersih debu?
3. Bagaimana mengakses sistem kendali robot servis pemberih debu?

1.3. Tujuan Perancangan

Tujuan dari tugas akhir ini adalah sebagai berikut:

1. Pengguna dapat mengendalikan robot servis pembersih debu melalui internet.
2. Pengguna dapat memonitoring kondisi lingkungan disekitar robot servis pembersih debu.
3. Pengguna dapat mengakses sistem kendali robot servis pembersih debu dengan mudah dan nyaman.

1.4. Batasan Masalah

Batasan masalah dari tugas akhir ini adalah sebagai berikut:

1. Robot servis yang digunakan adalah robot pembersih debu otonom Chuwi iLife V5.
2. Komponen pemrosesan yang digunakan adalah *System on Chip* (SoC) Raspberry Pi 2 dan mikrokontroler Arduino Nano.
3. Aktuator yang digunakan adalah LED IR333.
4. Kendali jarak jauh menggunakan jaringan internet tanpa proxy.

1.5. Metodologi

Langkah-langkah yang dikerjakan pada tugas akhir ini adalah sebagai berikut :

1. Studi Literatur

Pada tahap ini dilakukan pengumpulan dasar teori yang

menunjang dalam penulisan Tugas Akhir. Dasar teori ini dapat diambil dari buku-buku, jurnal, dan artikel-artikel di internet dan forum-forum diskusi internet.

2. Perancangan Perangkat Keras

Perancangan hardware pada tahap ini meliputi perancangan kamera, unit pemrosesan berupa arduino dan raspberry pi, tempat penyimpanan debu, dan *power supply*. Kamera, *transmitter* sinyal inframerah dan *power supply* akan dihubungkan dengan unit processing supaya dapat diolah oleh perangkat lunak.

3. Perancangan Perangkat Lunak

Perangkat lunak dirancang dengan pembuatan *source code* untuk melakukan interaksi dengan robot servis, sistem kendali berbasis *internet of things* dan antarmuka *webpage*. Untuk interaksi dengan robot servis, dilakukan dengan menganalisa sistem komunikasi pada robot servis. Lalu dilakukan *encoding* agar robot servis dapat dikendalikan sesuai perintah yang diberikan. Untuk sistem kendali berbasis *internet of things*, sistem kendali dilakukan oleh operator secara manual melalui peralatan berbasis jaringan internet seperti *smartphone* dan komputer. Untuk antarmuka *webpage*, diperlukan *source code* yang digunakan sebagai interaksi dengan pengguna berupa tombol perintah dan tampilan gambar melalui kamera.

4. Pengujian Sistem

Pengujian alat dilakukan untuk menentukan keandalan dari sistem yang telah dirancang. Pengujian dilakukan untuk melihat apakah software dan hardware dapat bekerja secara baik dan sesuai dengan tujuan perancangan sistem.

Pengujian sistem dilakukan dengan mengendalikan robot servis melalui peralatan berbasis jaringan internet seperti *smartphone* atau komputer. Apakah robot servis dapat mengikuti perintah yang diberikan secara tepat atau tidak.

5. Analisa

Analisa yang dilakukan terhadap hasil dari pengujian bertujuan untuk menentukan karakteristik kendali robot servis pembersih debu baik dari sisi software maupun hardware yang telah dibuat.

Apabila karakteristik kendali robot servis dari software dan hardware yang telah dibuat masih belum sesuai, maka perlu dilakukan perancangan ulang pada sistem dan diuji kembali.

6. Penulisan Laporan Tugas Akhir

Tahap penulisan laporan tugas akhir adalah tahapan terakhir dari proses pengerjaan tugas akhir ini. Laporan tugas akhir berisi seluruh hal yang berkaitan dengan tugas akhir yang telah dikerjakan yaitu meliputi pendahuluan, tinjauan pustaka, pengujian, dan penutup.

1.6. Sistematika Penulisan

Dalam buku tugas akhir ini, pembahasan mengenai sistem yang dibuat terbagi menjadi lima bab dengan sistematika penulisan sebagai berikut:

- **Bab 1 : PENDAHULUAN**
Bab ini meliputi latar belakang, perumusan masalah, tujuan penelitian, batasan masalah, metodologi, sistematika penulisan, dan relevansi.
- **Bab 2 : TINJAUAN PUSTAKA**
Bab ini menjelaskan tentang *literature* yang dibutuhkan dalam pengerjaan tugas akhir ini, yang meliputi teori dasar tentang sinyal inframerah, *internet of things*, dan antarmuka *web-browser*.
- **Bab 3: PERANCANGAN SISTEM**
Bab ini menjelaskan tentang perencanaan pernagkat keras, perangkat lunak, dan integrasi dari beberapa sistem.
- **Bab 4 : PENGUJIAN**
Bab ini menjelaskan tentang hasil dari pengujian yang dilakukan terhadap sistem secara keseluruhan
- **Bab 5 : PENUTUP**
Bab ini berisi kesimpulan yang meliputi kelebihan maupun kekurangan pada alat yang telah dirancang dan saran untuk pengembangan alat selanjutnya

1.7. Relevansi

Hasil dari tugas akhir ini diharapkan dapat memberikan manfaat sebagai berikut:

1. Memudahkan pengguna dalam mengendalikan peralatan elektronika, khususnya robot pembersih debu otonom.
2. Sebagai dasar penelitian lebih lanjut, sehingga sistem kendali jarak jauh berbasis *internet of things* dapat berkembang menjadi lebih baik.

Halaman ini sengaja dikosongkan

BAB II

TINJAUAN PUSTAKA

Bab ini akan meninjau pustaka yang berhubungan dengan keseluruhan sistem yang akan dibuat pada tugas akhir ini. Tinjauan pustaka tersebut meliputi pustaka tentang robot servis Chuwi iLife v5, sinyal inframerah, IR333, *internet of things*, LAMP, adruino nano, raspberry pi, modul kamera raspberry pi, komunikasi serial, dan IPS.

2.1. Robot Servis Pembersih Debu Chuwi iLife V5

Robot servis adalah robot yang bertugas untuk membantu pekerjaan manusia yang bersifat berbahaya atau repetitif. Robot servis pada umumnya bekerja dengan sistem autonomus dengan opsi pengendalian manual. Salah satu robot servis adalah robot servis pembersih debu yang dapat dilihat pada Gambar 2.1.

Robot Servis Chuwi iLife V5 adalah sebuah robot servis berbasis ARM8 yang berfungsi sebagai pembersih debu. Robot servis ini berputar mengelilingi ruangan *indoor* dengan algoritma tertentu dan menghisap masuk debu di ruangan ke dalam tempat penyimpanan debu. Tabel spesifikasi robot servis Chuwi iLife V5 dapat dilihat pada Tabel 2.1.



Gambar 2.1 Robot Pembersih Debu Chuwi iLife V5 [1]

Tabel 2.1 Spesifikasi Chuwi iLife V5 [2]

Dimensi (Diameter, Tinggi, dan Berat)	Diameter :300mm; Tinggi :75mm; Berat :1.9kg.
Kapasitas Penyimpanan	0.3L
Daya Hisap	850 Pa
Tegangan Masuk / Kerja	11V 0.3A / 05V 0.5A
Kapasitas Baterai	2600mAh
Sensor	10 sensor jarak, 2 sensor tekanan, 3 sensor ketinggian, dan 5 penerima sinyal inframerah

Robot servis Chuwi iLife V5 yang memiliki jangkauan kerja hingga 150m² ini, memiliki beberapa metode kerja yang dikendalikan melalui *remote-control* berbasis inframerah. Metode kerja dari robot servis ini meliputi metode pembersihan manual, pembersihan otomatis, pembersihan area, pembersihan tepi, dan *home*. Metode kerja pada robot servis pembersih debu Chuwi iLife dapat dilihat pada Tabel 2.2 [2].

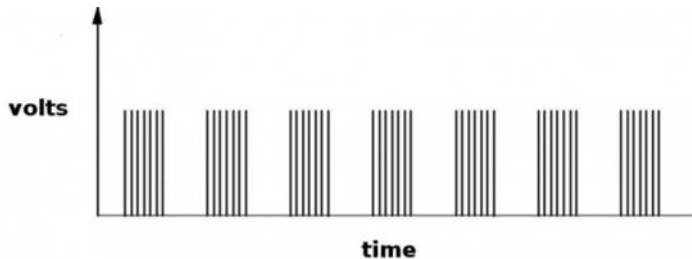
Tabel 2.2 Metode Kerja Chuwi iLife V5

Pergerakan Manual	Robot servis bergerak sesuai perintah pengguna (maju, mundur, kanan, kiri), dimana : <ul style="list-style-type: none">– Maju: bergerak maju sejauh 60cm– Mundur: bergerak mundur sejauh 5cm– Kanan: berputar ke kanan sebesar 22°– Kiri: berputar ke kiri sebesar 22°
Pembersihan Otomatis	Robot servis membersihkan ruangan secara kontinu dengan pergerakan acak
Pembersihan Area (<i>Spot-Clean</i>)	Robot servis memberishkan area (<i>spot</i>) tertentu secara kontinu
Pembersihan Tepi (<i>Edge-Clean</i>)	Robot servis membersihkan ruangan secara kontinu dengan acuan tepi ruangan
<i>Home</i>	Robot servis kembali ke lokasi pengisian baterai (<i>dock station</i>)

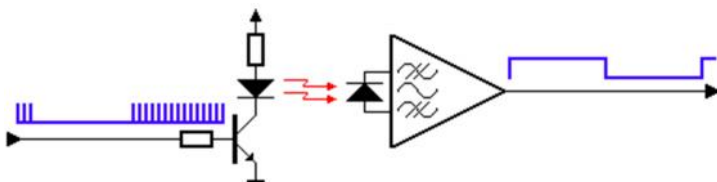
2.2. Sinyal Inframerah

Sinyal inframerah adalah penggunaan cahaya inframerah , radiasi elektromagnetik dari panjang gelombang antara cahaya tampak dan radiasi gelombang radio, sebagai media komunikasi jarak dekat. Penggunaan sinyal inframerah sebagai media komunikasi dapat ditemukan pada pengendali jarak jauh televisi, speaker, maupun aplikasi elektronika lainnya.

Sinyal inframerah bekerja dengan memodulasi cahaya inframerah pada frekuensi tertentu. Cahaya inframerah dimodulasi dengan cara mengubah besaran gelombang periodik baik sebagian atau seluruhnya nyala atau redup. Sinyal inframerah umumnya dipancarkan dalam modulasi 38kHz, dimana fraksi sinyal inframerah dipancarkan sebanyak 38.000 kali setiap detik. Ilustrasi modulasi cahaya inframerah oleh pengirim sinyal inframerah dapat dilihat pada Gambar 2.2. Sinyal inframerah yang telah diterima oleh receiver, kemudian diolah menjadi gelombang biner dengan mengendalikan jarak antara sinyal inframerah sehingga dapat diolah oleh mikrokontroller. Ilustrasi komunikasi menggunakan media sinyal inframerah dapat dilihat pada Gambar 2.3.



Gambar 2.2 Ilustrasi modulasi cahaya inframerah [3]



Gambar 2.3 Ilustrasi komunikasi menggunakan media sinyal inframerah [3]

2.3. IR333

IR333 adalah sebuah *emitter*, sumber energi cahaya, sinyal inframerah. *Emitter* sinyal inframerah menghasilkan sinyal inframerah yang mentransmisikan informasi dan perintah dari suatu alat elektronika ke peralatan elektronika lainnya. Emitter sinyal inframerah umumnya digunakan dalam jumlah besar untuk menghasilkan intensitas sinyal dan sudut pandang yang lebih baik.

Ada berbagai tipe *emitter* sinyal inframerah berdasarkan fungsi dan spesifikasinya seperti IR323, HIR204, QEE113, QEC112, OP280KT, OP180, dan TSAL6400. Setiap jenis *emitter* sinyal inframerah memiliki fungsi dan aplikasi masing-masing. Pada tugas akhir ini *emitter* sinyal inframerah yang digunakan adalah IR333. Tabel spesifikasi *emitter* sinyal inframerah IR333 dapat dilihat pada Tabel 2.3.



Gambar 2.4 IR333

Tabel 2.3 Spesifikasi IR333 [4]

Material	GaAlAs
Warna Lensa	Biru
Panjang Gelombang	940nm
Sudut Pandang	40°
Tegangan Kerja	2.6V
Arus Kerja	100mA

2.4. Arduino Nano

Arduino Nano adalah sebuah mini board berbasis mikro-kontroler ATmega328. Arduino nano memiliki 14 pin digital input/output (pin D0-D13) yang terdiri dari 8 pin analog input (pin A0-A7), 6 pin digital output (pin 3, 5, 6, 9, 10, 11) yang dapat digunakan untuk menghasilkan sinyal PWM (*Pulse Width Modulation*), sebuah kristal osilator 16 MHz, sebuah koneksi Mini-B USB, sebuah ICSP header, dan sebuah tombol reset. Arduino nano dapat dioperasikan dengan menggunakan port USB computer atau USB charger dengan tegangan yang direkomendasikan sebesar 5 Volt. Gambar arduino nano dapat dilihat pada Gambar 2.5 [5].

Ada beberapa tipe Arduino berdasarkan fungsi dan spesifikasinya seperti Arduino Uno, Duemilanove, Diecimila, NG Rev. C, Nuova Generazione, USB, Arduino Serial, Mega, FIO, Lilypad, BT, dan Arduino Mini. Arduino yang digunakan pada tugas akhir ini adalah Arduino Nano. Tabel spesifikasi mikrokontroler arduino nano dapat dilihat pada Tabel 2.4.



Gambar 2.5 Arduino Nano Rear

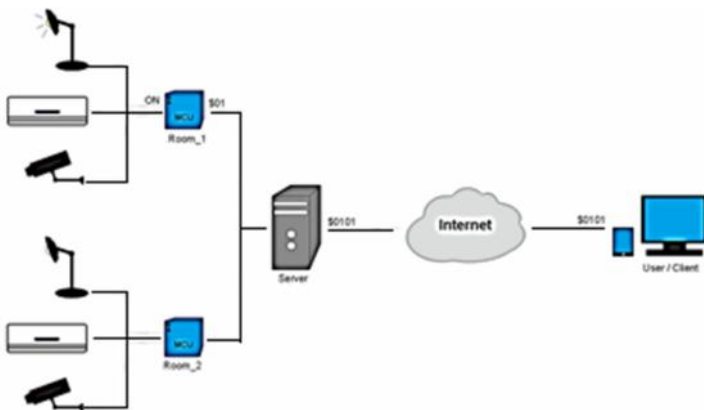
Tabel 2.4 Spesifikasi Arduino Nano [5]

Mikrokontroler	ATmega328
Dimensi	1.85cm x 4.318cm x 18mm
Berat	5g
Tegangan Kerja	5V
Clock Speed	16MHz
Memori	32KB
SRAM	2KB
EEPROM	1KB
Arus DC tiap pin I/O	40 mA

2.5. Internet of Things

Internet of Things adalah istilah yang digunakan dalam pemanfaatan koneksi internet secara terus menerus yang menghubungkan benda dengan internet baik sebagai pertukaran data, kendali jarak jauh, maupun hubungan dengan benda fisik. Definisi *Internet of Things* seringkali disalahartikan dengan jaringan sensor yang terhubung ke internet, *Big Data*, maupun komunikasi *Machine-to-machine*. *Internet of Things* berbeda dengan jaringan sensor dimana jaringan sensor tidak membutuhkan internet untuk mengelola data. *Internet of Things* juga berbeda dengan *Big Data* dimana *Internet of Things* menangkap dan menghasilkan data bukan menyimpan data (cloud) di *big store*. *Internet of Things* juga berbeda dengan *machine-to-machine* karena *Internet of Things* membutuhkan manusia untuk mengakses sistem[6].

Internet of Things sebagai kendali jarak jauh berfungsi memudahkan pengguna dalam mengendalikan benda fisik/aktuator melalui jaringan internet. *internet of things* memanfaatkan interaksi antar sesama mesin yang terhubung secara otomatis tanpa campur tangan manusia dalam jarak yang dibatasi oleh jaringan internet untuk melaksanakan perintah yang telah ditanam oleh pengguna. Ilustrasi tentang cara kerja *internet of things* sebagai kendali jarak jauh dapat dilihat pada Gambar 2.6.



Gambar 2.6 Ilustrasi cara kerja *internet of things*

2.6. LAMP



Gambar 2.7 Logo LAMP

LAMP (Linux, Apache, MySQL, PHP) adalah bundel perangkat lunak yang digunakan untuk keperluan *web-service*. LAMP merupakan akronim dari empat komponen *open source* utama: Sistem Operasi Linux, Apache HTTP Server, sistem manajemen database MySQL, dan bahasa pemrograman PHP. Keempat komponen tersebut bekerja secara sinkron sebagai komponen utama penyedia servis *web server*.

2.6.1. Linux

Linux adalah sistem operasi komputer bertipe Unix. Linux merupakan hasil pengembangan perangkat lunak bebas dan sumber terbuka utama sehingga, kode sumber linux dapat dimodifikasi, digunakan, dan didistribusikan kembali secara bebas oleh siapa saja.

Sistem operasi Linux banyak digunakan pada *web-server* dikarenakan Linux tidak bergantung kepada vendor (*vendor independence*), biaya operasional yang rendah, dan kompatibilitas yang tinggi dibandingkan versi UNIX tak bebas, serta faktor keamanan dan kestabilannya yang tinggi dibandingkan dengan sistem operasi lainnya seperti Microsoft Windows.

2.6.2. Apache

Server HTTP Apache adalah server web yang dapat dijalankan di banyak sistem operasi (Unix, BSD, Linux, Microsoft Windows dan Novell Netware serta platform lainnya) yang berguna untuk melayani dan memfungsikan situs web. Apache merupakan perangkat lunak sumber terbuka dikembangkan oleh

komunitas terbuka yang terdiri dari pengembang-pengembang dibawah naungan Apache Software Foundation.

2.6.3. My SQL

MySQL adalah sebuah implementasi dari sistem manajemen basisdata relasional (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). Setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya; SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian basisdata, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

2.6.4. PHP

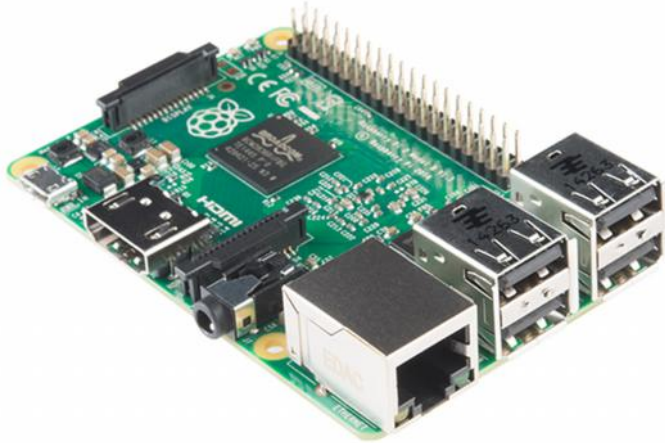
Hypertext Preprocessor (PHP) adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP berfungsi untuk memprogram situs web dinamisdimana PHP hanya mengeksekusi kode yang ditulis dalam pembatas sebagaimana ditentukan oleh dasar sintaks PHP.

2.7. HTTP

Hypertext Transfer Protocol (HTTP) adalah sebuah protokol jaringan lapisan aplikasi yang digunakan untuk sistem informasi terdistribusi, kolaboratif, dan menggunakan hipermedia. Sederhananya, HTTP adalah standar komunikasi yang digunakan untuk mengatur permintaan dan respon data antara *browser* pengguna dan web server.

2.8. Raspberry Pi

Raspberry pi adalah *single-board computer* yang dikembangkan di Inggris oleh Raspberry Pi *Foundation* yang bertujuan untuk mengenalkan komputer dasar kepada para pelajar. Raspberry Pi memiliki 27 GPIO, 4 USB ports, video/audio output, *High Definition Multimedia Interface* (HDMI), dan Raw LCD (DSI). Raspberry PI dapat dioperasikan dengan menggunakan USB charger dengan tegangan yang direkomendasikan sebesar 5 Volt.



Gambar 2.8 Raspberry Pi 2 [7]

Ada beberapa tipe Raspberry Pi berdasarkan fungsi dan spesifikasinya seperti Raspberry Pi Zero, Raspberry Pi A, Raspberry Pi B+, Raspberry Pi 2, dan Raspberry Pi 3. Setiap jenis Raspberry Pi memiliki fungsi dan aplikasi masing-masing, disini Raspberry Pi yang digunakan adalah Raspberry Pi 2. Tabel spesifikasi *single-board computer* Raspberry Pi 2 dapat dilihat pada Tabel 2.5 [7].

Tabel 2.5 Spesifikasi Raspberry Pi 2

CPU (Central Processing Unit)	Broadcom 900 MHz BCM2836 ARMv7 Quad Core Processor SoC
GPU (Graphics Processing Unit)	Broadcom Video Core IV
Memory	1GB RAM
<i>Low-Level Peripherals</i>	UART, I ² C, SPI, dan 27 GPIO
Tegangan Kerja	5V
Arus Kerja	600mA – 2.400mA
Sistem Operasi	Windows 10, Debian GNU/Linux, Fedora, Arch Linux, dan RISC OS.

2.9. Modul Kamera Raspberry Pi

Modul kamera Raspberry Pi adalah modul kamera *add-on* yang dirancang khusus untuk Raspberry Pi. Modul kamera Raspberry Pi memiliki dimensi dan berat yang kecil sehingga cocok digunakan untuk keperluan mobile.

Ada beberapa tipe modul kamera Raspberry Pi berdasarkan fungsi dan spesifikasinya seperti modul kamera v1, modul kamera v2, dan modul kamera No-Ir . Setiap jenis Raspberry Pi memiliki fungsi dan aplikasi masing-masing, disini modul kamera Raspberry Pi yang digunakan adalah modul kamera Raspberry Pi v1. Tabel spesifikasi modul kamera Raspberry Pi v1 dapat dilihat pada Tabel 2.6 [8].



Gambar 2.9 Modul kamera Raspberry Pi [8]

Tabel 2.6 Spesifikasi modul kamera Raspberry Pi v1

Ukuran	25mm x 20mm x 9mm
Berat	3g
Kualitas Gambar	Up to 2592 x 1944 pixel
Kualitas Video	1080p30, 720p60 dan 640x480p60/90

2.10. Komunikasi Serial

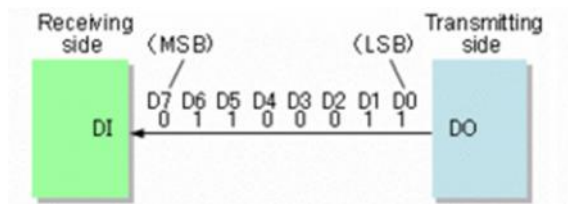
Komunikasi serial adalah proses pengiriman data secara serial sebanyak satu bit dalam satu waktu melalui saluran komunikasi atau bus computer. Hal ini berkebalikan dengan komunikasi paralel dimana beberapa bit dikirimkan secara langsung.

Komunikasi serial merupakan protocol komunikasi yang banyak digunakan dalam jaringan komputer dikarenakan biaya kabel yang digunakan rendah (hanya memerlukan 2 kabel) dan proses sinkronisasi yang mudah. Ilustrasi singkat tentang cara kerja dari komunikasi serial dapat dilihat pada Gambar 2.10 [9].

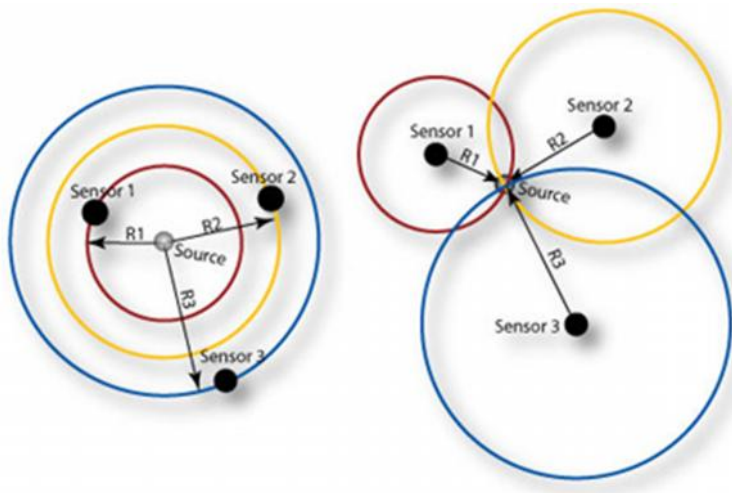
2.11. Indoor Positioning System (IPS)

Indoor Positioning System (IPS) adalah sistem yang digunakan untuk mencari lokasi dari suatu objek didalam sebuah bangunan dengan menggunakan gelombang radio, medan magnet, atau sensor lainnya. IPS melakukan pelacakan lokasi berdasarkan informasi yang diberikan oleh alat pengukuran, dimana sistem IPS memerlukan minimal 3 alat pengukuran untuk mendapatkan hasil yang maksimal.

Pada jaringan komunikasi wireless, sistem pelacakan objek umumnya menggunakan teknik lokalisasi intensitas penerimaan sinyal atau *received signal strength indication (RSSI)* untuk melakukan pelacakan objek berdasarkan kekuatan sinyal dari peralatan pengguna terhadap berbagai akses point. Metode pelacakan dengan RSSI dengan teknik trilaterasi merupakan salah satu metode pelacakan yang memiliki tingkat reabilitas yang tinggi karena kemudahan dalam implementasi sistem pelacakan. Akurasi dari pelacakan objek menggunakan intensitas penerimaan sinyal bergantung pada banyaknya akses point dan stabilitas sinyal pada sistem pelacakan. Ilustrasi penggunaan RSSI pada IPS dapat dilihat pada Gambar 2.11 [10].



Gambar 2.10 Ilustrasi cara kerja komunikasi serial



Gambar 2.11 Ilustrasi penggunaan RSSI pada IPS

BAB III

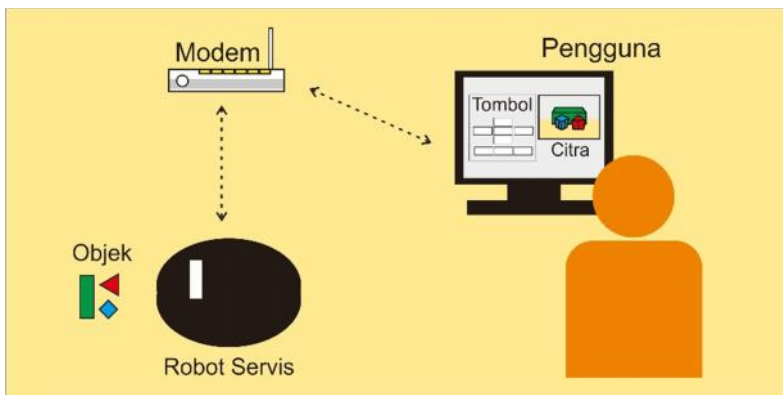
PERANCANGAN SISTEM

Pada bab ini akan dibahas perancangan keseluruhan sistem dimulai dari blok diagram dan flowchart sistem kendali jarak jauh berbasis *internet of things* sampai pada perancangan perangkat keras dan lunak dari sistem.

Tugas akhir ini menggunakan internet sebagai media komunikasi antara pengguna dan sistem kendali. Sistem kendali ini digunakan untuk mengendalikan robot servis sesuai dengan perintah yang diinginkan oleh pengguna. Pada sistem ini, pengguna akan memonitoring kondisi lingkungan sekitar melalui tampilan citra yang dikirimkan oleh sistem dan memberikan perintah baik perintah otonom maupun perintah yang dipandu pengguna kepada sistem. Ilustrasi kendali jarak jauh robot servis berbasis *Internet of Things* dapat dilihat pada Gambar 3.1.

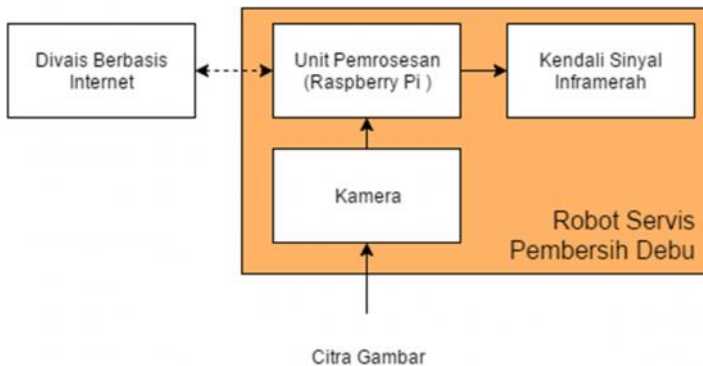
3.1. Diagram Blok dan *Flowchart* Sistem

Diagram blok dan *flowchart* sistem bertujuan untuk menjelaskan konsep keseluruhan perancangan sistem kendali jarak jauh robot servis berbasis *internet of things*. Sistem ini dirancang untuk memudahkan pengguna dalam mengendalikan robot servis melalui peralatan berbasis jaringan internet.



Gambar 3.1 Ilustrasi kendali jarak jauh robot servis

3.1.1. Diagram Blok



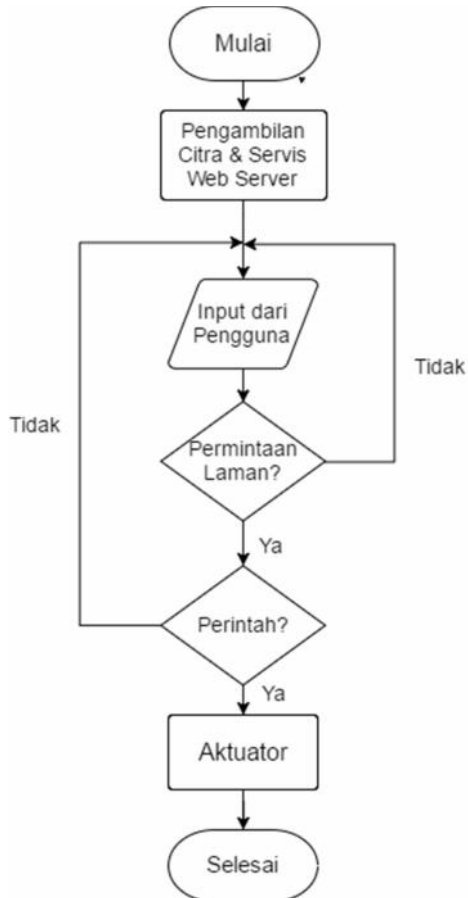
Gambar 3.2 Diagram Blok Sistem

Pada Gambar 3.2 dijelaskan diagram blok dari sistem sebagai berikut :

- Robot servis pembersih debu bertindak sebagai aktuator dari sistem kendali jarak jauh dimana robot servis akan dikendalikan oleh pengguna melalui kendali sinyal inframerah.
- Unit pemrosesan raspberry pi bertindak sebagai *web server*. Raspberry pi menunggu permintaan *web-page* dari pengguna, memproses data pada *Central Processing Unit* (CPU), mengambil informasi dari *hard disk* komputer, dan menampilkan *web-page* kepada pengguna. Data yang diproses pada raspberry pi diteruskan ke mikrokontroler arduino untuk diproses lebih lanjut.
- Kamera berupa modul kamera raspberry pi v1 mengambil citra gambar dari lingkungan di sekitar robot servis dan meneruskannya ke raspberry pi untuk ditampilkan pada halaman *web-page* yang akan diakses oleh pengguna.
- Kendali sinyal inframerah bertindak sebagai kendali robot servis pembersih debu. Sistem kendali sinyal inframerah untuk mengendalikan robot servis pembersih debu diproses menggunakan mikrokontroler arduino melalui emitter

sinyal inframerah. Mikrokontroller arduino memproses data yang diterima dari raspberry pi melalui komunikasi serial menjadi sinyal infra merah yang akan dikirim melalui *emitter* sinyal infra merah untuk mengendalikan robot servis pembersih debu.

3.1.2. Flowchart



Gambar 3.3 Flowchart Sistem

Pada Gambar 3.3 dijelaskan flowchart dari sistem sebagai berikut :

1. Saat sistem dinyalakan, sistem akan mengaktifkan driver kamera dan servis perangkat lunak *web-server* yang sudah terintegrasi dengan raspberry pi.
2. Server membaca permintaan laman *web-page* dari pengguna yang telah terhubung melalui jaringan internet. Jika server telah menerima permintaan laman *web-page* dari pengguna, maka server akan mencari permintaan *web-page* yang tersimpan pada hardisk server dan menampilkan halaman *web-page* ke layar pengguna.
3. Setelah server mengirimkan dan menampilkan antarmuka *web-page* pada layar pengguna, server akan menunggu interaksi atau perintah yang dikirimkan oleh pengguna sambil menampilkan *feedback* berupa antarmuka citra dari kamera pada antarmuka *web-page*.
4. Perintah yang telah diinput oleh pengguna melalui tombol pada antarmuka *web-page* akan diterima oleh server dan diproses untuk mengendalikan aktuatur berupa robot servis.

3.2. Perancangan Robot Servis Pembersih Debu

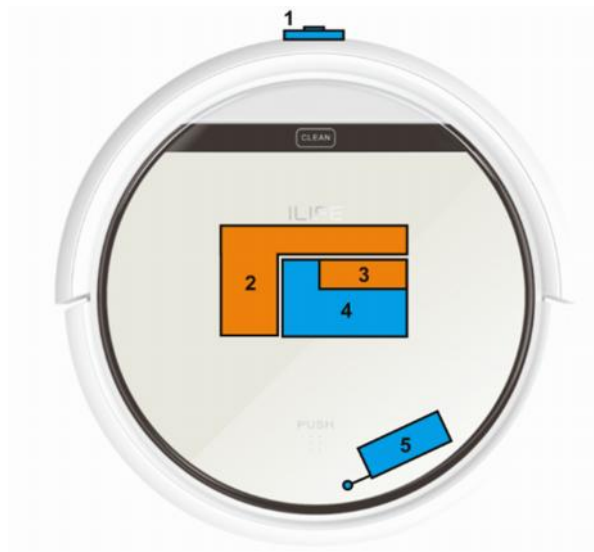
Perancangan robot servis pembersih debu bertujuan untuk menempatkan sistem kendali jarak jauh robot servis pembersih debu dan merancang robot servis pembersih debu sebagai aktuatur dari sistem. Perancangan perangkat keras terdiri dari perancangan robot servis, penempatan arduino nano, kamera, raspberry pi, tempat penyimpanan debu, dan suplai daya.

3.2.1. Robot Servis

Robot servis pembersih debu dirancang dengan desain mekanik yang tipis dan elegan sehingga robot servis mampu menjangkau area yang sulit dibersihkan. Komponen perangkat keras yang digunakan dalam rancang bangun kendali jarak jauh berbasis *internet of things* pada robot servis seperti raspberry pi, arduino nano, modul kamera, dan suplai daya didesain sedemikian rupa sehingga tidak mengganggu fungsionalitas robot pembersih debu otonom. Foto dan diagram robot pembersih debu berbasis *internet of thing* dapat dilihat pada Gambar 3.4 dan 3.5.



Gambar 3.4 Foto Robot Servis



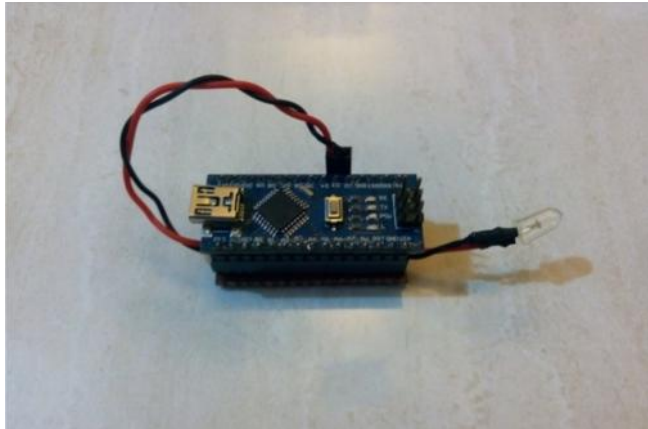
Gambar 3.5 Diagram Robot Servis

Pada Gambar 3.5 dijelaskan diagram dari robot servis berbasis *internet of things* sebagai berikut :

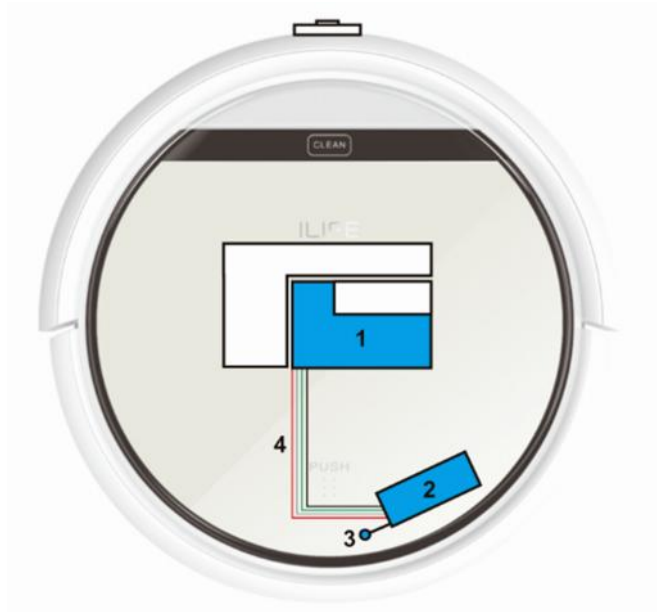
1. Modul kamera raspberry pi v1. Modul kamera raspberry pi v1 diletakkan pada bagian depan dari robot servis pembersih debu untuk memudahkan pengguna dalam memonitoring kondisi lingkungan di sekitar robot servis pembersih debu yang akan dibersihkan.
2. Tempat penyimpanan debu. Tempat penyimpanan debu menyimpan debu yang disedot oleh robot servis pembersih debu. Tempat penyimpanan debu diletakkan dekat dengan motor *vacuum* untuk mendapatkan hasil yang maksimal.
3. LM2596, modul *buck converter* yang digunakan untuk mengkonversi nilai tegangan dari suplai daya robot servis menjadi tegangan yang sesuai dengan suplai daya yang dibutuhkan oleh sistem kendali robot servis pembersih debu.
4. *Single-board computer* raspberry pi, Raspberry pi sebagai unit pemrosesan utama diletakkan pada bagian tengah dari robot servis pembersih debu untuk memudahkan akses dari perangkat keras lainnya yang terhubung ke raspberry pi.
5. Mikrokontroler arduino nano dan *emitter* sinyal inframerah. Mikrokontroler arduino nano dan *emitter* sinyal inframerah sebagai unit kendali robot servis diletakkan di bagian belakang robot servis dekat dengan *receiver* sinyal inframerah sehingga memudahkan proses kendali robot servis.

3.2.2. Arduino Nano

Mikrokontroler arduino nano berfungsi sebagai unit pemrosesan kendali robot servis sehingga harus diletakkan secara baik. Mikrokontroler arduino nano mendapatkan sumber tenaga dan perintah kendali dari *single-board computer* raspberry pi melalui kabel serial. Kabel serial yang menghubungkan arduino nano memiliki format *Universal Serial Bus (USB) male* dan mini-B *male*. Foto dan diagram mikrokontroler arduino dapat dilihat pada Gambar 3.6 dan Gambar 3.7.



Gambar 3.6 Foto Arduino Nano



Gambar 3.7 Diagram arduino nano pada robot servis

Pada Gambar 3.7 dijelaskan diagram mikrokontroler arduino nano pada robot servis sebagai berikut :

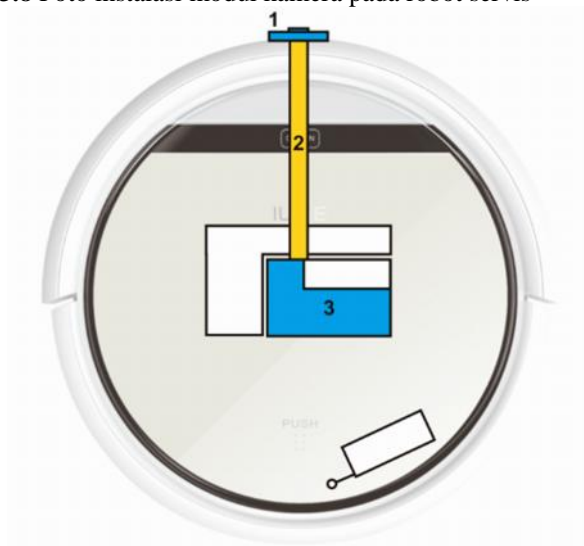
1. *Single-board computer* raspberry pi, Raspberry pi sebagai sumber suplai daya dari mikrokontroler arduino berfungsi memberikan perintah kepada mikrokontroler arduino melalui kabel serial.
2. Mikrokontroler arduino nano sebagai unit pemrosesan kendali robot servis ditempatkan di bagian belakang robot servis dekat dengan *receiver* sinyal inframerah untuk memudahkan kendali robot servis.
3. IR333, *emitter* sinyal inframerah, berfungsi sebagai aktuator dari mikrokontroler arduino nano. IR333 akan mengirimkan sinyal inframerah yang telah diencode oleh mikrokontroler arduino nano.
4. Kabel serial berfungsi untuk sebagai media pengiriman daya dan data kepada mikrokontroler arduino nano dari raspberry pi. Kabel serial yang digunakan menggunakan model *Universal Serial Bus* (USB) male ke *Mini-B male*. Kabel serial terdiri dari 4 kabel utama yaitu: vcc (merah), data- (putih), data+ (hijau), dan ground (hitam). Kabel vcc (merah) mensuplai tegangan sebesar 5v ke arduino nano sebagai sumber daya bagi arduino nano untuk melakukan pemrosesan. Kabel data positif (hijau) dan negatif (putih) berfungsi sebagai media pengiriman perintah kepada mikrokontroler arduino. Kabel ground (hitam) selain berfungsi sebagai grounding mikrokontroler arduino, juga sebagai referensi pengiriman data.

3.2.3. Kamera

Modul kamera raspberry pi v1 berfungsi sebagai perangkat keras dalam pengambilan citra gambar dalam sistem. Modul kamera raspberry pi ditempatkan di bagian depan robot servis pembersih debu untuk memudahkan pengguna dalam memonitoring kondisi lingkungan disekitar robot servis. Modul kamera dihubungkan dengan *single-board computer* raspberry pi melalui kabel fleksibel sepanjang 50cm. Foto instalasi dan diagram modul kamera raspberry pi dapat dilihat pada Gambar 3.8 dan 3.9.



Gambar 3.8 Foto instalasi modul kamera pada robot servis



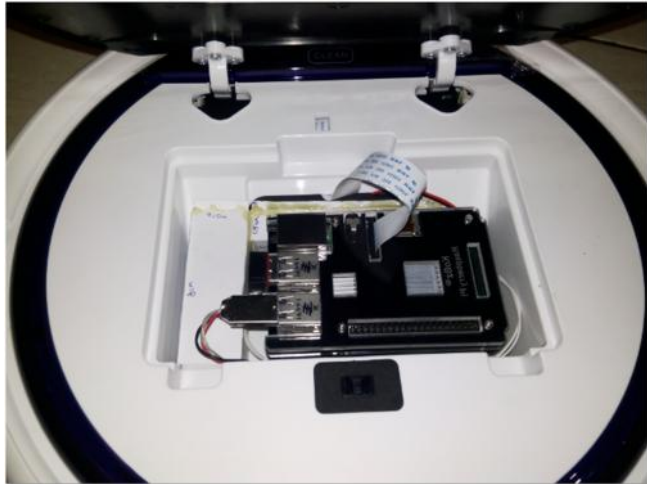
Gambar 3.9 Diagram modul kamera pada robot servis

Pada Gambar 3.9 dijelaskan diagram modul kamera raspberry pi v1 pada robot servis sebagai berikut :

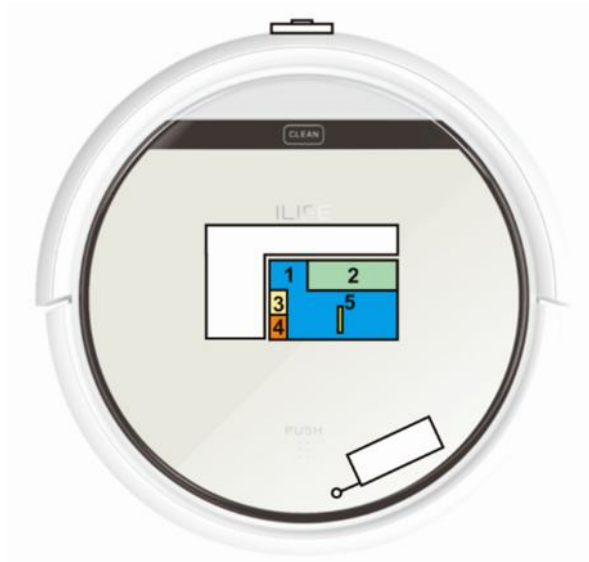
1. Modul kamera raspberry pi v1. Modul kamera raspberry pi v1 ditempatkan dibagian depan robot servis untuk memudahkan pengguna dalam memonitoring kondisi lingkungan yang akan diberikan.
2. Kabel fleksibel. Kabel fleksibel berfungsi untuk menghubungkan modul kamera raspberry pi v1 dengan *single-board computer* raspberry pi. Kabel fleksibel yang digunakan adalah kabel fleksibel 15 pin dengan panjang 50cm. Kabel fleksibel yang menghubungkan modul kamera raspberry pi v1 dengan *single-board computer* raspberry pi secara garis besar terdiri dari dua jaringan data, satu jaringan *clock*, jaringan sda, jaringan scl, sumber tegangan 3V ,dan ground.
3. *Single-board computer* raspberry pi. *Single-board computer* raspberry pi berfungsi untuk mengambil citra gambar dari kamera dan memproses gambar tersebut menjadi format video untuk ditampilkan pada antarmuka web-page yang akan diakses oleh pengguna. *Single-board computer* raspberry pi terhubung dengan kamera melalui socket camera.

3.2.4. Raspberry Pi

Single-board computer raspberry pi berfungsi sebagai penyedia servis *web server* dan unit pemrosesan antarmuka *web-page* pada sistem. Raspberry pi ditempatkan pada bagian tengah robot servis pembersih debu sehingga memudahkan akses dari perangkat keras lainnya ke *single-board computer* raspberry pi. Perangkat keras yang terhubung dengan *single-board computer* raspberry pi adalah modul kamera raspberry pi v1, mikrokontroler arduino nano dengan *emitter* sinyal inframerah, dan LM2596 sebagai suplai daya sebesar 3.5W hingga 10W dengan tegangan masukan ke *single-board computer* raspberry pi sebesar 5V. Foto instalasi dan diagram *single-board computer* raspberry pi dapat dilihat pada Gambar 3.10 dan 3.11.



Gambar 3.10 Foto instalasi raspberry pi pada robot servis



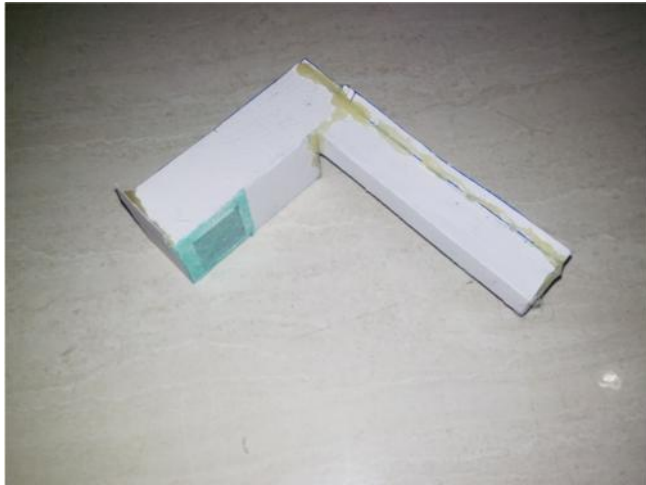
Gambar 3.11 Diagram raspberry pi pada robot servis

Pada Gambar 3.11 dijelaskan diagram *single-board computer* raspberry pi pada robot servis sebagai berikut :

1. *Single-board computer* raspberry pi. Raspberry pi berfungsi sebagai unit pemrosesan utama dalam sistem kendali robot servis pembersih debu berbasis *internet of things*. Raspberry pi ditempatkan pada bagian tengah robot servis pembersih debu untuk memudahkan akses perangkat keras lainnya ke raspberry pi.
2. LM2596 sebagai sumber daya raspberry pi mengkonversi tegangan dari tegangan baterai robot servis pembersih debu sebesar 11V menjadi tegangan sebesar 5V.
3. Wifi dongle. Wifi dongle digunakan sebagai perangkat keras penghubung raspberry pi dengan jaringan internet. Wifi dongle pada raspberry pi 2 dapat langsung digunakan tanpa perlu melakukan instalasi driver.
4. Female USB digunakan sebagai penghubung raspberry pi dengan mikrokontroller arduino nano. Melalui kabel serial yang terhubung pada *female* USB di raspberry pi, raspberry pi akan memberikan perintah kepada mikrokontroler arduino nano.
5. Socket kamera. Socket kamera merupakan perangkat keras yang digunakan untuk menghubungkan modul kamera raspberry pi v1 dengan *single-board computer* raspberry pi melalui kabel fleksibel sebanyak 15pin. Melalui kabel fleksibel pada socket kamera, *single-board computer* raspberry pi akan mengambil citra gambar dan ditampilkan pada antarmuka *web-page*.

3.2.5. Tempat penyimpanan debu

Tempat penyimpanan debu pada robot servis pembersih debu dimodifikasi sedemikian rupa untuk meminimalisir ukuran tempat penyimpanan debu asli dari robot servis pembersih debu tanpa mengurangi fungsionalitas dari tempat penyimpanan debu asli. Berikut adalah foto dan diagram tempat penyimpanan debu pada sistem kendali jarak jauh robot servis pembersih debu berbasis *internet of things*:



Gambar 3.12 Foto tampak atas tempat penyimpanan debu



Gambar 3.13 Foto tampak bawah tempat penyimpanan debu



Gambar 3.14 Foto sebelum instalasi tempat penyimpanan debu



Gambar 3.15 Foto sesudah instalasi tempat penyimpanan debu



Gambar 3.16 Diagram tempat penyimpanan debu

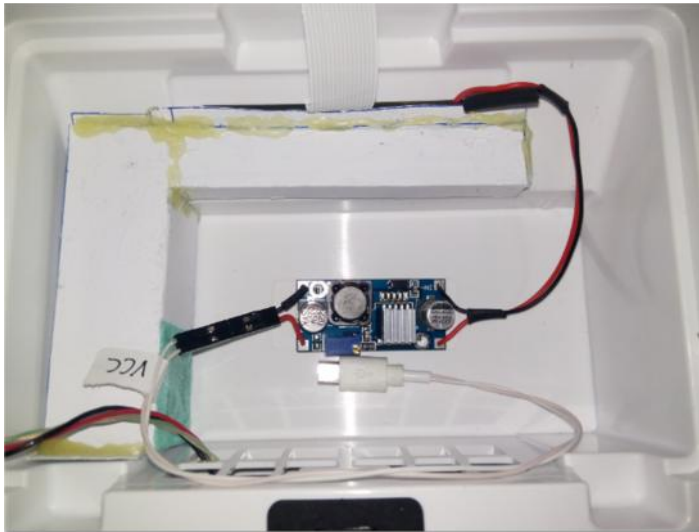
Pada Gambar 3.16 dijelaskan diagram tempat penyimpanan debu pada robot servis sebagai berikut :

1. Masukan tempat penyimpanan debu. Bagian masukan pada tempat penyimpanan debu berfungsi sebagai jalan masuk bagi debu yang disedot oleh robot servis, robot pembersih debu. Masukan tempat penyimpanan debu telah disesuaikan dengan masukan debu pada robot pembersih debu sehingga meminimalisir adanya debu di bagian unit pemorsesan.
2. Tempat penyimpan debu. Tempat penyimpanan debu berfungsi sebagai tempat untuk menyimpan debu yang hasil dari proses pembersihan yang dilakukan oleh robot servis berupa robot pembersih debu otonom.
3. Ventilasi udara. Ventilasi udara berfungsi sebagai jalan keluar dari udara yang disedot oleh robot pembersih debu otonom. Selain itu, ventilasi udara berfungsi untuk mencegah debu masuk ke bagian suplai daya dan *single-board computer raspberry pi*.

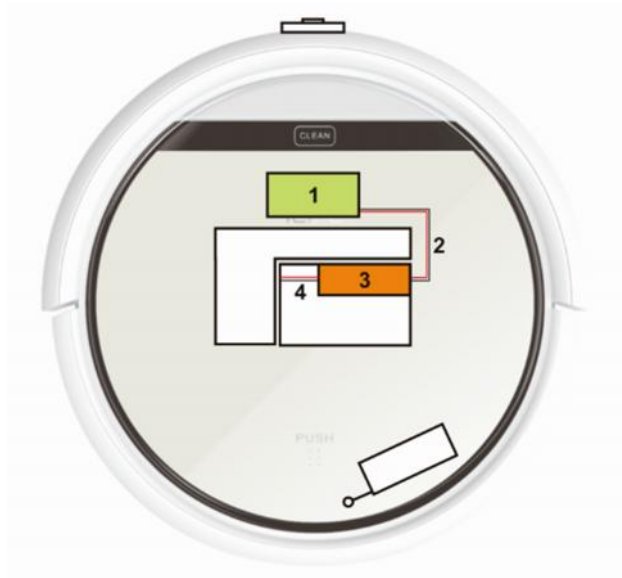
3.2.6. Suplai Daya

Suplai daya dalam rancang bangun kendali robot servis berbasis *internet of things* menggunakan baterai utama dari robot servis penyedot debu yang telah dikonversi menjadi tegangan yang sesuai dengan komponen perangkat keras sistem kendali raspberry pi. Rangkaian suplai daya menggunakan *buck converter* LM2596 yang mengkonversi tegangan baterai sebesar 11.1V menjadi 5V dengan arus maksimal sebesar 3A. Berikut adalah foto instalasi suplai daya pada robot servis.

Pada Gambar 3.17 terlihat bahwa *buck converter* LM2596 ditempatkan sedemikian rupa pada tempat penyimpanan debu sehingga rangkaian *buck converter* LM2596 tidak mengganggu suplai daya original pada robot servis pembersih debu. Hasil proses konversi tegangan sebesar 5V digunakan untuk mensuplai *single-board computer* raspberry pi melalui kabel *micro USB* yang terhubung pada keluaran *buck converter* LM2596. Foto diagram suplai daya pada robot servis pembersih debu dapat dilihat pada Gambar 3.18.



Gambar 3.17 Foto instalasi suplai daya



Gambar 3.18 Diagram suplai daya pada robot servis

Pada Gambar 3.18 dijelaskan diagram suplai daya pada robot servis sebagai berikut :

1. Baterai robot servis. Baterai pada robot servis memiliki tegangan sebesar 11V dengan kapasitas 2500mAh.
2. Kabel input power suplai. Kabel input power suplai berupa kabel vcc (merah) dan kabel ground (hitam). Input power suplai nantinya akan diproses oleh *buck converter* LM2596.
3. *Buck converter* LM2596. *Buck converter* LM2596 mengkonversi tegangan input sebesar 11V menjadi tegangan output sebesar 5V yang sesuai dengan tegangan yang diperlukan oleh *single-board computer* raspberry pi. *Buck converter* LM2596 dapat mensuplai daya hingga 15W.
4. Kabel output power suplai. Kabel output power suplai berupa kabel vcc (merah) dan kabel ground (hitam).

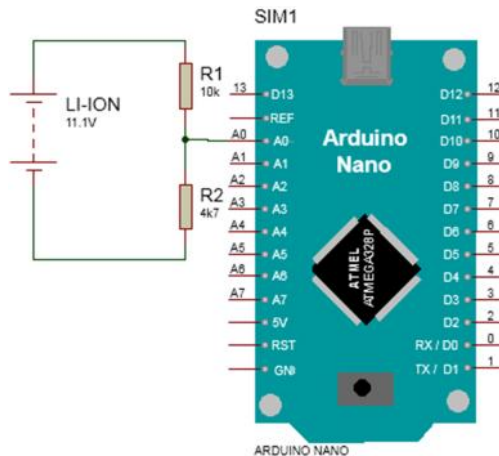
Output power suplai dari *buck converter* LM2596 terhubung dengan input *single-board computer* raspberry pi melalui kabel *micro USB*.

3.2.7. Perancangan Monitoring Daya

Perancangan monitoring daya dalam rancang bangun kendali jarak jauh robot servis pembersih debu berbasis *internet of things* bertujuan untuk memberikan feedback kepada pengguna tentang kondisi daya baterai pada robot servis pembersih debu. Monitoring daya dilakukan dengan mengukur tegangan pada baterai lithion-ion pada robot servis dimana tegangan kerja robot servis saat baterai penuh sebesar 11.1V dan tegangan cutoff saat baterai habis sebesar 8.4V [11]. Rangkaian monitoring daya pada robot servis pembersih debu dapat dilihat pada Gambar 3.19.

3.3. Perancangan Unit Pemrosesan

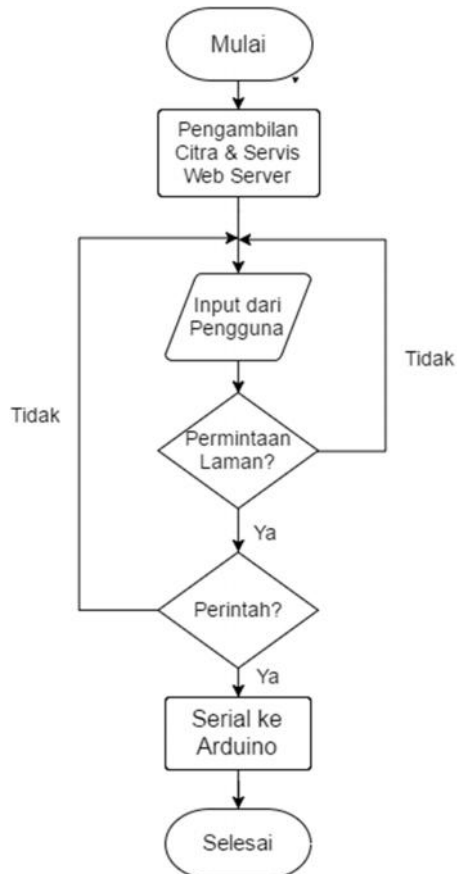
Perancangan unit pemrosesan pada *single-board computer* raspberry pi bertujuan untuk melakukan proses servis web-server, perancangan kendali *user-guided*, dan perancangan antarmuka *web-page*.



Gambar 3.19 Rangkaian Monitoring Daya

3.3.1. Perancangan Web Server

Perancangan perangkat lunak *web-server* bertujuan untuk menghubungkan sistem kendali jarak jauh dengan jaringan internet sehingga pengguna dapat mengendalikan robot servis melalui perangkat berbasis jaringan internet. Perancangan *web-server* dilakukan pada *single-board computer* raspberry pi. *Flowchart* dari *web-server* pada raspberry pi dapat dilihat pada Gambar 3.20.

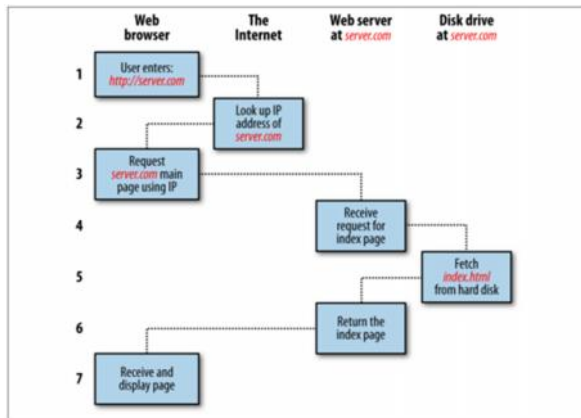


Gambar 3.20 *Flowchart web-server*

Pada Gambar 3.20 dijelaskan *flowchart web-server* pada *single-board computer raspberry pi*:

1. Saat raspberry pi dinyalakan, raspberry pi akan mengaktifkan driver kamera dan servis apache sebagai algoritma pemrograman utama dari *web-server*.
2. *Web-server* apache menunggu permintaan *web-page* dari pengguna yang terhubung melalui jaringan internet. Jika server telah menerima permintaan *web-page* dari pengguna, maka server akan mencari permintaan *web-page* di hardisk server dan menampilkan *web-page* ke layar pengguna.
3. *Web server* akan menunggu perintah yang dikirimkan oleh pengguna sambil menampilkan *feedback* berupa tampilan kamera pada antarmuka *web-page*.
4. Perintah yang telah diinput oleh pengguna melalui tombol pada antarmuka *web-page* akan diterima oleh server dan diteruskan ke mikrokontroler arduino nano untuk diproses lebih lanjut sesuai dengan permintaan pengguna.

Berikut adalah diagram sekuensial antara client dan server pada servis web server:



Gambar 3.21 Diagram sekuensial *client/server* [12]

Pada Gambar 3.21 dijelaskan hubungan interaksi antara klien dengan server, dimana langkah langkah dalam permintaan dan balasan dari server sebagai berikut :

1. Klien memasukan alamat `http://server.com` pada *address bar* di *web-browser*.
2. *Web-browser* klien mencari alamat *Internet Protocol* (IP) untuk `server.com` pada jaringan yang terhubung ke internet.
3. *Web-browser* klien melakukan permintaan *homepage* pada `www.server.com`
4. Permintaan masuk ke *web-server* pada `www.server.com` melalui internet.
5. Setelah menerima permintaan dari klien, *web-server* mencari data *homepage* berupa `index.html` pada hardisk *web-server*.
6. *Webpage* dikirimkan oleh server dan diterima oleh *web-browser*.
7. *Web-browser* menampilkan *web-page* kepada klien.

3.3.2. Perancangan User Guided Control

Perancangan *user-guided control* merupakan kendali jarak jauh robot servis yang memerlukan pengawasan dari pengguna. *User-guided control* bertujuan untuk memudahkan pengguna dalam mengendalikan robot servis sehingga pengguna tidak perlu untuk menekan perintah tertentu secara berulang – ulang. Berikut adalah daftar perintah *user-guided control* pada perancangan kendali jarak jauh robot servis berbasis *internet of things* :

Tabel 3.1 Daftar perintah *user-guided control*

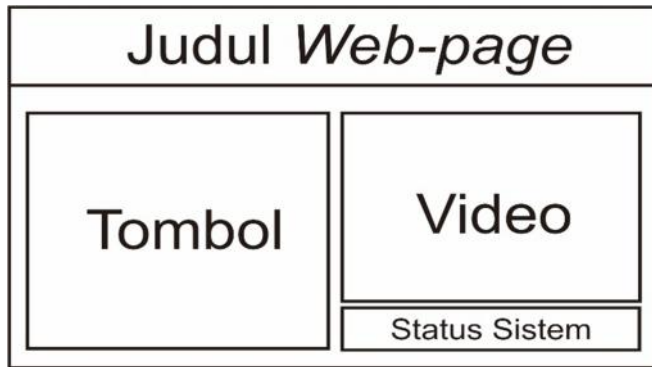
Perintah	Keterangan
<i>View-Left</i>	Robot Servis akan berputar sebesar 360° ke kiri
<i>Keep Forward</i>	Robot servis akan terus bergerak maju
<i>View-Right</i>	Robot Servis akan berputar sebesar 360° ke kanan

Pada Tabel 3.1 dijelaskan daftar perintah *user-guided control* sebagai berikut:

- *View-left*, dimana robot servis pembersih debu akan berputar sebesar 360° ke kiri untuk memberikan tampilan citra lingkungan di sekitar robot servis dimulai dari bagian kiri robot. Robot servis pembersih debu dapat dihentikan oleh pengguna saat berputar dengan menekan kembali tombol *view-left* atau menekan tombol *clean* pada antarmuka *web-page*.
- *Keep Forward*, dimana robot servis pembersih debu akan terus bergerak maju. Robot servis pembersih debu akan terus bergerak maju sampai pengguna menekan kembali tombol *keep forward* atau menekan tombol *clean* pada antar muka *web-page*.
- *View-right*, dimana robot servis pembersih debu akan berputar sebesar 360° ke kanan untuk memberikan tampilan citra lingkungan di sekitar robot servis pembersih debu dimulai dari bagian kanan robot servis. Robot servis pembersih debu dapat dihentikan oleh pengguna saat berputar dengan menekan kembali tombol *view-right* atau menekan tombol *clean* pada antarmuka *web-page*.

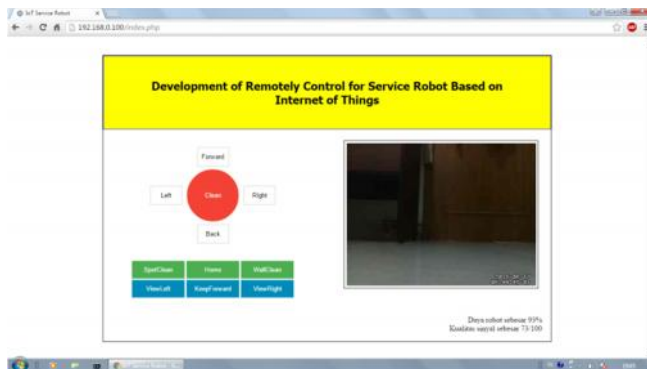
3.3.3. Perancangan Antarmuka *Webpage*

Perancangan antarmuka *web-page* bertujuan sebagai sarana interaksi antara pengguna dengan server sehingga pengguna dapat memberikan perintah dan memonitoring kondisi dari robot servis pembersih debu maupun lingkungan di sekitar robot servis pembersih debu. Perancangan antarmuka *web-page* dilakukan pada *single-board computer raspberry pi* melalui servis LAMP (*Linux Apache MySQL, and PHP*) dengan menggunakan bahasa pemrograman Hyper Text Markup Language (HTML) dan diolah dengan Cascading Style Sheet (CSS). Rancangan antarmuka *web-page* yang terdiri dari tombol, video, dan status dari sistem berupa kondisi suplai daya dan sinyal dari sistem dapat dilihat pada Gambar 3.22.



Gambar 3.22 Rancangan antarmuka *web-page*

Pada Gambar 3.22 dijelaskan rancangan antarmuka *web-page* yang terdiri dari tombol perintah, *streaming video*, dan status dari sistem. Tombol perintah dalam antarmuka *web-page* digunakan untuk mempermudah pengguna dalam mengendalikan robot servis. *Streaming video* dalam antaramuka *web-page* digunakan untuk memonitoring lingkungan sekitar robot servis. Sedangkan status sistem digunakan untuk memonitoring suplai daya dan sinyal pada sistem kendali. Berikut adalah antarmuka *web-page* pada berbagai peralatan berbasis jaringan internet:



Gambar 3.23 Antarmuka *web-page* pada computer

3.4. Perancangan Kamera

Data citra yang didapat dari modul kamera raspberry pi v1 disimpan ke dalam bentuk gambar dengan ukuran hingga 2592 x 1944 pixel. Semakin besar ukuran pixel yang disimpan oleh kamera, maka semakin besar juga waktu yang dibutuhkan oleh *single-board computer* raspberry pi untuk memproses gambar yang akan ditampilkan ke antarmuka web-page. Oleh karena itu, dalam perancangan pengambilan citra pada kendali jarak jauh robot servis berbasis *internet of things* digunakan pengambilan citra dengan ukuran gambar sebesar 320 x 240 pixel.

Gambar yang telah disimpan dalam hardisk raspberry pi selanjutnya akan diproses oleh *single-board computer* raspberry pi menjadi gambar bergerak (mjpeg) yang akan ditampilkan dalam antarmuka *web-page*. Format *Motion JPEG* (mjpeg) digunakan sebagai feedback citra pada antarmuka webpage dikarenakan format mjpeg dapat diakses oleh sebagian besar *web-browser* seperti Safari, Google Chrome, Mozilla Firefox, dan Opera.



Gambar 3.24 Frame Citra

3.5. Perancangan Kendali Sinyal Inframerah

Perancangan perangkat lunak kendali sinyal inframerah bertujuan untuk mengendalikan robot servis melalui sinyal inframerah yang telah diencode. Perancangan kendali sinyal inframerah dilakukan pada mikrokontroller arduino nano dengan *emitter* sinyal inframerah terhubung ke pin D3. *Flowchart* sistem kendali sinyal inframerah dapat dilihat pada Gambar 3.25.



Gambar 3.25 *Flowchart* sistem kendali sinyal inframerah

Pada Gambar 3.25 dijelaskan *flowchart* sistem kendali sinyal inframerah :

1. Saat sistem kendali sinyal inframerah diaktifkan, mikrokontroler arduino nano yang mendapat suplai daya dari *single-board computer* raspberry pi akan membaca data serial yang akan dikirimkan oleh *single-board computer* raspberry pi melalui kabel serial.
2. Jika ada perintah yang dikirimkan oleh *single-board computer* raspberry pi, mikrokontroler arduino nano akan memproses perintah dengan mencocokkan perintah dengan daftar kode inframerah yang terdapat dalam memori arduino.
3. Jika tidak ada perintah yang dikirimkan oleh *single-board computer* raspberry pi, maka mikokontroler akan menunggu hingga data serial dari *single-board computer* raspberry pi dikirimkan.
4. Setelah memdapatkan kode inframerah yang sesuai, mikrokontroler arduino nano akan mendekode sinyal inframerah dan dikirimkan ke *emitter* sinyal inframerah melalui pin D3.

Adapun tabel kode sinyal inframerah yang akan digunakan untuk proses dekodifikasi adalah:

Tabel 3.2 Kode sinyal inframerah

Perintah	Kode Hex	Tipe Dekodifikasi
Clean	2AA22DD	NEC 32 bits
Up	2AA55AA	NEC 32 bits
Down	2AA6699	NEC 32 bits
Right	2AA44BB	NEC 32 bits
Left	2AA33CC	NEC 32 bits
Spot-Clean	2AA7788	NEC 32 bits
Home	2AA8877	NEC 32 bits
Edge-Clean	2AA9966	NEC 32 bits

BAB IV PENGUJIAN

Pada bab ini akan dibahas mengenai pengujian dari sistem yang telah dirancang. Bab ini bertujuan untuk mengetahui apakah tujuan dalam perancangan sistem pada tugas akhir ini telah terlaksana atau tidak.

4.1. Pengujian Kendali Jarak Jauh

Pengujian yang dilakukan pada pengujian kendali jarak jauh bertujuan untuk mengetahui keandalan kendali jarak jauh berbasis *internet of things*. Pengujian dilakukan dengan memberikan perintah kendali secara spesifik melalui perangkat berbasis internet seperti komputer dan *smartphone* yang terhubung ke sistem melalui jaringan internet.

Pengujian dilakukan dengan menggunakan jaringan lokal (*localhost*) untuk meminimalisir gangguan yang terjadi karena konfigurasi jaringan. Pengujian dimulai dengan memberikan perintah melalui tombol pada *web-page* secara spesifik dengan pengujian pada setiap perintah dilakukan sebanyak sepuluh kali. Dari pengujian tersebut didapat data-data sebagai berikut:

Tabel 4.1 Tabel pengujian kendali jarak jauh

No.	Perintah	Tingkat Keberhasilan	Keterangan
1	<i>Clean</i>	100%	Robot membersihkan
2	<i>Up</i>	100%	Robot bergerak maju
3	<i>Down</i>	100%	Robot bergerak mundur
4	<i>Right</i>	100%	Robot berputar 45° ke kanan
5	<i>Left</i>	100%	Robot berputar 45° ke kiri
6	<i>Spot-Clean</i>	100%	Robot membersihkan
7	<i>Home</i>	100%	Robot kembali ke <i>dock</i>
8	<i>Wall-Clean</i>	100%	Robot membersihkan
9	<i>View-Left</i>	100%	Robot berputar 360° ke kiri
10	<i>Keep Forward</i>	100%	Robot bergerak maju
11	<i>View-Right</i>	100%	Robot berputar 360° ke

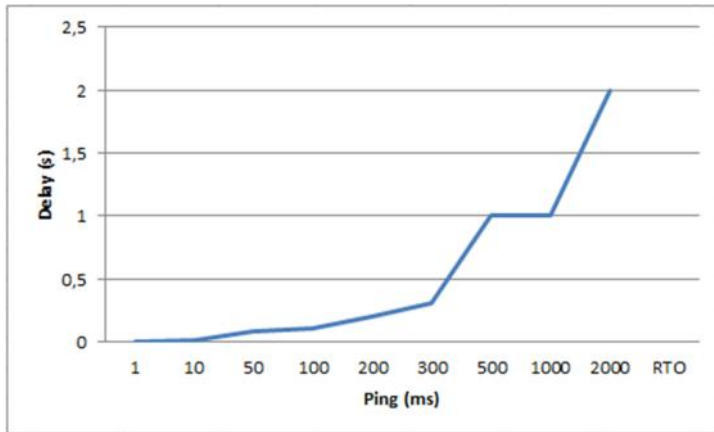
Dari data diatas dapat disimpulkan bahwa kendali jarak jauh dalam rancang bangun tugas akhir ini memiliki tingkat keberhasilan yang tinggi yaitu sebesar 100% pada jaringan internet lokal (*localhost*).

4.2. Pengujian Responsivitas Kendali Jarak Jauh

Pengujian yang dilakukan pada pengujian konektivitas kendali jarak jauh bertujuan untuk mengetahui responsibilitas sistem pada jaringan berbasis TCP/IP (*Transmission Control Protocol/Internet Protocol*). Pengujian pada responsivitas sistem ini dilakukan pada jaringan local pada router Tenda N301 dengan melakukan pengukuran perbedaan waktu pada saat perintah diberikan oleh pengguna melalui antarmuka *web-page* dengan respon yang diberikan oleh robot servis.

Tabel 4.2 Tabel pengujian responsivitas kendali jarak jauh

No.	Ping	Delay	Keterangan
1	1 ms	< 1s	Robot servis merespon permintaan pengguna
2	10 ms	< 1s	Robot servis merespon permintaan pengguna
3	50 ms	< 1s	Robot servis merespon permintaan pengguna
4	100 ms	< 1s	Robot servis merespon permintaan pengguna
5	200 ms	< 1s	Robot servis merespon permintaan pengguna
6	300 ms	< 1s	Robot servis merespon permintaan pengguna
7	500 ms	1s	Robot servis merespon permintaan pengguna
8	1000 ms	1s	Robot servis merespon permintaan pengguna
9	2000 ms	2s	Robot servis merespon permintaan pengguna
10	RTO	-	Robot servis tidak merespon permintaan pengguna



Gambar 4.1 Grafik hasil pengujian responsivitas kendali jarak jauh

Dari data diatas dapat disimpulkan bahwa keterlambatan respon sistem terhadap perintah yang diberikan oleh pengguna berbanding lurus dengan kenaikan PING (*Packet Internet Gopher*) pada jaringan TCP/IP. Dimana pengujian dilakukan sebanyak 4036 kali dengan paket loss sebesar 1%, ping terendah sebesar 2ms, ping terbesar sebesar 1160, dengan rata rata ping sebesar 24ms.

4.3. Pengujian *User-Guided Control*

Pengujian pada pengujian *user-guided control* bertujuan untuk mengetahui ketepatan fungsi pada *user-guided control*. Pengujian pada kendali jarak jauh yang dipandu oleh pengguna ini dilakukan dengan melakukan perhitungan kesalahan antara hasil yang diinginkan pada *user-guided control* dan hasil yang diperoleh saat pengujian. Pengujian *user-guided control* dibagi menjadi pengujian pada kendali *view-left*, pengujian pada kendali *keep forward*, dan pengujian pada kendali *view-right*.

4.3.1. Pengujian *View-Left*

Pengujian pada *user-guided control view-left* bertujuan untuk mengetahui ketepatan robot servis saat diberikan perintah *view-left* oleh pengguna, dimana robot akan berputar sebesar 360° ke kiri untuk memberikan tampilan citra lingkungan di sekitar robot servis dimulai dari bagian kiri robot. Selain ketepatan robot

servis dalam berputar 360° ke kiri, pengujian terhadap *user-guided control view-left* juga meliputi kendali robot, dimana robot servis dapat dihentikan oleh pengguna saat berputar dengan menekan kembali tombol *view-left* atau menekan tombol *clean* pada antarmuka *web-page*. Data-data hasil pengujian *view-left* dapat dilihat pada Tabel 4.3.

Dari data pada Tabel 4.3 dapat disimpulkan bahwa kendali jarak jauh *view-left* yang dipandu oleh pengguna memiliki ketidaktepatan hasil saat berputar dengan rata-rata sebesar 1.5°.

Tabel 4.3 Tabel pengujian *view-left*

No.	Perintah	Hasil	Error	Keterangan
1	<i>View-Left</i>	358°	2°	Robot dapat dihentikan saat sedang berputar
2	<i>View-Left</i>	359°	1°	Robot dapat dihentikan saat sedang berputar
3	<i>View-Left</i>	365°	5°	Robot dapat dihentikan saat sedang berputar
4	<i>View-Left</i>	360°	0°	Robot dapat dihentikan saat sedang berputar
5	<i>View-Left</i>	359°	1°	Robot dapat dihentikan saat sedang berputar
6	<i>View-Left</i>	358°	2°	Robot dapat dihentikan saat sedang berputar
7	<i>View-Left</i>	360°	0°	Robot dapat dihentikan saat sedang berputar
8	<i>View-Left</i>	358°	2°	Robot dapat dihentikan saat sedang berputar
9	<i>View-Left</i>	360°	0°	Robot dapat dihentikan saat sedang berputar
10	<i>View-Left</i>	358°	2°	Robot dapat dihentikan saat sedang berputar

4.3.2. Pengujian *Keep Forward*

Pengujian pada *user-guided control keep forward* bertujuan untuk mengetahui keandalan robot servis saat diberikan perintah *keep forward* oleh pengguna, dimana robot servis akan terus bergerak maju hingga pengguna menekan kembali tombol *keep forward* atau menekan tombol *clean* pada antarmuka *web-page*. Data-data hasil pengujian *keep forward* dapat dilihat pada Tabel 4.4.

Tabel 4.4 Tabel pengujian *keep forward*

No.	Perintah	Hasil	Keterangan
1	<i>Keep Forward</i>	✓	Robot bergerak maju dan dapat dihentikan oleh pengguna.
2	<i>Keep Forward</i>	✓	Robot bergerak maju dan dapat dihentikan oleh pengguna.
3	<i>Keep Forward</i>	✗	Robot bergerak maju setelah 3s dan dapat dihentikan oleh pengguna.
4	<i>Keep Forward</i>	✓	Robot bergerak maju dan dapat dihentikan oleh pengguna.
5	<i>Keep Forward</i>	✓	Robot bergerak maju dan dapat dihentikan oleh pengguna.
6	<i>Keep Forward</i>	✗	Robot bergerak maju setelah 3s dan dapat dihentikan oleh pengguna.
7	<i>Keep Forward</i>	✓	Robot bergerak maju dan dapat dihentikan oleh pengguna.
8	<i>Keep Forward</i>	✓	Robot bergerak maju dan dapat dihentikan oleh pengguna.
9	<i>Keep Forward</i>	✓	Robot bergerak maju dan dapat dihentikan oleh pengguna.
10	<i>Keep Forward</i>	✓	Robot bergerak maju dan dapat dihentikan oleh pengguna.

Dari data pada Tabel 4.4 dapat disimpulkan bahwa kendali jarak jauh *keep forward* yang dipandu oleh pengguna memiliki tingkat keandalan yang cukup tinggi yaitu sebesar 80%.

4.3.3. Pengujian *View-Right*

Pengujian pada *user-guided control view-right* bertujuan untuk mengetahui ketepatan robot servis saat diberikan perintah *view-right* oleh pengguna, dimana robot akan berputar sebesar 360° ke kanan untuk memberikan tampilan citra lingkungan di sekitar robot servis dimulai dari bagian kanan robot. Selain ketepatan robot servis dalam berputar 360° ke kanan, pengujian terhadap *user-guided control view-right* juga meliputi kendali robot, dimana robot servis dapat dihentikan oleh pengguna saat berputar dengan menekan tombol *view-right* atau tombol *clean* pada antarmuka *web-page*. Berikut adalah data-data hasil pengujian *view-right* :

Tabel 4.5 Tabel pengujian *view-right*

No.	Perintah	Hasil	Error	Keterangan
1	<i>View-Right</i>	360°	0°	Robot dapat dihentikan saat berputar
2	<i>View-Right</i>	360°	0°	Robot dapat dihentikan saat berputar
3	<i>View-Right</i>	365°	5°	Robot dapat dihentikan saat berputar
4	<i>View-Right</i>	358°	2°	Robot dapat dihentikan saat berputar
5	<i>View-Right</i>	360°	0°	Robot dapat dihentikan saat berputar
6	<i>View-Right</i>	363°	3°	Robot dapat dihentikan saat berputar
7	<i>View-Right</i>	359°	1°	Robot dapat dihentikan saat berputar
8	<i>View-Right</i>	360°	0°	Robot dapat dihentikan saat berputar
9	<i>View-Right</i>	362°	2°	Robot dapat dihentikan saat berputar
10	<i>View-Right</i>	360°	0°	Robot dapat dihentikan saat berputar

Dari data pada Tabel 4.5 dapat disimpulkan bahwa kendali jarak jauh *view-right* yang dipandu oleh pengguna memiliki ketidaktepatan hasil saat berputar dengan rata-rata sebesar 1.3°.

4.4. Pengujian Antarmuka *Web-page*

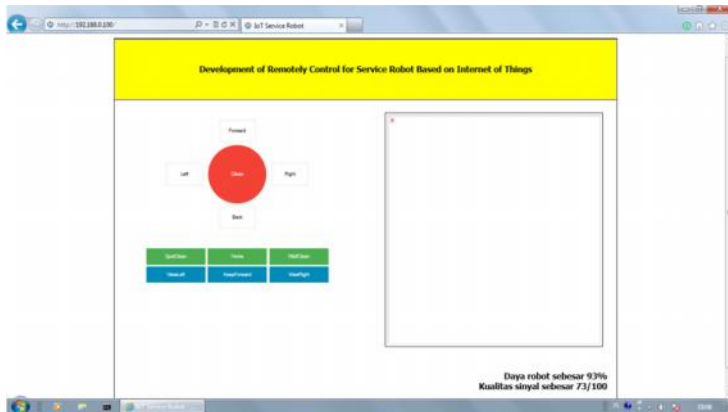
Pengujian pada pengujian antarmuka *web-page* bertujuan untuk mengetahui kompatibilitas antarmuka *web-page* pada berbagai *browser* dan peralatan berbasis internet. Pengujian pada antarmuka *web-page* dilakukan dengan mengakses halaman *web-page* pada peralatan dan *browser* yang diuji. Pengujian pada antarmuka *web-page* dibagi menjadi pengujian antarmuka *web-page* pada komputer dan pengujian antarmuka *web-page* pada *smartphone*.

4.4.1. Pengujian pada Komputer

Pengujian antarmuka *web-page* pada komputer bertujuan untuk mengetahui kompatibilitas antarmuka *web-page* pada berbagai *browser* di komputer. Pengujian dilakukan dengan mengakses halaman *web-page* pada komputer melalui berbagai *browser*. Berikut adalah data-data pengujian antarmuka *web-page* pada komputer:

Tabel 4.6 Tabel kompatibilitas antarmuka *web-page* pada berbagai *browser* di komputer

No.	<i>Browser</i>	Kompatibilitas	Keterangan
1	Internet Explorer	Tidak Kompatibel	Tampilan kamera pada antarmuka <i>web-page</i> tidak berfungsi dengan baik
2	Google Chrome	Kompatibel	Antarmuka <i>web-page</i> ditampilkan dengan baik oleh <i>browser</i>
3	Mozilla Firefox	Kompatibel	Antarmuka <i>web-page</i> ditampilkan dengan baik oleh <i>browser</i>
4	Opera	Kompatibel	Antarmuka <i>web-page</i> ditampilkan dengan baik oleh <i>browser</i>



Gambar 4.2 Tampilan antarmuka *web-page* pada Internet Explorer di komputer



Gambar 4.3 Tampilan antarmuka *web-page* pada Google Chrome di komputer



Gambar 4.4 Tampilan antarmuka *web-page* pada Mozzila Firefox di komputer



Gambar 4.5 Tampilan antarmuka *web-page* pada Opera di komputer

Dari data diatas dapat disimpulkan bahwa tampilan antarmuka *web-page* pada berbagai browser di komputer tidak sepenuhnya kompatibel. Hal ini dikarenakan Internet Explorer

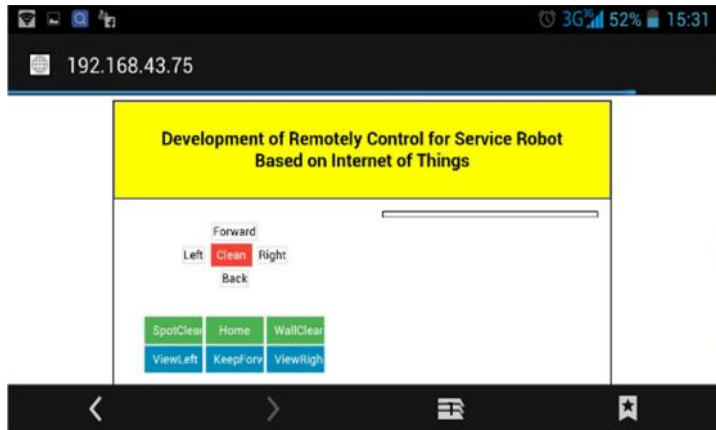
sebagai *default browser* di sistem operasi Microsoft tidak mendukung adanya layanan mjpeg yang digunakan untuk menampilkan tampilan kamera.

4.4.2. Pengujian pada *Smartphone*

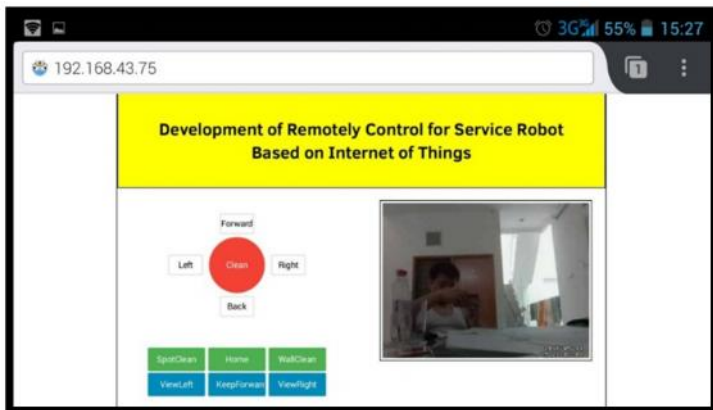
Pengujian antarmuka *web-page* pada *smartphone* bertujuan untuk mengetahui kompatibilitas antarmuka *web-page* pada berbagai *browser* di *smartphone*. Pengujian dilakukan dengan mengakses halaman *web-page* pada *smartphone* melalui berbagai *browser*. Berikut adalah data-data pengujian antarmuka *web-page* pada *smartphone*:

Tabel 4.7 Tabel kompatibilitas antarmuka *web-page* pada berbagai *browser* di *smartphone*

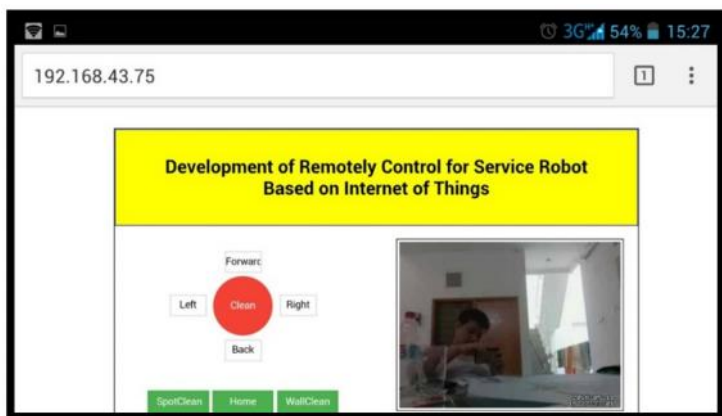
No.	Browser	Kompatibilitas	Keterangan
1	Mozilla Firefox	Kompatibel	Antarmuka <i>web-page</i> ditampilkan dengan baik oleh <i>browser</i>
2	Default Browser	Tidak Kompatibel	Tampilan kamera dan <i>styling</i> tombol tidak berfungsi dengan baik
3	Google Chrome	Kompatibel	Antarmuka <i>web-page</i> ditampilkan dengan baik oleh <i>browser</i>
4	Opera	Tidak Kompatibel	Tampilan kamera dan <i>styling</i> tombol pada antarmuka <i>web-page</i> tidak berfungsi dengan baik
5	UC Browser	Tidak Kompatibel	Tampilan kamera pada antarmuka <i>web-page</i> tidak berfungsi dengan baik
6	CM Browser	Tidak Kompatibel	Tampilan kamera dan <i>styling</i> tombol tidak berfungsi dengan baik



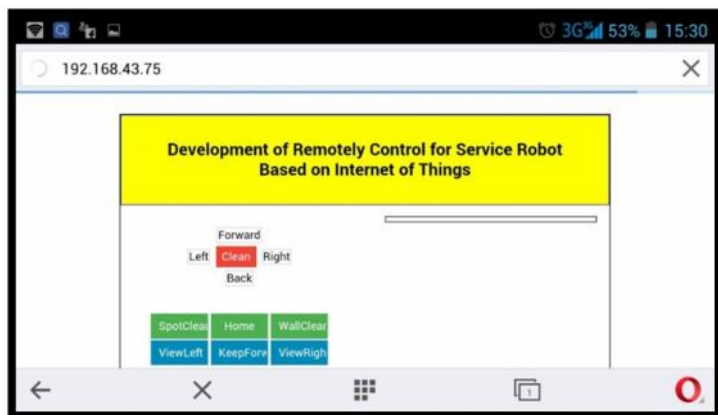
Gambar 4.6 Tampilan antarmuka *web-page* pada Default Browser di *smartphone*



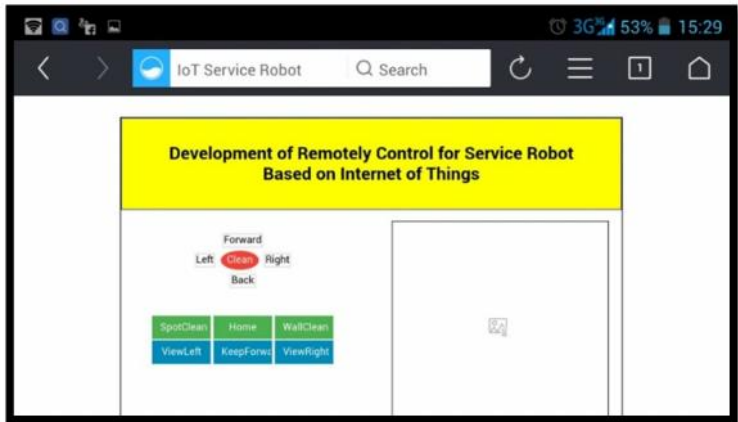
Gambar 4.7 Tampilan antarmuka *web-page* pada Mozilla Firefox di *smartphone*



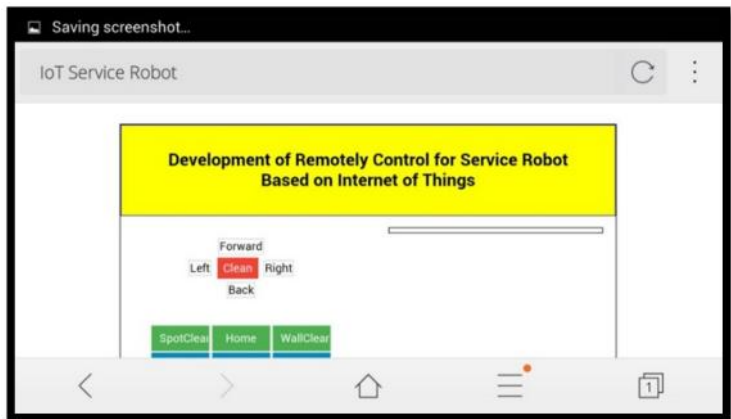
Gambar 4.8 Tampilan antarmuka *web-page* pada Google Chrome di *smartphone*



Gambar 4.9 Tampilan antarmuka *web-page* pada Opera di *smartphone*



Gambar 4.10 Tampilan antarmuka *web-page* pada UC Browser di *smartphone*



Gambar 4.11 Tampilan antarmuka *web-page* pada CM Browser di *smartphone*

Dari data diatas dapat disimpulkan bahwa tampilan antarmuka *web-page* pada berbagai browser di *smartphone* tidak sepenuhnya kompatibel. Hal ini dikarenakan beberapa *browser* seperti Default Browser, UC Browser, CM Browser, dan Opera tidak mendukung adanya layanan mjpeg dan beberapa *styling* pada css yang digunakan untuk menampilkan tampilan kamera dan memperhalus tampilan *web-page*.

LAMPIRAN

Berikut program pada Arduino Nano :

```
#include <IRremote.h>

IRsend irsend;
int command = 48;
int n = 0;

void setup() {
  Serial.begin(9600);
}

void loop() {
  /** serial code **/
  //if there is anything on the serial port,
  read it
  if (Serial.available() > 0) {
    //store it in the usbnumber variable
    command = Serial.read();
    Serial.println(command);
  }
  /** remote code **/
  if(command == 49){
    Serial.println("Clean");
    for (int i = 0; i < 3; i++) {
      irsend.sendNEC(0x2AA22DD, 32);
      delay(40);
    }
    command = 48;
  } else if (command == 50){
    Serial.println("Forward");
    for (int i = 0; i < 3; i++) {
      irsend.sendNEC(0x2AA55AA, 32);
      delay(40);
    }
    command = 48;
  } else if (command == 51){
```

```

Serial.println("Right");
for (int i = 0; i < 3; i++) {
  irsend.sendNEC(0x2AA44BB, 32);
  delay(40);
}
command = 48;
} else if (command == 52){
  Serial.println("Left");
  for (int i = 0; i < 3; i++) {
    irsend.sendNEC(0x2AA33CC, 32);
    delay(40);
  }
  command = 48;
} else if (command == 53){
  Serial.println("Backward");
  for (int i = 0; i < 3; i++) {
    irsend.sendNEC(0x2AA6699, 32);
    delay(40);
  }
  command = 48;
} else if (command == 54){
  Serial.println("Spot-Clean");
  for (int i = 0; i < 3; i++) {
    irsend.sendNEC(0x2AA7788, 32);
    delay(40);
  }
  command = 48;
} else if (command == 55){
  Serial.println("Home");
  for (int i = 0; i < 3; i++) {
    irsend.sendNEC(0x2AA8877, 32);
    delay(40);
  }
  command = 48;
} else if (command == 56){
  Serial.println("Wall-Clean");
  for (int i = 0; i < 3; i++) {
    irsend.sendNEC(0x2AA9966, 32);

```

```

        delay(40);
    }
    command = 48;
} else if (command == 113){
    Serial.println("View-Left");
    n=0;
    while(n<29){
        n++;
        command = Serial.read();
        if(command == 49) break;
        if(command == 113) break;
        Serial.println("Left");
        for (int i = 0; i < 3; i++) {
            irsend.sendNEC(0x2AA33CC, 32);
        }
    }
    command = 48;
} else if (command == 119){
    Serial.println("Keep-Forward");
    for (int i = 0; i < 3; i++) {
        irsend.sendNEC(0x2AA55AA, 32);
        delay(40);
    }
    delay(200);
    for (int i = 0; i < 3; i++) {
        irsend.sendNEC(0x2AA55AA, 32);
    }
    delay(100);
    while(1){
        command = Serial.read();
        if(command == 49) break;
        if(command == 119) break;
        Serial.println("Forward");
        for (int i = 0; i < 3; i++) {
            irsend.sendNEC(0x2AA55AA, 32);
            delay(40);
        }
        delay(3000);
    }
}

```

```

        for (int i = 0; i < 3; i++) {
            irsend.sendNEC(0x2AA55AA, 32);
        }
    }
    command = 48;
} else if (command == 101){
    Serial.println("View-Right");
    n=0;
    while(n<29){
        n++;
        command = Serial.read();
        if(command == 49) break;
        if(command == 101) break;
        Serial.println("Right");
        for (int i = 0; i < 3; i++) {
            irsend.sendNEC(0x2AA44BB, 32);
        }
    }
    command = 48;
}
}
}

```

Berikut program Python pada Raspberry Pi :

```

import sys
import subprocess
import serial
import MySQLdb
ser=serial.Serial('/dev/ttyUSB0',9600)

def main():
    interface = "wlan0"
    cells=[[[]]]
    sensor=[0]
    quality = ""
    signal = ""
    proc = subprocess.Popen(["iwconfig",
interface],stdout=subprocess.PIPE,
universal_newlines=True)

```



```

out, err = proc.communicate()

""" parse iwconfig out to cells """
for line in out.split("\n"):
    cells.append(line.rstrip())

""" get specific data to signal string """
quality = cells[6]
signal = quality[43:45]
print(signal)

ser.open()
read_serial = ser.readline()
sensor[0] =str(int(read_serial,16))
print(sensor[0])

db =
MySQLdb.connect("localhost","root","root","sensor_log")
cursor = db.cursor()
try:
    cursor.execute("INSERT INTO
`table_sensor_log`(`adc`,`signal`) VALUES
(%s, %s)",(str(sensor[0]),str(signal)))
    db.commit()
    print ('Ok')
except:
    db.rollback()
    print ('shit')
db.close()
while True:
    try:
        main()
    except:
        print ('something wrong')

```

Berikut program HTML pada Raspberry Pi :

```
<!doctype html>

<html lang="en">
<head>
    <meta charset="utf-8" />
    <title>IoT Service Robot</title>
    <link rel="shortcut icon" type="image/x-
icon" href="favicon.ico" />
    <link rel="stylesheet" href="main.css">
</head>
<body>
    <div id="big_wrapper">
        <header id="top_header">
            <h1> Development of Remotely Control
for Service Robot Based on Internet of Things
<h1>
        </header>
        <div id="content_wrapper">
            <form method="post"
action="index.php">
                <div id="button_input">
                    <br/>
                    <input class="dummy"
type="submit" value="Dummy" name="rcmd">
                    <input class="button_group0"
type="submit" value="Forward" name="rcmd">
                    <input class="dummy"
type="submit" value="Dummy" name="rcmd"><br/>
                    <input class="button_group0"
type="submit" value="Left" name="rcmd">
                    <input class="button"
type="submit" value="Clean" name="rcmd">
                    <input class="button_group0"
type="submit" value="Right" name="rcmd"><br/>
                    <input class="dummy"
type="submit" value="Dummy" name="rcmd">
```

```



```

Berikut program PHP pada Raspberry Pi :

```
<?php

$verz="1.0";
$comPort = "/dev/ttyUSB0";

if (isset($_POST["rcmd"])) {
$rcmd = $_POST["rcmd"];
switch ($rcmd) {
    case Clean:
        $fp =fopen($comPort, "w");
        fwrite($fp, 1);
        sleep(2);
        fclose($fp);
        break;
    case Forward:
        $fp =fopen($comPort, "w");
        fwrite($fp, 2);
        sleep(2);
        fclose($fp);
        break;
    case Right:
        $fp =fopen($comPort, "w");
        fwrite($fp, 3);
        sleep(2);
        fclose($fp);
        break;
    case Left:
        $fp =fopen($comPort, "w");
        fwrite($fp, 4);
        sleep(2);
        fclose($fp);
        break;
    case Back:
        $fp =fopen($comPort, "w");
        fwrite($fp, 5);
        sleep(2);
        fclose($fp);
    }
```

```

break;
case Home:
    $fp =fopen($comPort, "w");
    fwrite($fp, 7);
    sleep(2);
    fclose($fp);
break;
case WallClean:
    $fp =fopen($comPort, "w");
    fwrite($fp, 8);
    sleep(2);
    fclose($fp);
break;
case SpotClean:
    $fp =fopen($comPort, "w");
    fwrite($fp, 6);
    sleep(2);
    fclose($fp);
break;
case ViewLeft:
    $fp =fopen($comPort, "w");
    fwrite($fp, "q");
    sleep(2);
    fclose($fp);
break;
case KeepForward:
    $fp =fopen($comPort, "w");
    fwrite($fp, "w");
    sleep(2);
    fclose($fp);
break;
case ViewRight:
    $fp =fopen($comPort, "w");
    fwrite($fp, "e");
    sleep(2);
    fclose($fp);
break;
default:

```

```

        die('Crap, something went wrong. The
page just puked.');
```

```

    }
}
?>

<?php
$host="localhost";
$user="root";
$password='root';
$database='sensor_log';
$table='table_sensor_log';
$con=mysql_connect($host,$user,$password);
mysql_select_db($database);
$result=mysql_query("SELECT * FROM $table ORDER
BY ID DESC LIMIT 1");
while($row = mysql_fetch_assoc($result)){
    $stringDaya = $row['adc'];
    $stringSignal = $row['signal'];
    $printDaya = "Daya robot sebesar
$stringDaya%";
    $printSignal = "Kualitas sinyal sebesar
$stringSignal/100";
    print ($printDaya);
    echo '<br/>';
    print ($printSignal);
}
?>

```

Berikut program CSS pada Raspberry Pi :

```
.dummy{
    visibility          :hidden;
}
.button {
    background-color    : #f44336;
    width               : 8vw;
    height              : 8vw;
    padding             : 1% 2%;
    color               : white;
    border              : 2px solid #f44336;
    border-radius       : 50%;
    cursor              : pointer;
    transition-duration : 0.4s;
    float               : center;
    margin              : 1.3%;
}
.button:hover {
    background-color    : #de3428;
    color               : white;
}
.button_group0 {
    width               : 5vw;
    height              : 3vw;
    background          : url(../images/coba.png);
    *background-color   : #e7e7e7;
    border              : 2px solid #e7e7e7;
    color               : black;
    text-align          : center;
    text-decoration     : none;
    *display             : inline-block;
    font-size           : 80% Tohama;
    *transition-duration: 0.4s;
    cursor              : pointer;
    float               : center;
}
.button_group0:hover {
    border              : 2px solid #e7e7e7;
```

```

        *background-color    : white;
        color                 : black;
    }
    .button_group1 {
        width                 : 25%;
        height                : 80%;
        background-color      : #4CAF50;
        color                 : white;
        border                : 2px solid #4CAF50;
        padding               : 2% 3%;
        text-align            : center;
        text-decoration        : none;
        display               : inline-block;
        font-size             : 80% Tohama;
        transition-duration   : 0.4s;
        cursor                : pointer;
        float                 : center;
        margin-bottom         : 0.8%;
    }
    .button_group1:hover {
        background-color      : white;
        color                 : black;
    }
    .button_group2 {
        width                 : 25%;
        height                : 80%;
        background-color      : #008CBA;
        color                 : white;
        border                : 2px solid #008CBA;
        padding               : 2% 3%;
        text-align            : center;
        text-decoration        : none;
        display               : inline-block;
        font-size             : 80% Tohama Auto;
        transition-duration   : 0.4s;
        cursor                : pointer;
        float                 : center;
    }
}

```



```

.button_group2:hover {
    background-color : white;
    color            : black;
}
*{
    margin :0px;
    padding :0px;
}
h1{
    font :bold 150% Tahoma;
}
h2{
    font :bold 100% Tahoma;
}
h3{
    font :80% Tahoma;
}
body{
    text-align :center;
}
input[type=submit] {
    *width           :25%;
    *height          :18.5%;
    display          :inline;
    *margin-bottom   :0.8%;
}
img{
    border           :1px solid black;
    padding          :1%;
    margin           :1% auto;
    width            :90%;
    height           :width/0.75;
    max-height       :100%;
}
#big_wrapper{
    border           :1px solid black;
    width            :70%;

```

```

        height                :100%;
        margin                 :3% auto;
    }
    #content_wrapper{
        margin-top              :0%;
    }
    #top_header{
        background              :yellow;
        border                   :1px solid blue;
        padding                  :5%;
        text-align               :center;
    }
    #button_input{
        *border                  :1px solid black;
        float                    :left;
        padding                  :1%;
        margin                   :1% auto;
        width                    :47%;
        *height                  :350px;
        text-align               :center;
    }
    #camera{
        float                    :right;
        *border                   :1px solid black;
        padding                  :1%;
        margin                   :1% auto;
        width                    :47%;
        *height                  :200px;
        text-align               :center;
    }
    #bottom_footer{
        text-align               :right;
        clear                    :both;
        padding                  :2%;
    }
}

```

KESIMPULAN

Kesimpulan yang diperoleh pada tugas akhir ini adalah :

1. Berdasarkan data yang diperoleh dari pengujian sistem, dapat disimpulkan bahwa kendali jarak jauh berbasis *internet of things* memiliki tingkat kehandalan yang tinggi yaitu sebesar 100%. Dengan menggunakan spesifikasi sistem yang digunakan pada tugas akhir ini, robot servis dapat menjalankan seluruh perintah yang diinputkan oleh pengguna.
2. *User-guided control* sebagai kendali jarak jauh yang dipandu oleh pengguna memiliki ketepatan fungsi yang baik dimana kendali *view left* memiliki kesalahan sebesar 1.5° , kendali *keep forward* tidak memiliki tingkat keberhasilan sebesar 80%, dan kendali *view-right* memiliki kesalahan sebesar 1.3° .
3. Antarmuka *web-page* dari server yang diakses oleh pengguna melalui peralatan berbasis internet baik komputer maupun *smartphone* tidak seluruhnya kompatibel dengan berbagai layanan *browser*. Layanan *browser* yang kompatibel dengan tugas akhir ini adalah Google Chrome dan Mozilla Firefox yang dapat diakses melalui komputer maupun *smartphone*.

DAFTAR PUSTAKA

- [1] _____, "*Chuwi iLife v5 Robotic Vacuum Cleaner Module*", <URL:http://www.gearbest.com/smart-home/pp_217065.html>, 15th June 2016.
- [2] Nicole, "*Chuwi Ilife v5s Robotic Vacuum Cleaner*", <URL:<http://www.xiaomitoday.com/chuwi-ilife-v5s-robotic-vacuum-can-mop-floor/>>, 5th May 2016.
- [3] Alronzo, "*IR Communication*", <URL:<https://learn.sparkfun.com/tutorials/ir-communication>>, 15th June 2016.
- [4] Tasi, Janie, "*Technical Data Sheet 5mm Infrared LED*", Everlight Electronics Co., Ltd., Taiwan, 2005.
- [5] _____, "*Arduino Board Nano*", <URL: <https://www.arduino.cc/en/Main/ArduinoBoardNano>>, 15th June 2016.
- [6] Waher, Peter, "*Learning Internet of Thing*", Packt Publishing Ltd., Brimingham, 2015.
- [7] _____, "*Raspberry Pi on Sale Now at \$35*", <URL <https://www.raspberrypi.org/blog/raspberry-pi-2-on-sale/>>:, 2nd February 2015.
- [8] _____, "*Raspberry Pi Camera Module*", <URL <https://www.sparkfun.com/products/11868>>, 15th June 2016.
- [9] _____, "*Serial Communication*", <URL http://www.wikiwand.com/en/Serial_communication>, 15th June 2016.
- [10] _____, "*Wi-Fi Positioning System*", <URL https://en.wikipedia.org/wiki/Wi-Fi_positioning_system>, 15th June 2016.
- [11] _____, "*Tenergy 11.1 V 2200mAh Lithium Ion Battery Pack*", Tenergy Cooperation, California, 2009.
- [12] Nixon, Robin, "*Learning PHP, My SQL, JavaScript, CSS & HTML5*", O'Reilly Media, Inc., California, 2014.

BIODATA PENULIS



Adrie Sentosa, lahir di Denpasar, Bali pada tanggal 18 Agustus 1995. Anak pertama dari pasangan Tedjo Pranoto dan Lanny Trisnawati ini memulai pendidikan di TK Metodist, Medan. Setelah menyelesaikan pendidikan dasar di SD PL Bernardus, Semarang, penulis melanjutkan pendidikan menengah di SMP PL Domenico Savio, Semarang dan SMA Kolose Loyola, Semarang. Pada tahun 2012 penulis memulai pendidikan di Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember (ITS) Surabaya. Selama kuliah, penulis aktif membantu penyelenggaraan kegiatan akademik baik didalam maupun diluar kampus. Koordinator praktikum rangkaian listrik semester ganjil 2015/2016 dapat dihubungi melalui alamat email :adrie12@mhs.ee.its.ac.id.