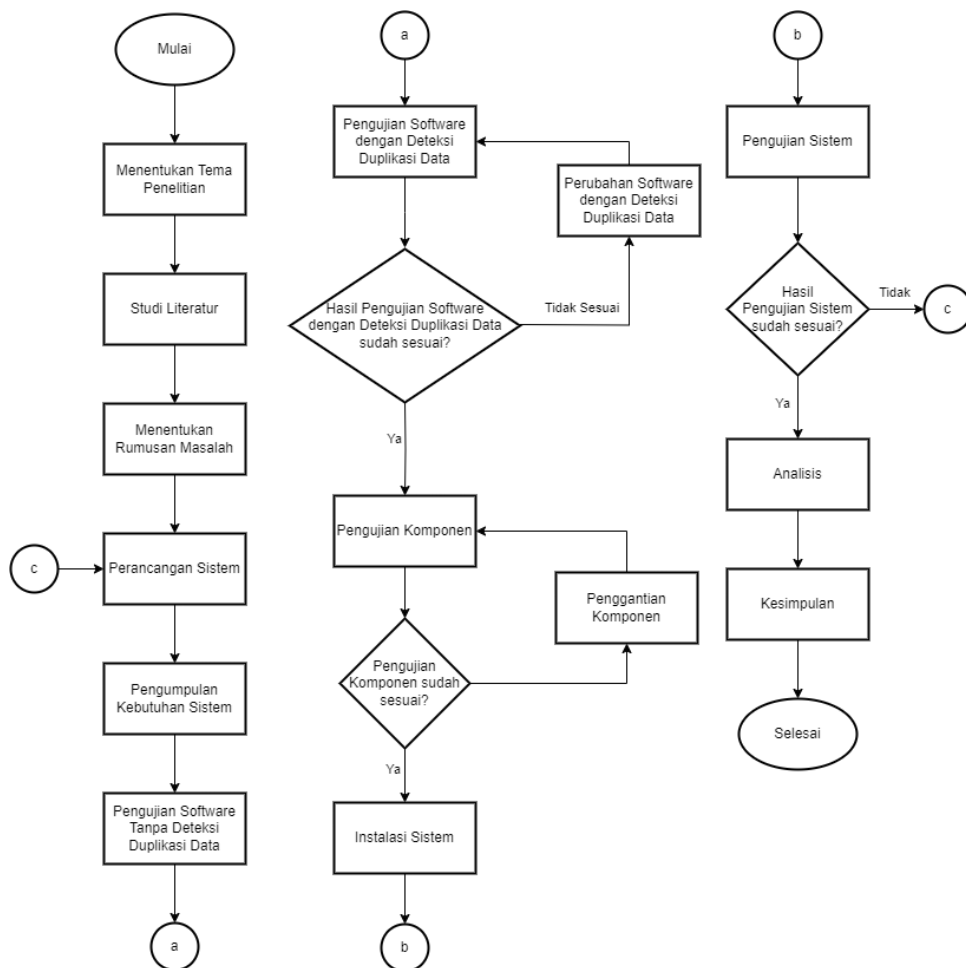


## BAB III METODE PENELITIAN

### 3.1. Flowchart Penelitian

Penelitian dibagi menjadi beberapa tahapan penelitian agar alur penelitian terorganisir dan diharapkan keluaran yang dihasilkan sesuai. Tahapan penelitian ditujukan dalam *flowchart* pada gambar 3.1.



Gambar 3.1 *Flowchart* Penelitian

#### 3.1.1. Menentukan Tema Penelitian

Menentukan tema penelitian merupakan tahapan penelitian untuk menentukan tema penelitian yang akan diteliti. Tema penelitian ditujukan agar

penelitian yang dilakukan dapat terfokus pada satu bahasan tema. Pada penelitian ini tema yang diambil mengenai pengiriman data mengikuti konsep *green* IoT.

### **3. 1. 2.Studi Literatur**

Studi literatur merupakan tahapan penelitian untuk mengumpulkan serta mengkaji informasi dan referensi mengenai tema penelitian. Studi literatur dilakukan untuk menentukan rumusan masalah yang akan diambil serta menentukan garis besar sistem yang akan dibangun. Informasi yang dikaji didapatkan dari jurnal penelitian, buku, serta sumber lainnya.

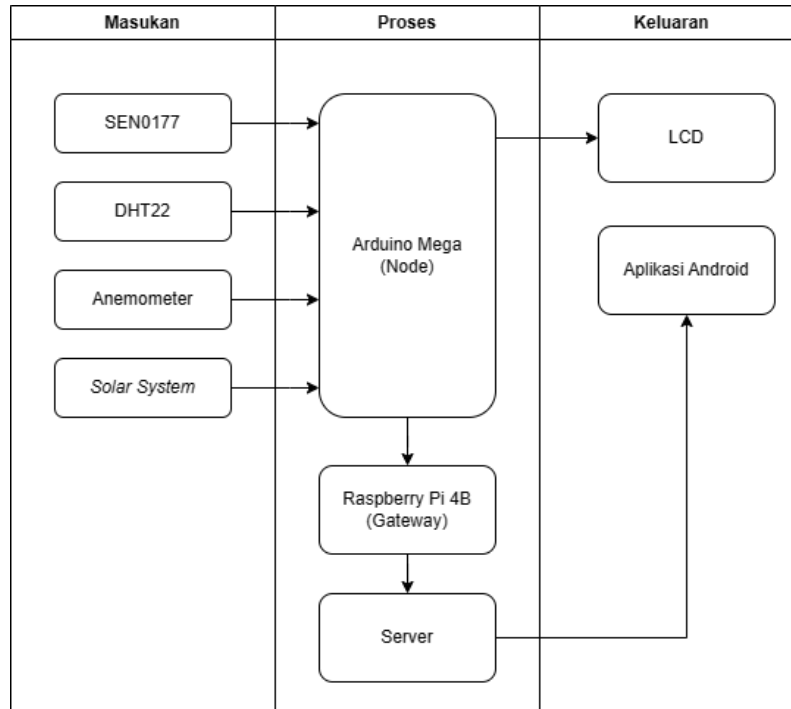
### **3. 1. 3.Menentukan Rumusan Masalah**

Menentukan rumusan masalah merupakan tahapan penelitian untuk menentukan masalah yang akan diangkat dalam penelitian. Penentuan rumusan masalah dilakukan untuk menentukan rancangan sistem yang akan dibangun.

### **3. 1. 4.Perancangan Sistem**

Perancangan sistem merupakan tahapan penelitian untuk merancang desain sistem yang akan dibangun diantaranya pembuatan alur kerja sistem serta arsitektur sistem. Perancangan sistem ditujukan sebagai pedoman dalam pembuatan sistem untuk meminimalisasi kesalahan kerja. Ketika pengujian sistem menghasilkan keluaran yang tidak sesuai dengan parameter yang diinginkan, maka penelitian kembali pada tahapan perancangan sistem untuk mengevaluasi rancangan sehingga sistem menjadi lebih baik dan harapannya keluaran yang dihasilkan sesuai dengan parameter yang dibutuhkan.

### 3. 1. 4. 1. Alur Kerja Sistem



Gambar 3. 2 Alur kerja sistem

Gambar 3.2 merupakan garis besar alur kerja dari sistem monitoring kualitas udara yang akan dikerjakan. Sistem terdiri dari tiga bagian yaitu masukan, proses, serta keluaran. SEN0177 merupakan sensor untuk pembacaan konsentrasi partikulat di udara (PM 1.0, PM 2.5, dan PM 10), DHT22 merupakan sensor pembacaan suhu dan kelembaban di udara, dan Anemometer merupakan sensor yang digunakan untuk pengukuran kecepatan angin. *Solar system* merupakan penyokong daya pada *node* dan terbagi atas solar panel dan MPPT. Sistem terinspirasi dari LoRaWAN yang menggunakan topologi star sehingga memiliki *node* dan *gateway* dengan penghubung diantara keduanya menggunakan jaringan LoRa. *Gateway* pada sistem meneruskan data hasil pembacaan sensor dari *node* menuju server. Aplikasi android merupakan keluaran dari sistem yang menampilkan data-data sensor yaitu data partikulat (PM2.5 dan PM10), suhu serta

kelembaban udara, dan kecepatan angin yang telah diterima dan disimpan oleh server dari *gateway*.

### 3. 1. 5. Pengumpulan Kebutuhan Sistem

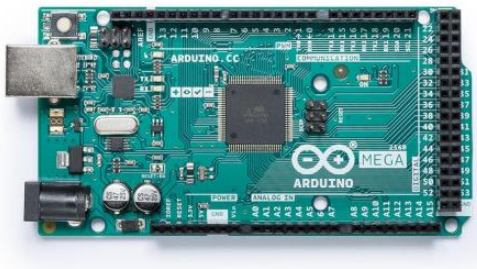
Pengumpulan kebutuhan sistem merupakan proses pengumpulan komponen yang dibutuhkan pada pembangunan sistem. Daftar komponen yang digunakan pada penelitian ini dapat dilihat pada Tabel 3.1 berikut.

Tabel 3. 1 Komponen Kebutuhan Sistem

No	Komponen	Kegunaan	Alasan Memilih Komponen
1	Arduino Mega	Sebagai node	Memiliki I/O lebih banyak daripada Arduino Uno.
2	Raspberry Pi 4B	Sebagai Gateway	Memiliki kemampuan komputasi seperti komputer namun dengan ukuran kecil dan harga terjangkau.
3	SEN0177	Sebagai sensor partikulat (PM2.5 dan PM10)	Dapat mendeteksi partikulat (PM1.0, PM2.5, PM10) dengan keakuratan baik.
4	DHT22	Sebagai sensor suhu dan kelembaban udara	Memiliki harga yang terjangkau.
5	SEN0170	Sebagai sensor kecepatan angin	Dapat mendeteksi kecepatan angin dengan tingkat presisi cukup baik.

No	Komponen	Kegunaan	Alasan Memilih Komponen
6	Panel Surya polycrystalline	Mengubah energi matahari menjadi listrik	Dapat menghasilkan daya walaupun dalam kondisi cahaya kurang sempurna.
7	Maximum Power Point Tracking (MPPT)	Sebagai <i>charger controller</i> sistem surya	Memiliki efisiensi yang lebih baik daripada metode PWM.
8	Baterai Sinus Electric SE20-12	Sebagai tempat penyimpanan energi cadangan untuk sistem	Sumber energi sesuai dengan kebutuhan sistem.
9	SX1262 868M Lora HAT	Sebagai perantara dalam pengiriman data dari <i>node</i> menuju <i>gateway</i>	Memiliki harga yang cukup terjangkau dengan jarak pengiriman maksimal 3km pada area terbuka.
10	Modul LM2596	Sebagai modul penurun tegangan	Mudah dalam penggunaannya.

### 3. 1. 5. 1. Arduino Mega



Gambar 3. 3. Tampak atas Arduino Mega

Sumber: (Arduino, 2009)

Gambar 3.3 merupakan tampak atas dari Arduino mega yang merupakan sebuah development board buatan Arduino. Arduino Mega bekerja pada tegangan 5V dengan menggunakan mikrokontroller Atmega2560 sebagai prosessornya dengan memori 256 KB ROM, 8 KB RAM dan 4 KB EEPROM sesuai pada Tabel 4.1 mengenai spesifikasi Arduino Mega.

Tabel 3. 2 Spesifikasi Arduino Mega

Pin I/O Digital	13
Pin Input Analog	16
Pin PWM	15
Komunikasi	UART (4), I2C (1), SPI (1)
Tegangan I/O	5V
Tegangan Input (Nominal)	7-12V
Arus DC Pin I/O	20mA
Processor	Atmega2560 16 MHz
Memori	8KB SRAM, 256KB SRAM, 4KB EEPROM

Sumber: (Arduino, 2009)

### 3. 1. 5. 2. Raspberry Pi 4 Model B



Gambar 3. 4 Tampak Atas Raspberry Pi 4 Model B

Gambar 3.4 merupakan tampak atas dari Raspberry Pi 4 Model B yang merupakan minikomputer buatan perusahaan Raspberry Pi (Trading) Ltd dimana merupakan penerus raspberry pi sebelumnya dengan peningkatan CPU, GPU, I/O, dan tetap memiliki bentukan dan harga yang sama dengan model sebelumnya

(Raspberry Pi (Trading) Ltd, 2019). Raspberry Pi 4 Model B menggunakan prosesor 64-bit quad core ARM-Cortex A72 1.5GHz dengan RAM 4GB LPDDR4. Terdapat 802.11 b/g/n/ac Wireless LAN untuk koneksi ke jaringan wifi dengan GPIO berjumlah 40 pin mendukung komunikasi UART, I2C, dan SPI. Spesifikasi lengkap Raspberry Pi 4 Model B terdapat pada Tabel 4.2.

Tabel 3. 3 Spesifikasi Raspberry Pi 4 Model B

Processor	Broadcom BCM2711, quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
Memori	4GB LPDDR4 dengan ECC
Konektivitas	<ul style="list-style-type: none"> <li>- 2.4 GHz and 5.0 GHz IEEE 802.11b/g/n/ac wireless LAN</li> <li>- Bluetooth 5.0, BLE</li> <li>- Gigabit Ethernet</li> <li>- 2 × USB 3.0 ports</li> <li>- 2 × USB 2.0 ports</li> </ul>
GPIO	40 dengan dukungan: <ul style="list-style-type: none"> <li>- 6 × UART</li> <li>- 6 × I2C</li> <li>- 5 × SPI</li> <li>- 1 × SDIO</li> <li>- 1 × DPI</li> <li>- 2 × Channel PWM</li> <li>- 3 × Keluaran GPCLK</li> </ul>
Daya Masukan	5VDC via USB-C (minimal 3A) 5VDC via GPIO header (minimal 3A) <i>Power over Ethernet (PoE)–enabled</i> <i>(requires separate PoE HAT)</i>
Suhu Operasi	0-50°C

Sumber: (Raspberry Pi (Trading) Ltd, 2023)

### 3. 1. 5. 3.Sensor SEN0177



Gambar 3. 5 Tampak atas sensor SEN0177

Sumber: (DFRobot, 2016)

Gambar 3.5 merupakan unit sensor SEN0177 buatan DFRobot dengan kemampuan mengukur konsentrasi partikel di udara dengan ukuran berkisar 0.3 – 10 mikron. SEN0177 bekerja pada tegangan 4.95 ~ 5.05 V dengan response time  $\leq$  10 ms serta mendukung pengukuran PM 1.0, PM2.5, dan PM10. Spesifikasi lengkapnya terdapat pada Tabel 3.4.

Sensor SEN0177 bekerja dengan membaca persebaran cahaya dari laser pada sudut tertentu. Hasil pembacaan ini kemudian diolah dengan menggunakan transformasi fourier dan sederet perhitungan kompleks lainnya sehingga menghasilkan data PM 1.0, PM 2.5 dan PM 10. Berdasarkan penelitian (Migos et al., 2019), SEN0177 memiliki hasil pengukuran yang lebih kredibel dan sesuai dengan pengukuran menggunakan perangkat pengukuran PM Dylos DC1100 yang telah beredar di pasaran.

Tabel 3. 4 Spesifikasi Sensor SEN0177

Tegangan Operasi	4.95-5.05V DC
Arus Maksimal	120mA



Diameter PM Terukur	- 0.3-1.0 $\mu\text{m}$ - 1.0-2.5 $\mu\text{m}$ - 2.5-10 $\mu\text{m}$
Range PM Terukur	0-999 $\mu\text{g}/\text{m}^3$
Response Time	$\leq 10$ detik
Suhu Operasi	-20-50 $^{\circ}\text{C}$
Kelembaban Operasi	0-99% RH

Sumber: (DFRobot, 2016)

### 3. 1. 5. 4. Sensor Kecepatan Angin



Gambar 3. 6 Sensor Kecepatan Angin

Sumber: (DFRobot, 2020)

Gambar 3.6 merupakan unit sensor untuk pembacaan kecepatan angin yang berjalan pada tegangan 9-24V dengan range pengukuran 0-30m/s, resolusi 0.1m/s, serta error rate  $\pm 3\%$  sesuai dengan spesifikasi pada Tabel 4.4. Sensor ini bekerja dengan memanfaatkan putaran dari ketiga mangkok yang berada di posisi paling atas sensor. Perputaran ketiga mangkok kemudian diterjemahkan dengan bantuan generator sinyal sehingga menghasilkan hasil pembacaan sensor (Suwarti, Mulyono and Prasetyo, 2017). Hasil pembacaan sensor dikirimkan dalam bentuk analog oleh sensor dan diterima oleh I/O analog mikrokontroler.

Tabel 3. 5 Spesifikasi Sensor Kecepatan Angin

Jenis	Anemometer 3 Cup
Material	Aluminium Alloy
Sinyal Output	0-5V
Tegangan Masukan	9-24V DC
Resolusi	0.1m/s
Error Sistem	±3%
Range pengukuran efisien	0-30m/s
Suhu Operasi	-40-80 °C
Kelembaban Operasi	35-85% (Tanpa Kondensasi)

Sumber: (DFRobot, 2020)

### 3. 1. 5. 5.Sensor DHT22



Gambar 3. 7 Sensor DHT22

Sumber: (DFRobot, 2017)

Gambar 3.7 menampilkan unit DHT22 yang merupakan sensor pengukur kelembaban udara serta suhu relatif dengan *range* pengukuran -40 ~ 80 °C untuk suhu relatif dan 0 – 100% untuk kelembaban udara serta akurasi ±2%RH(maksimal ±5%RH) untuk kelembaban udara dan <math>\leq \pm 0.5^{\circ}\text{C}</math> untuk suhu relatif (Liu, 2013). Spesifikasi lengkap sensor DHT22 terdapat pada Tabel 4.5. Sensor DHT22 melakukan pengukuran kelembaban dengan menggunakan sensor kelembaban kapasitif dan pengukuran suhu menggunakan thermistor dengan tipe NTC (Negative Temperature Coefficient) (DFRobot, 2017). Hasil pembacaan dari

keduanya kemudian diproses oleh mikrokontroler 8-bit dan menghasilkan hasil pembacaan suhu dan kelembaban yang dapat diterima oleh mikrokontroler seperti arduino mega menggunakan pin digital.

Tabel 3. 6 Spesifikasi Sensor DHT22

Tegangan Masukan	5V DC
Tegangan Keluaran	0-3.3 V DC
<i>Range</i> Suhu Terukur	-40-80 °C
Resolusi Pengukuran Suhu	0.1 °C
Error Sistem Pengukuran Suhu	<math>\pm 0.5^{\circ}\text{C}</math>
<i>Range</i> Kelembaban Terukur	0-100% RH
Resolusi Pengukuran Kelembaban	0.1% RH
Error Sistem Pengukuran Kelembaban	$\pm 2\%$ RH

Sumber: (DFRobot, 2017)

### 3. 1. 5. 6. SX1262 868M Lora HAT



Gambar 3. 8 Modul SX1262 868M LoRa HAT Waveshare

Sumber: (Waveshare, 2019)

SX1262 868M LoRa HAT adalah modul LoRa buatan perusahaan Waveshare. Modul LoRa ini ditujukan untuk Raspberry Pi, namun dengan penggunaan komunikasi UART membuat modul LoRa ini bisa juga digunakan oleh

Arduino dengan menggunakan rx/tx. Modul LoRa dari Waveshare ini menggunakan chip SX1268/SX1261 dengan protokol komunikasi private sehingga tidak mendukung LoRaWAN (Waveshare, 2019).

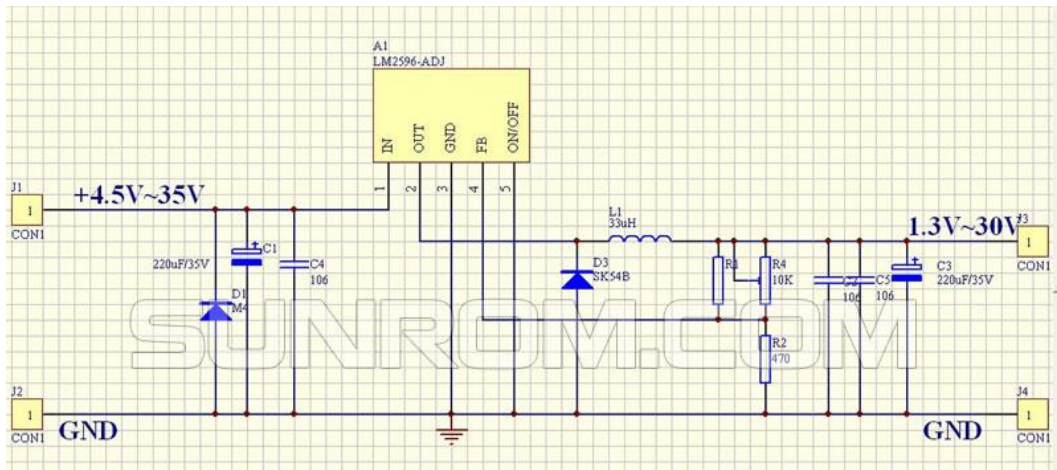
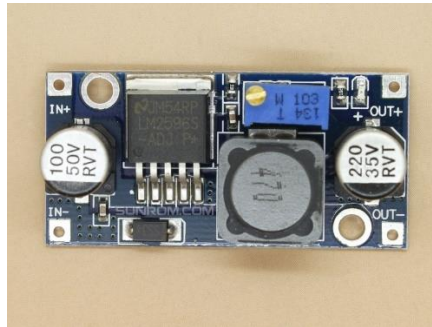
Modul ini dapat berjalan pada frekuensi 433/470/868/915 MHz yang merupakan frekuensi ISM tanpa lisensi. Dalam kondisi ideal, jarak komunikasi dapat mencapai jarak 5 km. Tegangan kerja modul sebesar 5V dengan arus transien saat pengiriman sebesar 100 mA dan arus saat menerima data sebesar 11 mA. Panjang data maksimal dalam pengiriman data adalah sebesar 240 Bytes dengan maksimal power pengiriman 22.0 dBm. Spesifikasi lengkap modul ini terdapat pada Tabel 4.6.

Tabel 3. 7 Spesifikasi SX1262 868M Lora HAT

Kebutuhan Arus Pengiriman	100 mA
Kebutuhan Arus Penerimaan	11 mA
Maksimal Daya Pengiriman	22.0 dBm (10, 13, 17, 22 dBm <i>Selectable</i> )
Panjang Data Pengiriman	240 Bytes (32, 64, 128, 240 Bytes <i>Selectable</i> )
<i>Working Bands</i>	410.125~493.125MHz or 850.125~930.125MHz
Sensitivitas Penerimaan	-147dBm@0.3Kbps ( <i>On air</i> )
<i>Interface</i>	UART
Tegangan Kerja	5 V DC
Suhu Operasi	-40-80 °C

Sumber: (Waveshare, 2019)

### 3. 1. 5. 7. Modul LM2596



Gambar 3. 9 (a) Tampak atas Modul LM2596 (b) Skematik Rangkaian Modul LM2596

Sumber: (Sunrom Electronics, 2014)

Modul LM2596 merupakan penurun tegangan DC menggunakan bantuan IC LM2596. Komponen kapasitor pada masukan dan keluaran modul berfungsi sebagai filter arus dan tegangan, dioda sebagai perlindungan arus terbalik, induktor sebagai penyimpan arus, dan resistor sebagai pengatur keluaran. Modul ini dapat menurunkan tegangan mulai dari 1.2V-37V dengan masukan 4.5V-40V dan arus yang mengalir maksimal 3A. Spesifikasi modul LM2596 terdapat pada Tabel 4.7. Gambar 3.9 (a) merupakan tampak atas dari modul LM2596 dengan rangkaian pada Gambar 3.9 (b).

Tabel 3. 8 Spesifikasi Modul LM2596

Tegangan Masukan	4.5-35V
Tegangan Keluaran	1.3-30V
Arus Keluaran	Maksimal 3A
Efisiensi Konversi Maksimal	92%
Frekuensi <i>Switching</i>	150 KHz
<i>Ripple</i> Keluaran	50mV (maksimal) 20M-bandwidth
Suhu Operasi	-40 hingga +85 °C

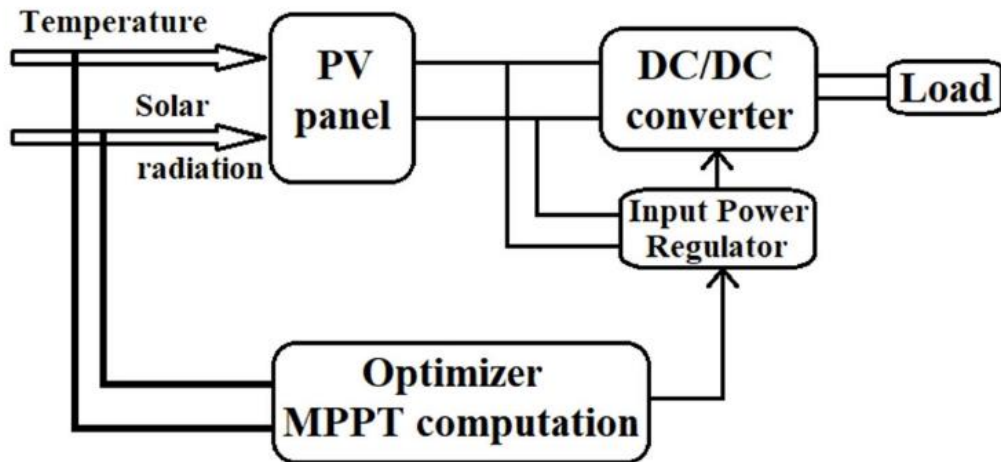
Sumber: (Sunrom Electronics, 2014)

### 3. 1. 5. 8. Maximum Power Point Tracking (MPPT)



Gambar 3. 10 Tampak Depan MPPT

Gambar 3.10 merupakan tampak depan unit MPPT yang digunakan pada penelitian ini. Secara sederhana MPPT bekerja dengan mengkalkulasikan nilai optimal dimana nilai optimal ini kemudian dikirimkan menuju konverter DC/DC sebagai nilai referensi (Hamidi *et al.*, 2020). Alur kerja tersebut digambarkan dalam Gambar 3.11. MPPT yang digunakan dapat menerima maksimum tegangan dari panel surya sebesar 50V. MPPT mendukung baterai 12 V dan 24 V dengan arus maksimum 50A.



Gambar 3. 11 Arsitektur Sederhana MPPT

Sumber: (Hamidi et al., 2020)

### 3. 1. 5. 9. Panel Surya Polycrystalline



Gambar 3. 12 Panel Surya Polysrystalline

Panel surya yang digunakan adalah panel surya polycrystalline seperti pada gambar 3.12. Panel surya ini memiliki peak power 50 W dengan maksimal tegangan 17.8 V, maksimal arus 2.81 A serta efisiensi sel 16.93 %. Toleransi daya pada panel surya ini adalah +- 3 %.

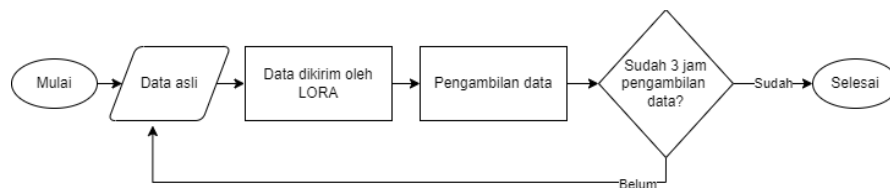
### 3. 1. 5. 10. Baterai 12 V



Gambar 3. 13 Baterai 12V

Gambar 3.13 merupakan bentuk unit baterai yang digunakan pada penelitian ini. Baterai merupakan buatan dari perusahaan Sinus Electric yang mampu mengalirkan tegangan 12 V dengan kapasitas maksimal 20Ah. Baterai ini merupakan baterai rechargeable yang dapat dilakukan pengisian ulang dan berjenis sealed lead acid.

### 3. 1. 6. Pengujian *Software* Tanpa Deteksi Duplikasi Data



Gambar 3. 14 Flowchart pengujian *software* tanpa deteksi duplikasi data

Gambar 3.14 merupakan flowchart dari pengujian *software* tanpa deteksi duplikasi data. Pengujian *software* tanpa deteksi duplikasi data adalah pengujian program sistem yang didalamnya tidak memiliki metode deteksi duplikasi data sehingga data yang dikirimkan oleh *node* adalah data asli tanpa ada perubahan akibat penggunaan metode deteksi duplikasi data. Pengujian dilakukan dalam durasi tiga jam dengan data yang diambil berupa data mengenai akurasi data sistem,

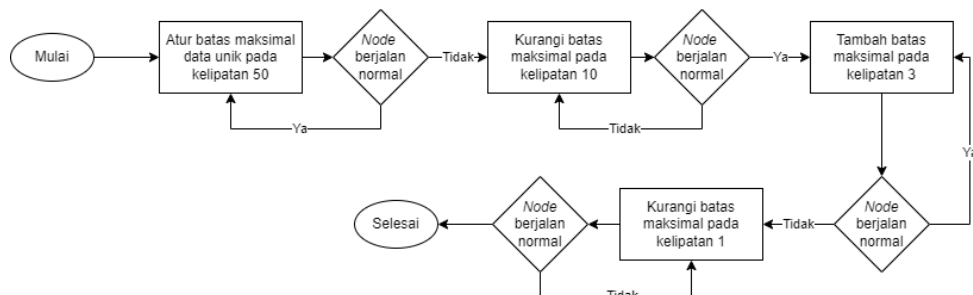


serta daya listrik yang dibutuhkan saat pengiriman data oleh modul LoRa yang terjadi diakibatkan oleh penggunaan sistem dengan *software* tanpa deteksi duplikasi data.

### 3. 1. 7. Pengujian *Software* dengan Deteksi Duplikasi Data

Pengujian *software* dengan deteksi duplikasi data merupakan pengujian untuk mengetahui akurasi pengiriman data serta daya listrik yang dibutuhkan oleh *node* ketika pada sistem diterapkan metode deteksi duplikasi data. Pengujian terbagi menjadi pengujian dengan data sensor acak, pengujian dengan data sensor selalu sama, pengujian batas maksimal data unik pada *node*, serta pengujian apabila jumlah data unik melebihi batas yang telah ditentukan.

#### 3. 1. 7. 1. Pengujian Batas Maksimal Data Unik pada *Node*

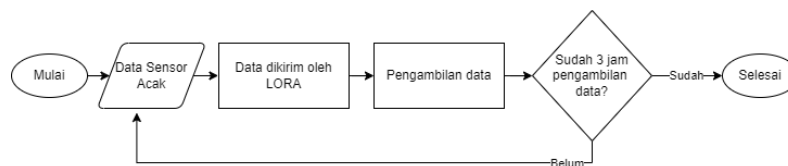


Gambar 3.15 Alur pengujian *software* dengan deteksi duplikasi data bagian batas maksimal data unik tersimpan pada *node*

Gambar 3.15 merupakan alur pengujian *software* untuk mengetahui batas data unik yang dapat disimpan pada *node* sebagai data pembandingan bagi data-data pembacaan terbaru. Pengujian dilakukan dengan mendambahkan dan mengurangi nilai batas pada kelipatan 50 untuk mengetahui gambaran kasar mengenai nilai maksimal batasnya. Setelah *node* tidak dapat berjalan dengan normal, maka nilai maksimal dikurangi pada kelipatan 10 untuk kemudian ditambahkan pada kelipatan 3. Apabila pada saat penambahan nilai maksimal pada kelipatan 3 didapatkan

bahwa *node* sudah tidak dapat berjalan normal maka nilai kelipatan kemudian dikurangi pada kelipatan 1 untuk mengetahui nilai tepat dimana *node* dapat menyimpan semua data unik tanpa mengalami kegagalan fungsi. Nilai tepat yang didapatkan pada saat pengurangan nilai pada kelipatan 1 adalah nilai maksimal dimana *node* dapat menyimpan data unik. Nilai tersebut kemudian menjadi nilai pembatas jumlah data unik yang dapat disimpan pada *node*.

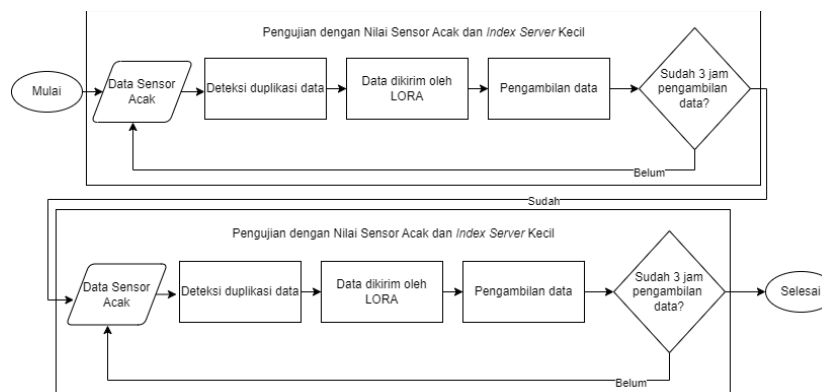
### 3.1.7.2. Pengujian Data Unik Melebihi Batas Maksimal



Gambar 3.16 Pengujian *software* dengan deteksi duplikasi data dan data unik melebihi batas maksimal

Gambar 3.16 merupakan alur pengujian *software* dengan deteksi duplikasi data dimana data unik yang dihasilkan lebih dari batas data unik yang telah ditentukan pada program *node*. Pada pengujian ini, batas data unik pada program diatur pada nilai 10 data unik. Pengujian dilakukan pada durasi 3 jam dengan data sensor acak.

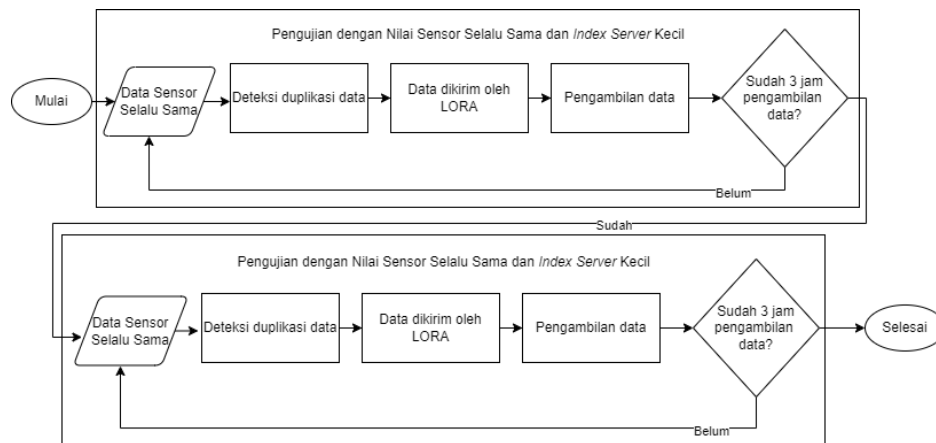
### 3.1.7.3. Pengujian Nilai Sensor Acak



Gambar 3.17 Diagram alur pengujian *software* dengan deteksi duplikasi data dengan nilai sensor acak

Gambar 3.17 adalah alur pengujian *software* dengan deteksi duplikasi data pada bagian pengujian dengan nilai sensor acak. Pengujian dilakukan pada dua skema dimana skema pertama data *index server* yang diberikan oleh *server* memiliki ukuran data kecil dan skema kedua data *index server* yang diberikan oleh *server* memiliki ukuran besar. Durasi pengujian untuk masing-masing skema adalah tiga jam sehingga apabila dilakukan akumulasi, durasi pengujian total selama enam jam.

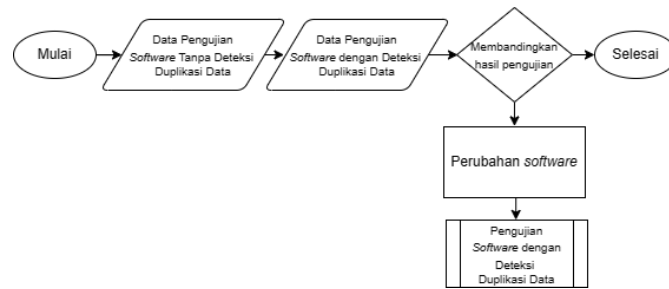
#### 3.1.7.4. Pengujian Nilai Sensor Selalu Sama



Gambar 3.18 Diagram alur pengujian *software* dengan deteksi duplikasi data dengan nilai sensor selalu sama

Gambar 3.18 merupakan alur pengujian *software* dengan deteksi duplikasi data pada pengujian dengan nilai sensor yang selalu sama. Pengujian dilakukan pada dua skema yaitu pada data *index server* berukuran kecil dan pada data *index server* berukuran besar. Pengujian dilakukan dengan durasi enam jam dimana masing-masing skema diberikan durasi selama tiga jam.

### 3. 1. 8. Perubahan *Software* dengan Deteksi Duplikasi Data



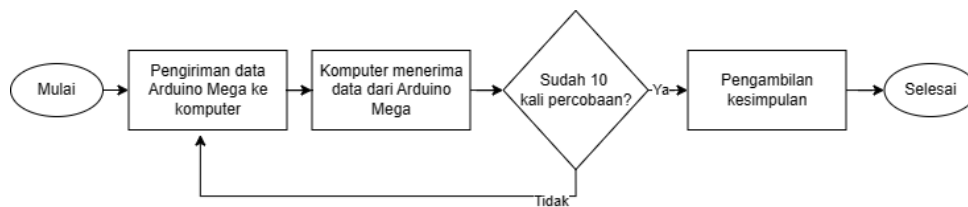
Gambar 3. 19 Flowchart perubahan *software* dengan deteksi duplikasi data

Gambar 3.19 merupakan *flowchart* perubahan *software* dengan deteksi duplikasi data. Setelah didapatkan hasil dari pengujian *software* dengan deteksi duplikasi data, data hasil pengujian tersebut kemudian dibandingkan dengan hasil pengujian *software* tanpa deteksi duplikasi data. Apabila hasil dari perbandingan tersebut telah sesuai dengan parameter yang diinginkan maka pengujian dapat dilakukan menuju tahap selanjutnya. Apabila hasil perbandingan tersebut tidak sesuai maka harus dilakukan perubahan terhadap *software* dengan deteksi duplikasi data. Perubahan *software* dengan deteksi duplikasi data dilakukan dengan harapan bahwa keluaran yang dihasilkan memiliki nilai yang lebih baik daripada *software* tanpa deteksi duplikasi data. Perubahan yang telah dilakukan terhadap *software* dengan deteksi duplikasi data kemudian dilakukan pengujian ulang untuk kemudian dibandingkan kembali dengan hasil dari pengujian *software* tanpa deteksi duplikasi data, apabila hasil perbandingan didapatkan bahwa *software* dengan deteksi duplikasi data lebih baik daripada *software* tanpa deteksi duplikasi data maka pengujian diteruskan pada pengujian selanjutnya dan apabila sebaliknya maka kembali melakukan perubahan *software* untuk mendapatkan hasil yang lebih baik.

### 3. 1. 9. Pengujian Komponen

Pengujian komponen merupakan proses pengujian untuk setiap komponen pada sistem yang akan dibangun. Pengujian dilakukan untuk memastikan komponen bekerja dengan baik. Pengujian komponen terbagi menjadi beberapa bagian sebagai berikut:

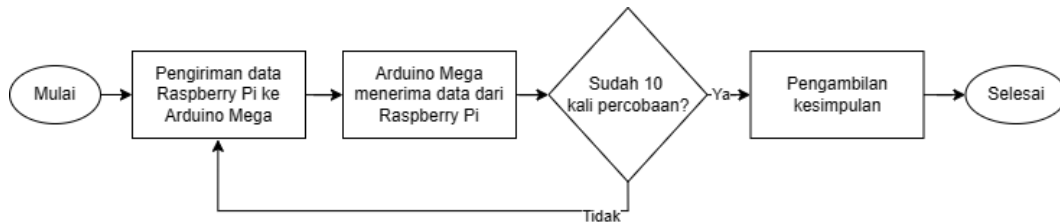
#### 3. 1. 9. 1. Pengujian Arduino Mega



Gambar 3.20 Flowchart pengujian Arduino Mega

Gambar 3.20 merupakan flowchart dari pengujian Arduino Mega. Pengujian Arduino Mega ditujukan untuk mengetahui kondisi Arduino Mega yang akan digunakan dalam sistem apakah dapat berjalan dengan baik mengikuti program yang telah ditanam didalamnya atau tidak. Arduino Mega melakukan pengiriman data dengan menggunakan UART menuju komputer. Data yang dikirimkan oleh Arduino Mega adalah data angka yang akan terus bertambah seiring pengiriman data. Data yang dikirimkan oleh Arduino Mega diterima oleh komputer dan ditampilkan dalam fitur serial monitor milik Arduino IDE. Apabila data yang diterima oleh komputer nilainya terus bertambah sebagai contoh dari 1 menuju 2 dan kemudian 3, maka Arduino Mega dapat disimpulkan berfungsi dengan baik. Pengujian dilakukan dalam 10 kali percobaan.

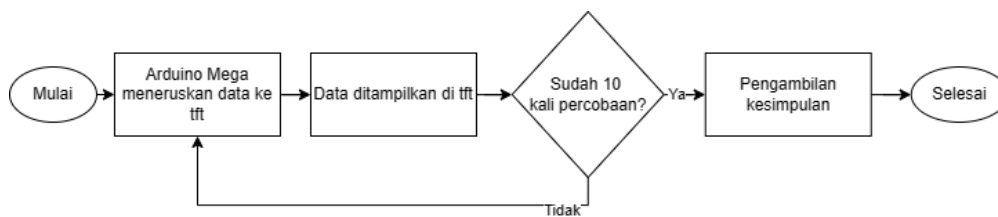
### 3. 1. 9. 2. Pengujian Raspberry Pi 4 Model B



Gambar 3. 21 Flowchart pengujian Raspberry Pi 4 Model B

Gambar 3.21 merupakan flowchart pengujian Raspberry Pi. Pengujian Raspberry Pi 4 Model B dilakukan untuk mengetahui apakah unit Raspberry Pi 4 Model B yang digunakan dapat berfungsi dengan baik atau tidak. Pengujian dilakukan dengan melakukan pengiriman data melalui UART dari Arduino Mega menuju Raspberry Pi 4 Model B. Data yang dikirimkan dari Arduino Mega adalah data iterasi angka yang terus bertambah dari 1 hingga mencapai 10. Apabila data yang diterima oleh Raspberry Pi 4 Model B untuk pertama kali adalah 1 kemudian disusul oleh angka 2 hingga angka 10 maka unit dalam kondisi baik.

### 3. 1. 9. 3. Pengujian Layar TFT

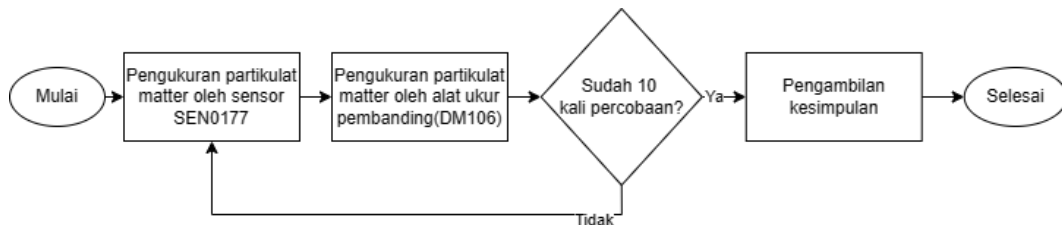


Gambar 3. 22 Flowchart pengujian layar TFT

Gambar 3.22 merupakan flowchart dari pengujian layar TFT. Pengujian layar TFT dilakukan untuk mengetahui kondisi layar TFT yang digunakan pada sistem apakah dapat menampilkan data sesuai yang diinstruksikan oleh Arduino Mega atau tidak. Pengujian dilakukan dengan menginstruksikan layar tft untuk menampilkan data iterasi dari 1 hingga 10 dari Arduino Mega. Data iterasi tersebut

akan dimulai dari 1 dan terus bertambah hingga data yang ditampilkan pada layar tft menjadi 10. Apabila data yang ditampilkan oleh layar tft adalah terus berganti dari 1 menjadi 2 hingga 10 maka layar tft dalam kondisi baik.

### 3. 1. 9. 4. Pengujian Sensor SEN0177



Gambar 3. 23 Flowchart pengujian sensor SEN0177

Gambar 3.23 merupakan flowchart pengujian sensor SEN0177. Pengujian sensor SEN0177 merupakan pengujian sensor SEN0177 sebagai sensor partikulat untuk mengetahui akurasi sensor terhadap alat ukur pembandingan. Pengukuran dilakukan pada 10 tempat yang berbeda sehingga hasil pembacaan partikulat bervariasi mengikuti konsentrasi partikulat pada 10 tempat tersebut. Apabila nilai error dari hasil pembandingan antara sensor dengan pembandingan adalah kurang dari 10% maka dapat disimpulkan bahwa sensor yang digunakan dalam kondisi baik dan dapat digunakan dalam sistem monitoring kualitas udara.

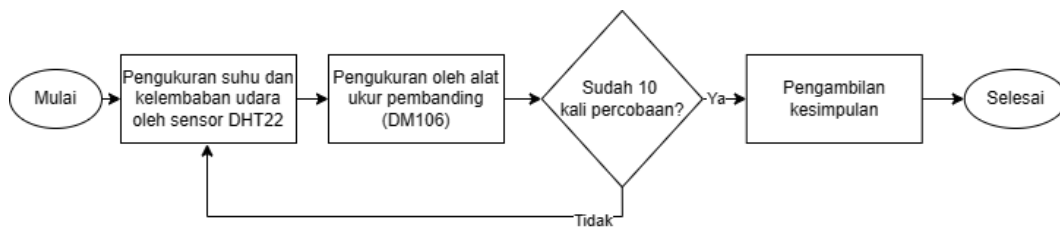
### 3. 1. 9. 5. Pengujian Sensor Kecepatan Angin



Gambar 3. 24 Flowchart pengujian sensor kecepatan angin

Gambar 3.24 adalah flowchart pengujian sensor kecepatan angin. Pengujian sensor kecepatan angin dilakukan untuk mengetahui akurasi sensor kecepatan angin yang digunakan terhadap alat pengukuran kecepatan angin pembanding yang telah terstandarisasi. Pengukuran dilakukan dengan membandingkan hasil pengukuran dari sensor kecepatan angin yang digunakan dalam penelitian dan hasil pengukuran dari alat pengukuran kecepatan angin pembanding yang telah terstandarisasi. Apabila nilai error dari hasil pembandingan antara sensor dengan alat pembanding adalah kurang dari 10% maka dapat disimpulkan bahwa sensor dalam kondisi baik.

### 3. 1. 9. 6. Pengujian Sensor DHT22

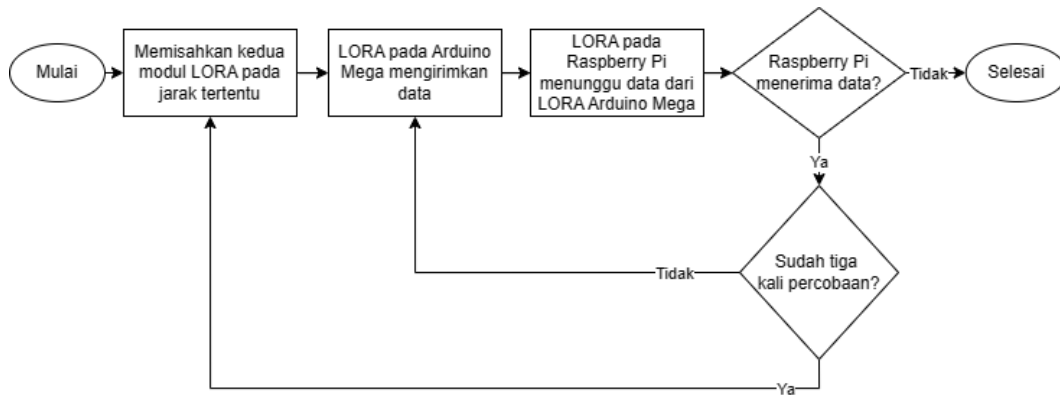


Gambar 3. 25 Flowchart pengujian sensor DHT22

Gambar 3.25 merupakan *flowchart* pengujian sensor DHT22. Pengujian sensor DHT22 dilakukan untuk mengetahui akurasi pembacaan suhu dan kelembaban udara dari sensor DHT22 dengan hasil pembacaan alat pengukuran suhu dan kelembaban pembanding. Pengujian dilakukan dengan membandingkan hasil pembacaan dari sensor DHT22 dengan alat pengukuran suhu dan kelembaban udara pembanding. Apabila nilai error yang dihasilkan dalam 10 kali percobaan adalah kurang dari 10% maka dapat disimpulkan bahwa sensor dapat berfungsi dengan baik.



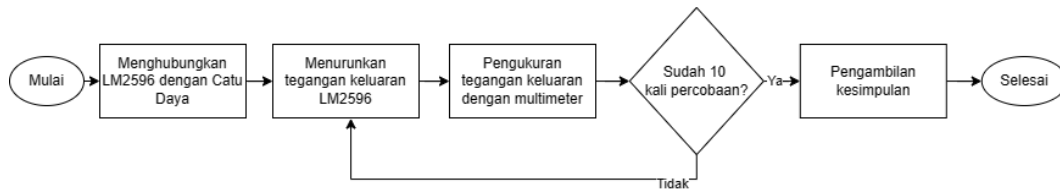
### 3. 1. 9. 7. Pengujian SX1262 868M Lora HAT



Gambar 3. 26 *Flowchart* pengujian modul LoRa SX1262 868M Lora HAT

Gambar 3.26 adalah *flowchart* dari pengujian modul LoRa. SX1262 868M Lora HAT merupakan modul LoRa yang digunakan pada penelitian ini. Pengujian SX1262 868M Lora HAT atau modul LoRa merupakan pengujian untuk mengetahui apakah modul LoRa yang digunakan dapat berfungsi dengan baik atau tidak. Pengujian dilakukan dengan melakukan pengiriman data dari modul LoRa yang berperan sebagai pengirim data dan modul LoRa yang berperan sebagai penerima data. Modul LoRa yang berperan sebagai pengirim terhubung dengan Arduino Mega dan modul LoRa yang berperan sebagai penerima terhubung dengan Raspberry Pi 4 Model B. Pengujian dilakukan dengan memisahkan kedua modul LoRa pada jarak tertentu dan apabila modul LoRa penerima mampu menerima data yang dikirimkan oleh modul LoRa pengirim maka modul LoRa dapat disimpulkan berfungsi dengan baik. Jarak antara modul LoRa pengirim dan modul LoRa penerima kemudian diperlebar hingga modul LoRa penerima tidak dapat menerima data yang dikirimkan oleh modul LoRa pengirim. Penambahan jarak antara modul LoRa pengirim dan modul LoRa penerima ditujukan untuk mengetahui jarak maksimal kedua modul LoRa dapat berkomunikasi.

### 3. 1. 9. 8. Pengujian Modul LM2596



Gambar 3. 27 Flowchart pengujian modul LM2596

Gambar 3.27 merupakan *flowchart* dari pengujian modul LM2596. Pengujian LM2596 adalah pengujian untuk mengetahui kondisi modul LM2594 dalam kondisi baik atau tidak dengan parameter bahwa modul LM2596 dapat menurunkan tegangan dari masukan berupa tegangan 9 V dari catu daya. Apabila tegangan keluaran terukur oleh multimeter dari modul LM2596 adalah kurang dari 9 V maka modul LM2596 dalam kondisi baik. Pengujian modul LM2596 dilakukan dalam 10 kali percobaan.

### 3. 1. 9. 9. Pengujian Maximum Power Point Tracking (MPPT)



Gambar 3. 28 Flowchart pengujian MPPT

Gambar 3.28 merupakan *flowchart* pengujian MPPT. Pengujian maximum power point tracking (MPPT) merupakan pengujian kondisi MPPT apakah dalam kondisi baik atau tidak. Pengujian dilakukan dengan melakukan pengecekan arus pada keluaran MPPT menuju baterai dengan bantuan multimeter. Pengujian dilakukan dalam 10 kali percobaan pengukuran. Apabila nilai arus muncul pada multimeter maka dapat disimpulkan bahwa unit MPPT dalam kondisi baik.

### 3. 1. 9. 10. Pengujian Panel Surya Polycrystalline



Gambar 3. 29 *Flowchart* pengujian panel surya

Gambar 3.29 merupakan *flowchart* dari pengujian panel surya polycrystalline yang akan digunakan. Pengujian panel surya polycrystalline merupakan pengujian untuk mengetahui kondisi unit panel surya yang digunakan apakah berfungsi dengan baik atau tidak. Pengujian dilakukan dengan menempatkan panel surya pada cahaya matahari untuk kemudian dilakukan pengecekan tegangan keluarannya menggunakan multimeter selama 10 kali percobaan. Apabila nilai tegangan muncul pada multimeter maka dapat disimpulkan bahwa panel surya dapat berfungsi dengan baik.

### 3. 1. 9. 11. Pengujian Baterai 12 V



Gambar 3. 30 *Flowchart* pengujian baterai

Gambar 3.30 merupakan *flowchart* pengujian baterai dimana baterai yang digunakan adalah baterai Sinus Electric SE20-12. Pengujian baterai 12 V merupakan pengujian untuk mengetahui kondisi unit baterai yang akan digunakan apakah dapat berfungsi dengan baik atau tidak. Pengujian dilakukan dengan melakukan percobaan menyalakan lampu secara langsung oleh baterai. Apabila lampu dapat menyala maka baterai dalam kondisi baik.

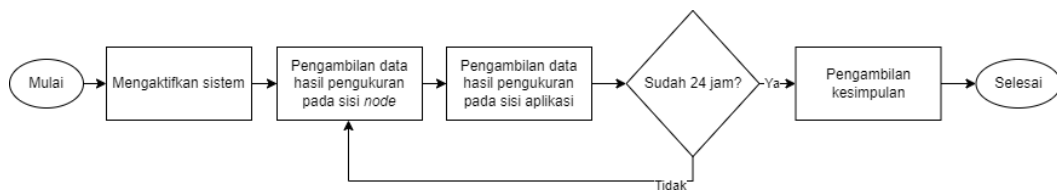
### 3. 1. 10. Penggantian Komponen

Apabila dalam pengujian komponen terdapat komponen yang tidak berfungsi dengan baik, maka dilakukan penggantian komponen. Penggantian komponen dilakukan dengan harapan komponen pengganti dapat berfungsi dengan baik sehingga mendukung kinerja sistem. Setelah didapatkan komponen pengganti, komponen pengganti tersebut kemudian dilakukan pengujian ulang untuk mengetahui kondisi komponen pengganti tersebut apakah dalam kondisi baik atau tidak.

### 3. 1. 11. Instalasi Sistem

Instalasi sistem merupakan proses penggabungan unit yang telah lolos uji. Penggabungan dilakukan melalui proses pengkabelan serta pemrograman sesuai dengan perencanaan.

### 3. 1. 12. Pengujian Sistem



Gambar 3. 31 *Flowchart* pengujian sistem

Gambar 3.31 merupakan *flowchart* dari pengujian sistem. Pengujian sistem adalah pengujian fungsionalitas sistem apakah sistem yang telah dibangun dapat berjalan dan menghasilkan keluaran yang diharapkan atau tidak. Keluaran yang diharapkan adalah tampilnya data hasil pembacaan sensor untuk PM1.0, PM 2.5, PM 10, suhu, kelembaban udara, serta kecepatan angin pada aplikasi android yang terhubung dengan sistem. Pengujian dilakukan dengan membandingkan hasil pembacaan pada *node* dengan hasil pembacaan yang diterima pada aplikasi android

dalam durasi 24 jam percobaan. Apabila data yang tampil adalah sama antara keduanya maka dapat disimpulkan bahwa sistem bekerja dengan baik.

### **3. 1. 13. Analisis**

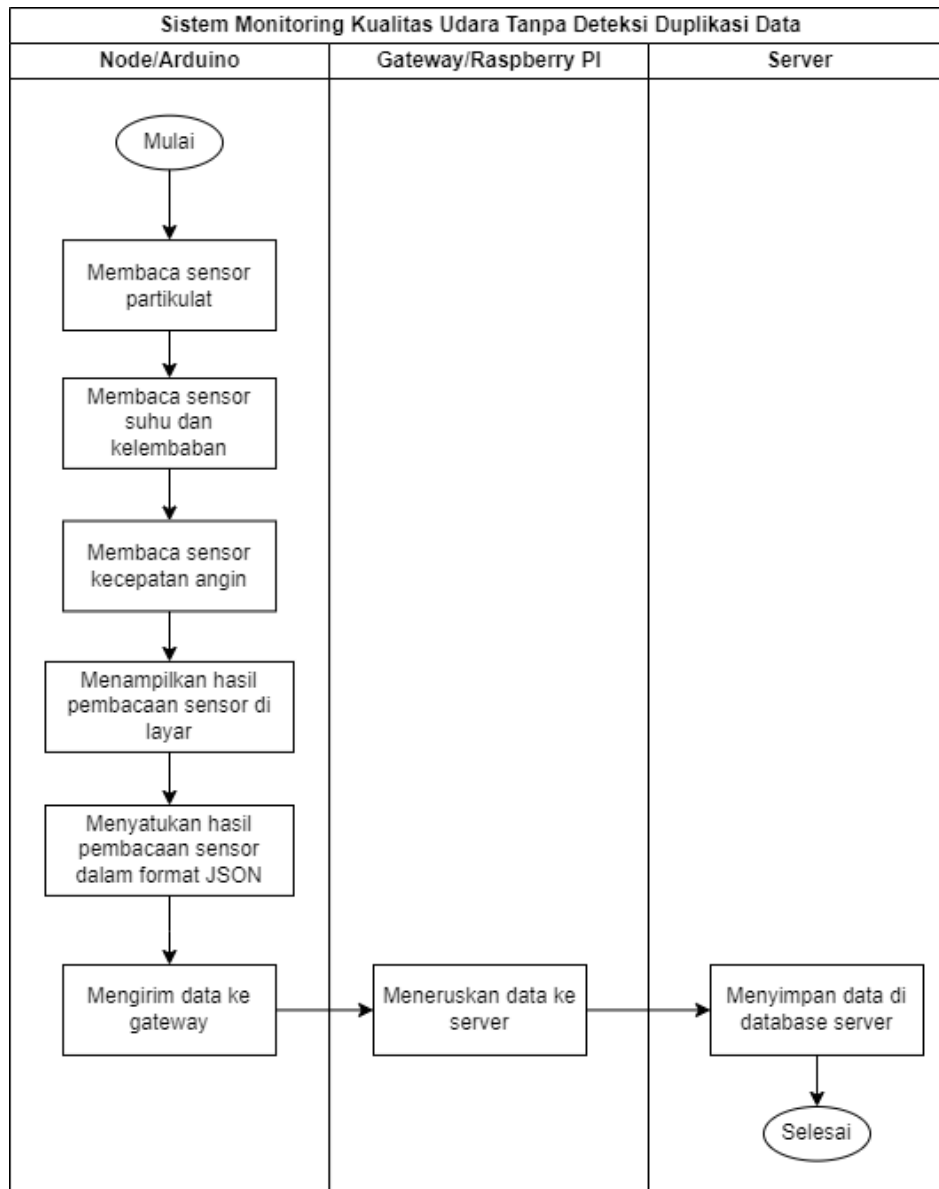
Analisis dilakukan dengan membandingkan antara kajian teori dengan hasil pengujian sistem yang telah dilakukan. Apabila terdapat perbedaan maka dilakukan analisis untuk mengetahui penyebab munculnya perbedaan tersebut. Apabila terjadi kesamaan maka hasil dari sistem yang dibangun sesuai dengan kajian teori.

### **3. 1. 14. Kesimpulan**

Kesimpulan merupakan proses penarikan kesimpulan dari hasil analisis yang telah dilakukan berdasarkan pengujian sistem sebelumnya. Penarikan kesimpulan dilakukan berdasarkan tujuan penelitian yang telah ditetapkan sebelumnya.

### 3. 2. Flowchart Sistem Monitoring Kualitas Udara Tanpa Deteksi Duplikasi

#### Data



Gambar 3. 32 *Flowchart* Sistem Monitoring Tanpa Deteksi Duplikasi Data

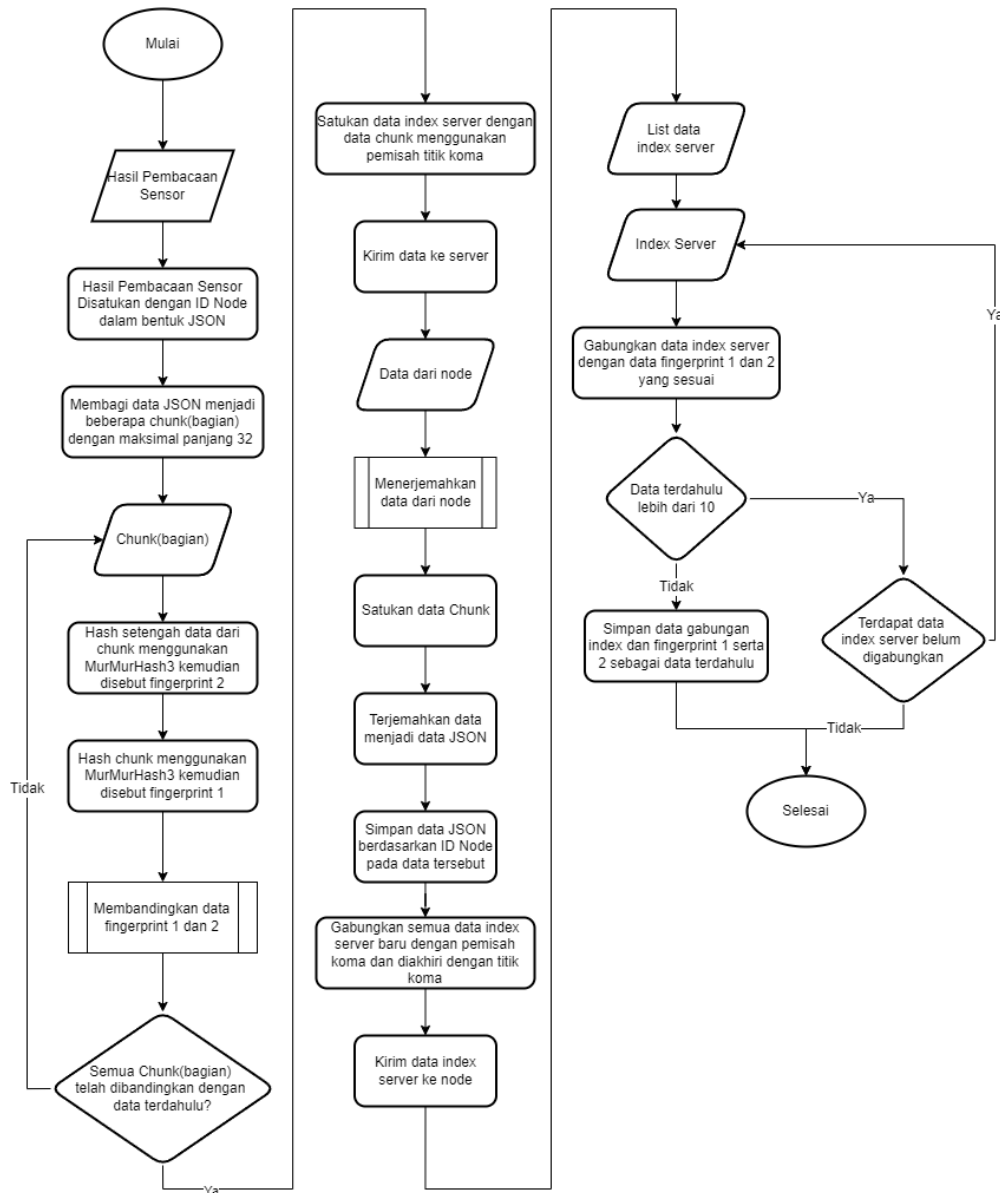
Gambar 3.32 adalah alur kerja dari sistem monitoring kualitas udara apabila metode deteksi duplikasi data tidak diterapkan. Pada gambar 3.4 dijelaskan bahwa setelah setiap sensor dibaca oleh node/arduino, hasil pembacaan setiap sensor tersebut kemudian disatukan dalam format json menjadi data contoh berikut ini:

```
{"id": "7d80e4e0","data": "ht120v|65,31,17,29,53,12.68,"}
```

Data “id” pada contoh diatas mewakili id pengenalan node/arduino pada database server dan data “data” pada contoh diatas mewakili data sensor hasil pembacaan. Karakter “h” pada data mewakili kelembaban udara, karakter “t” pada data mewakili suhu, karakter “1” pada data mewakili PM 1.0, karakter “2” pada data mewakili PM 2.5, karakter “0” pada data mewakili PM 10, dan karakter “v” pada data mewakili kecepatan angin. Data pembacaan sensor tidak akan mencantumkan hasil pembacaan untuk sensor yang tidak mengirimkan data pembacaan. Sehingga, bisa saja data yang dikirimkan tidak mencantumkan data kecepatan angin karena sensor kecepatan angin tidak mengirimkan data.

Data dengan format JSON dikirimkan oleh node/arduino menggunakan LoRa menuju gateway/raspberry pi. Pada gateway/raspberry pi, data dari arduino kemudian diteruskan melalui jaringan internet dengan protokol mqtt menuju server untuk kemudian disimpan dalam database server.

### 3.3. Flowchart Penerapan Deteksi Duplikasi Data pada Sistem



Gambar 3. 33 Flowchart Sistem Deteksi Duplikasi Data

Gambar 3.33 merupakan alur kerja dari sistem deteksi duplikasi data pada sistem monitoring. Sistem deteksi duplikasi data menerima data hasil pembacaan sensor untuk kemudian disatukan dengan *id node* menjadi sebuah data dengan format JSON. Data JSON yang dibuat memiliki *key* “id” yang berisi data *id node*, serta *key* “data” yang berisi data hasil pembacaan sensor.

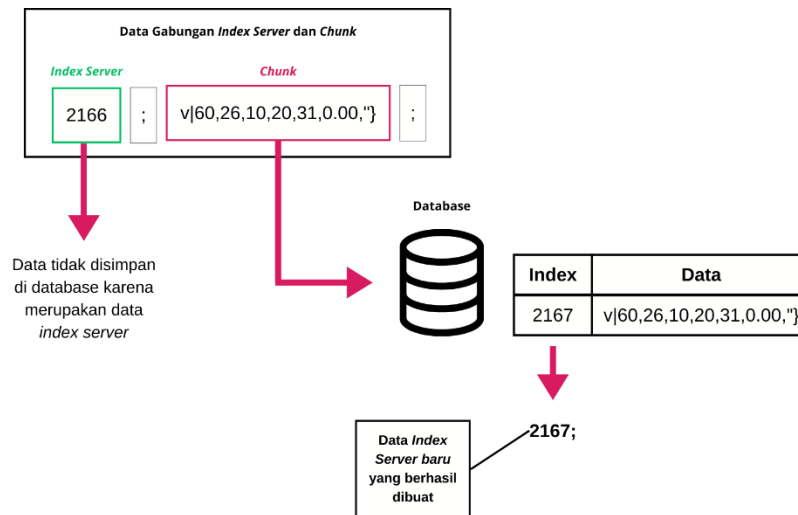


Data JSON kemudian dibagi menjadi beberapa *chunk* (bagian) dengan setiap *chunk*-nya memiliki panjang maksimal 32 karakter. Setiap *chunk* kemudian dilakukan proses *fingerprinting* dengan melakukan proses *hash* menggunakan algoritma *hash* murmurhash3. Proses *fingerprinting* terbagi menjadi dua dimana *fingerprint* pertama melakukan *hash* untuk setengah dari keseluruhan karakter pada setiap *chunk* dan *fingerprint* kedua melakukan *hash* untuk keseluruhan karakter pada *chunk*. Hasil *fingerprint* 1 dan *fingerprint* 2 kemudian dibandingkan dengan *fingerprint* 1 dan *fingerprint* 2 terdahulu lainnya yang tersimpan pada memori ram. Setiap *fingerprint* 1 dan *fingerprint* 2 yang tersimpan pada ram merupakan satu kesatuan sesuai dengan data *chunk* sebelum dilakukan proses *fingerprinting*. Pada kesatuan ini, juga tersimpan data *index server* yang merupakan tempat data *chunk* tersimpan pada server. Pada proses perbandingan, apabila *fingerprint* 1 dan *fingerprint* 2 terdeteksi sebagai duplikasi maka keluaran dari proses ini adalah data *index server* dan apabila terdeteksi sebagai data unik maka keluaran dari proses ini adalah data *chunk* sebelum dilakukan proses *fingerprinting*. Apabila semua *fingerprint* 1 dan *fingerprint* 2 yang mewakili setiap data *chunk* telah dibandingkan, proses dilanjutkan dengan menggabungkan setiap data *index server* dan data *chunk* keluaran dari proses perbandingan menjadi satu dengan karakter pemisah titik koma (“;”). Data *index server* dan data *chunk* yang telah disatukan kemudian dikirimkan menuju server untuk disimpan sebagai data terbaru serta sebagai upaya sinkronisasi data terdahulu antara server dengan *node*. Sinkronisasi ini diperlukan sehingga data yang dikirimkan oleh *node* menuju server dapat diterjemahkan menjadi data asli sebelum proses deteksi duplikasi data berjalan.

Setelah data dari *node* diterima oleh server, proses selanjutnya dilanjutkan dengan menerjemahkan data dari *node*. Penerjemahan dilakukan seperti pada Gambar 3.31. Pada proses penerjemahan, data *index server* beserta data *chunk* yang disatukan pada data dari *node* dipisahkan kemudian untuk data *index server* diambil data *chunk* berdasarkan *index server*-nya. Data-data *chunk* hasil penerjemahan kemudian disatukan membentuk data JSON yang berisi *id node* dengan *key* “id” dan berisi data pembacaan sensor dengan *key* “data”. Berdasarkan data *id node*, data kemudian disimpan sebagai data pembacaan terbaru dari *node* dengan *id node* yang sesuai. Selain menghasilkan data-data asli, pada proses penerjemahan juga dihasilkan data *index server* baru dimana tempat data *chunk* unik dari *node* disimpan. Data-data *index server* baru kemudian disatukan dengan karakter pemisah “;” dan diakhiri dengan karakter “;” untuk selanjutnya dikirimkan menuju *node*. Data *index server* baru dari server kemudian diterima oleh *node* dan kemudian dipasangkan satu-persatu bersamaan dengan data *fingerprint 1* dan *fingerprint 2* yang juga mewakili data *chunk* yang sama. Data *index server*, *fingerprint 1*, dan *fingerprint 2* yang telah digabungkan ini disimpan dalam memori ram *node* sebagai data terdahulu yang akan dibandingkan pada proses perbandingan. Apabila semua data *index server* telah disimpan maka proses deteksi duplikasi data berakhir.

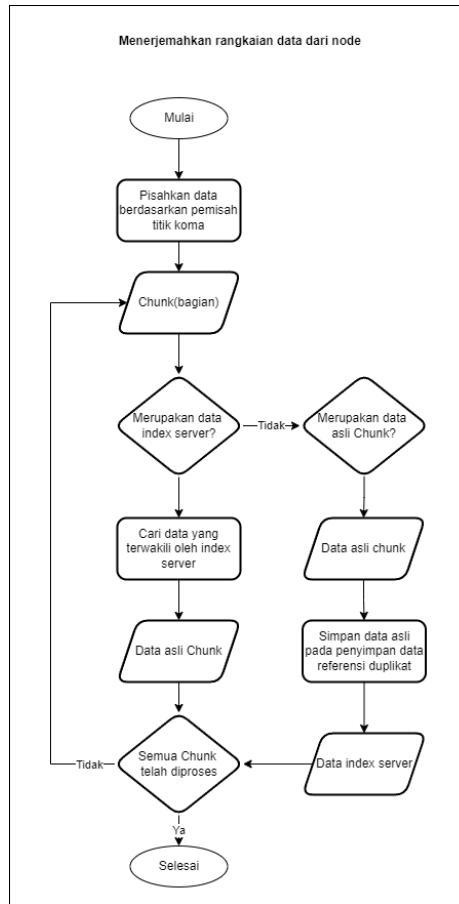
Data terdahulu pada *node* dibatasi hanya untuk 10 data terdahulu. Oleh karena itu, apabila data *chunk* baru pada *node* telah mencapai lebih dari 10 maka data *chunk* terbaru tersebut tidak akan disimpan data *fingerprint 1*, *fingerprint 2*, serta *index server*-nya. Sehingga, apabila data *chunk* tersebut ternyata memiliki kesamaan dengan data *chunk* lain setelahnya, data *chunk* tersebut akan dianggap

sebagai data unik dan server akan memberikan data *index* server ke *node*. Data *index* server tersebut akan diterima oleh *node* namun tidak akan disimpan sebagai data terdahulu karena telah melebihi 10 data terdahulu.



Gambar 3. 34 Ilustrasi Pembuatan Index Server

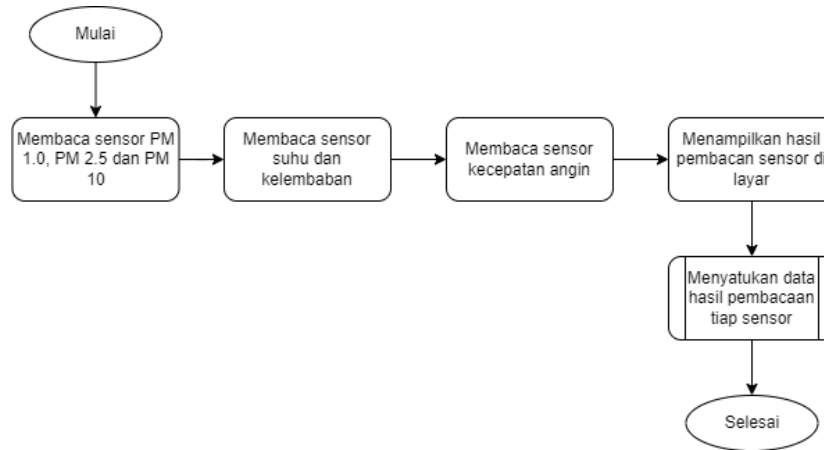
Data *index* server merupakan lokasi tempat server menyimpan data *chunk*. Pada Gambar 3.34 diilustrasikan mengenai alur kerja bagaimana *index server* dibuat dari data gabungan yang berasal dari *node*. Data gabungan dari *node* diterjemahkan apakah data-data tersebut adalah *chunk* atau data *index server* yang telah dibuat sebelumnya. Data dapat dikelompokkan menjadi data *index server* apabila data memiliki tipe data “integer” serta dapat dikelompokkan menjadi data *chunk* ketika data merupakan tipe data “string”. Data *index server* akan bertambah seiring penambahan data *chunk* baru ke server. Menggunakan data *index server* ini, server dapat mengetahui data asli yang dimaksudkan oleh *node* untuk ditampilkan sebagai data pembacaan terbaru.



Gambar 3. 35 Alur kerja menerjemahkan data dari *node* di server

Gambar 3.35 adalah alur kerja untuk penerjemahan data dari *node* yang diterima oleh server. Data yang dikirimkan oleh *node* merupakan data gabungan yang setiap data didalamnya mewakili satu data *chunk*. Bagian data dari *node* dapat berupa data *index* server dan juga data *chunk*. Apabila bagian data merupakan data *index* server maka bagian tersebut merupakan data duplikat sehingga oleh server dilakukan pencarian data *chunk*-nya sesuai dengan data *index* server yang dikirimkan. Apabila bagian data merupakan data *chunk* maka bagian tersebut merupakan data unik sehingga oleh server dilakukan penyimpanan data *chunk*. Letak dimana data *chunk* baru tersimpan kemudian diambil sebagai data *index*

server yang harus dikirimkan ke *node* sebagai upaya sinkronasi data *chunk* sehingga data dari *node* dapat diterjemahkan menjadi data asli dengan baik.



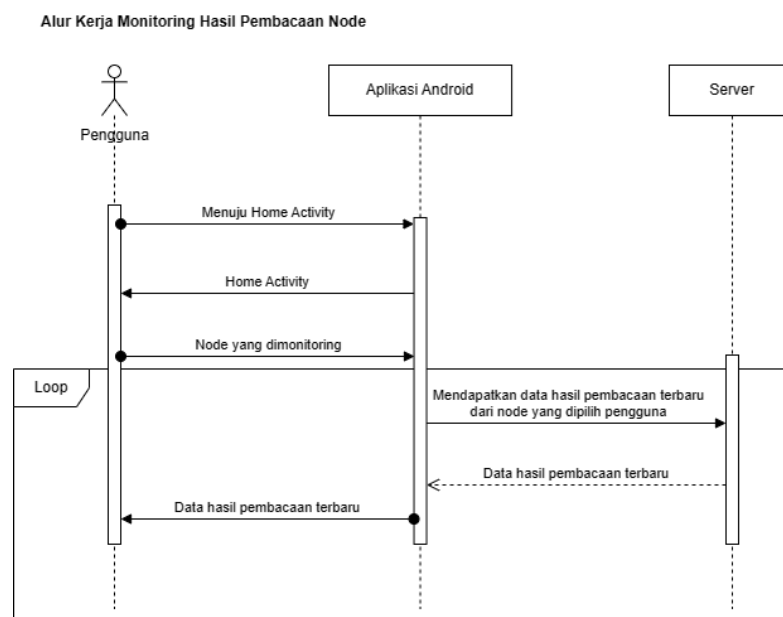
Gambar 3. 36 *Flowchart* Membaca Sensor pada Node/Arduino

Gambar 3.36 merupakan *flowchart* detail untuk pembacaan sensor. Proses membaca sensor diawali dengan pembacaan sensor partikulat (SEN0177) untuk membaca data PM 1.0, PM 2.5 dan PM 10. Kemudian proses dilanjutkan dengan pembacaan sensor suhu dan kelembaban yaitu sensor DHT22. Setelah pembacaan sensor DHT22 dilakukan pembacaan sensor kecepatan angin. Setelah pembacaan sensor kecepatan angin selesai, proses dilanjutkan dengan menampilkan hasil-hasil pembacaan sensor pada layar TFT untuk kemudian hasil-hasil dari pembacaan sensor disatukan menjadi satu data. Penyatuan data hasil-hasil pembacaan sensor dilakukan untuk menyederhanakan proses pengiriman data hasil-hasil pembacaan sensor sehingga proses pengiriman data hasil-hasil tersebut dapat dilakukan dalam satu kali pengiriman.

Data sensor disatukan menjadi bentuk seperti berikut:

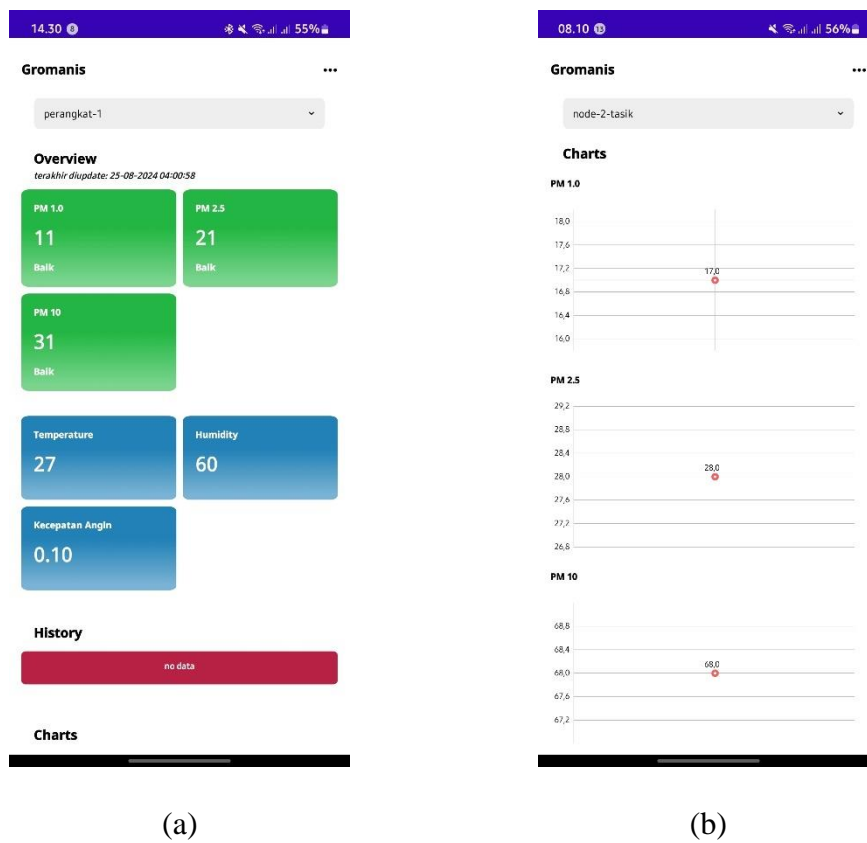
ht120v|65,31,17,29,53,12.68,

Data hasil pembacaan sensor yang telah disatukan terbagi menjadi dua bagian yang dipisahkan oleh karakter “|”. Bagian pertama merupakan bagian yang berisi label mengenai data hasil pembacaan sensor apa saja yang telah disatukan. Data label berbentuk karakter “h”, “t”, “1”, “2”, “0”, dan “v”. Label “h” mewakili data kelembaban udara, label “t” mewakili data suhu, label “1” mewakili data PM 1.0, label “2” mewakili data PM 2.5, label “0” mewakili data PM 10, dan label “v” mewakili data kecepatan angin. Bagian kedua dari gabungan data merupakan nilai hasil pembacaan. Nilai-nilai pembacaan pada bagian kedua saling dipisahkan oleh karakter pemisah koma (“,”). Urutan nilai pembacaan sesuai dengan urutan label pada bagian pertama. Untuk sensor yang tidak mengirimkan data maka data sensor tersebut tidak akan digabungkan dengan nilai hasil pembacaan sensor lain, sehingga semisal sensor kecepatan angin tidak memberikan nilai pembacaan maka tidak akan ada label “v” pada bagian pertama dan jumlah nilai hasil pembacaan pada bagian kedua juga berkurang satu.



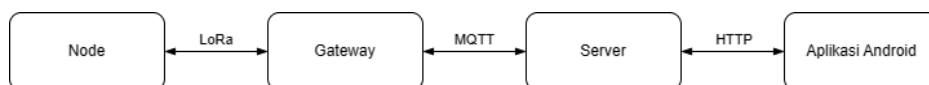
Gambar 3. 37. Alur Kerja Monitoring Hasil Pembacaan Node Pada Aplikasi Android

Gambar 3.37 menjelaskan mengenai alur kerja sistem untuk aplikasi android. Pada aplikasi android, pembacaan sensor dilakukan dengan melakukan request ke server untuk mendapatkan data hasil pembacaan terbaru. Data hasil pembacaan terbaru kemudian diekstraksi sehingga didapatkan data suhu, kelembaban, PM 1.0, PM 2.5, PM 10, dan kecepatan angin. Data ditampilkan pada *activity* Home sebagai teks dan grafik seperti pada gambar 3.38 dibawah ini.



Gambar 3. 38. (a) Tampilan *activity* Home bagian teks (b) Tampilan *activity* Home bagian grafik

### 3. 4. Arsitektur Sistem



Gambar 3. 39 Arsitektur sistem

Gambar 3.39 merupakan gambaran arsitektur sistem yang digunakan yaitu terdiri dari satu unit node, satu unit gateway, satu unit server, serta aplikasi android. *Node* pada sistem terhubung dalam topologi star dengan *gateway* dan terhubung menggunakan jaringan LoRa. *Gateway* terhubung dengan server dan berkomunikasi menggunakan protokol MQTT untuk menyimpan data hasil pembacaan sensor yang dikirimkan oleh *node* menuju *gateway* dan kemudian diteruskan menuju server. Data hasil pembacaan sensor yang tersimpan pada server dibaca secara berulang oleh aplikasi android dengan *interval* 3 detik sehingga data yang ditampilkan oleh aplikasi android adalah data hasil pembacaan sensor terbaru.

Node pada sistem bertugas sebagai pengambil data serta pengirim data menuju gateway. Pengambilan data dilakukan node dari hasil pembacaan sensor-sensor seperti SEN1077, SEN0170, serta DHT22. Pengiriman data node menuju gateway dilakukan melalui jaringan LoRa menggunakan modul SX1278. Sebelum data dikirimkan, dilakukan proses deteksi data duplikat untuk mencegah adanya duplikasi pada data yang dikirim.

Gateway pada sistem bertugas sebagai perantara antara node dengan server. Proses pada gateway berputar pada penerimaan data dari unit node dan pengiriman data dari node menuju server. Server bertugas sebagai penyimpan data hasil pembacaan sensor pada node serta sebagai penghubung antara data hasil pembacaan sensor dengan aplikasi android. Aplikasi android berperan sebagai penampil data sensor di sisi pengguna sistem. Data sensor yang ditampilkan pada aplikasi android didapatkan dari server.



Server bertugas sebagai tempat penyimpanan data hasil pembacaan sensor dari node. Kemudian, server bersama dengan aplikasi android berperan sebagai pengatur atau manajemen gateway dan device. Pada aplikasi android, pengguna aplikasi dapat mendaftarkan gateway dan device baru, menghapus gateway dan device yang ada, dan memonitoring hasil pembacaan sensor.

### 3.5. Pengambilan Data

Penjelasan lebih lanjut mengenai pengambilan data dijelaskan pada tabel 3.9.

Tabel 3.9 Cara Pengambilan Data

No	Variabel	Definisi Operasional	Cara Pengambilan Data
1.	Fungsionalitas komponen	Mengidentifikasi fungsionalitas setiap komponen agar sistem dapat berjalan sesuai perencanaan.	Melakukan observasi
2.	Kinerja sistem tanpa penerapan metode deteksi duplikasi data	Mengidentifikasi akurasi data, latensi saat pengiriman data node ke gateway, serta daya listrik yang diperlukan oleh modul LoRa saat pengiriman data pada sistem tanpa penerapan deteksi duplikasi data.	Melakukan observasi

No	Variabel	Definisi Operasional	Cara Pengambilan Data
3.	Kinerja sistem dengan penerapan metode deteksi duplikasi data	Mengidentifikasi akurasi data, latensi saat pengiriman data node ke gateway, serta daya listrik yang diperlukan oleh modul LoRa saat pengiriman data pada sistem dengan penerapan deteksi duplikasi data.	Melakukan observasi
4.	Fungsionalitas sistem	Mempelajari fungsionalitas sistem sesuai perencanaan.	Melakukan observasi

### 3.6. Waktu Lokasi Penelitian

Penelitian akan dilakukan selama 6 bulan sesuai matriks kerja penelitian pada Tabel 3.10 dengan lokasi penelitian ini akan dilaksanakan di kawasan Kampus Universitas Siliwangi.

Tabel 3. 10 Matriks Kerja Penelitian

No	Kegiatan	Bulan ke-1				Bulan ke-2				Bulan ke-3				Bulan ke-4				Bulan ke-5				Bulan ke-6			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Menentukan Tema Penelitian																								

2	Studi Literatur	■	■	■	■																				
3	Menentukan Rumusan Masalah			■	■																				
4	Perancangan Sistem					■	■	■																	
5	Pengumpulan Kebutuhan Sistem						■	■	■	■															
6	Pengujian <i>Software</i>									■	■	■													
7	Pengujian Komponen												■	■	■										
8	Penggantian Komponen												■	■	■										
9	Instalasi Sitem															■	■	■							
10	Pengujian Sistem																			■	■	■			
11	Analisis																								■