

BAB 3

METODE PENELITIAN

3.1 ALAT DAN BAHAN

3.1.1 PERANGKAT KERAS

Tabel 3.1 Perangkat Keras

| No | Perangkat Keras | Jumlah |
|----|--|--------|
| 1 | <i>NodeMCU V3 Lolin</i> | 3 |
| 2 | <i>Laptop Asus X451C</i> <i>Windows 8.1</i> | 1 |
| 3 | <i>Smartphone Sony Xperia XZ</i> | 1 |
| 4 | Sensor Ultrasonik HC-SR04 | 3 |
| 5 | LED Merah | 3 |

Penjelasan perangkat keras :

1. *NodeMCU V3 Lolin*

NodeMCU V3 Lolin mempunyai spesifikasi tegangan *input 5 Volt*, jumlah *pin I/O Digital 11*, *pin I/O Analog 1*, dan *USB to Serial Converter CH340G*. *Pin* yang digunakan yaitu *2 pin I/O Digital* untuk sensor ultrasonik dan *1 pin I/O Digital* untuk LED. Kemudian *USB to Serial Converter CH340G* digunakan untuk memasukan program ke *NodeMCU*. *NodeMCU* ini digunakan sebagai mikrokontroler tempat sampah pintar untuk memproses data masukan dari sensor ultrasonik yang *output*-nya berupa LED dan aplikasi (*Line* dan *Facebook Messenger*) pada *smartphone*.

2. *Laptop Asus X451C*

Laptop asus *X451C* mempunyai spesifikasi RAM 4 GB, kapasitas *harddisk 500 GB*, sistem operasi *Windows 8.1 Pro*, dan *processor Intel i3*. *Laptop* ini digunakan untuk menjalankan *software Arduino IDE* yang digunakan untuk memprogram *NodeMCU*.

3. *Smartphone Sony Xperia XZ*

Smartphone Sony Xperia XZ mempunyai spesifikasi RAM 3 GB, kapasitas *Memory Internal* 32 GB, sistem operasi *Android 8.0 Oreo* , dan *processor Snapdragon 820*. *Smartphone* ini digunakan untuk dapat menjalankan aplikasi *Line* dan *Facebook Messenger* yang akan menampilkan pemberitahuan ketinggian sampah.

4. Sensor Ultrasonik HC-SR04

Sensor ultrasonik HC-SR04 mempunyai spesifikasi tegangan *input* 5 Volt, konsumsi arus 5 mA, dan frekuensi operasi 40 KHz. Sensor ultrasonik ini digunakan untuk mendeteksi ketinggian sampah.

5. LED Merah

LED merah digunakan sebagai parameter sensor ultrasonik. Jika ketinggian sampah yang dideteksi sensor ultrasonik ≥ 16 cm maka LED menyala, kemudian jika ketinggian sampah yang dideteksi sensor ultrasonik < 16 cm maka LED mati.

3.1.2 PERANGKAT LUNAK

Tabel 3.2 Perangkat Lunak

| No | Perangkat Lunak | Jumlah |
|----|------------------------------------|--------|
| 1 | <i>Software Arduino IDE</i> | 1 |
| 2 | Aplikasi IFTTT | 1 |
| 4 | Aplikasi <i>Webhooks</i> | 1 |
| 5 | Aplikasi <i>Line</i> | 1 |
| 6 | Aplikasi <i>Facebook Messenger</i> | 1 |

Penjelasan perangkat lunak :

1. *Software Arduino IDE* versi 1.8.1

Software Arduino IDE ini digunakan untuk memprogram *NodeMCU* agar dapat terkoneksi dengan sensor dan modul yang dibutuhkan, serta untuk mengunggah kode program ke *NodeMCU*.

2. IFTTT versi 3.7.6 (IFTTT.com)

Digunakan untuk menghubungkan dua aplikasi atau *web* agar saling terhubung. Untuk pemicu (*trigger*) menggunakan *Webhooks* untuk

mengirimkan pesan pemberitahuan ketinggian sampah ke reaksi (*action*) menggunakan *Line* dan *Facebook Messenger*.

3. Aplikasi *Webhooks*

Digunakan sebagai pemicu dari IFTTT yang akan mengirimkan pesan pemberitahuan ketinggian sampah ke aplikasi *Line* dan *Facebook Messenger*.

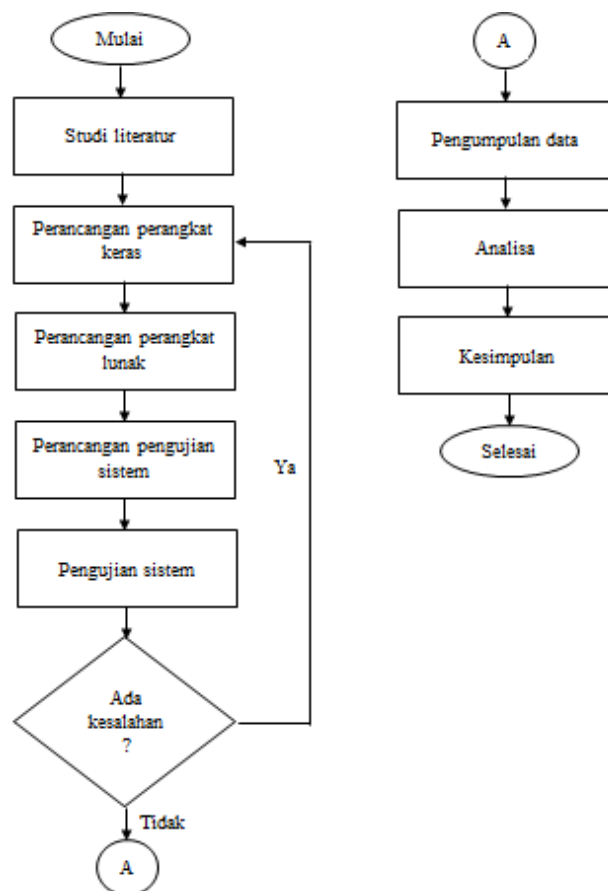
4. Aplikasi *Line* versi 8.19.2

Aplikasi ini digunakan sebagai reaksi dari IFTTT dan untuk menampilkan pemberitahuan berupa pesan ketinggian sampah melalui *smartphone*.

5. Aplikasi *Facebook Messenger* versi 207.0.0.13.99

Aplikasi ini digunakan sebagai reaksi dari IFTTT dan untuk menampilkan pemberitahuan berupa pesan ketinggian sampah melalui *smartphone*.

3.2 ALUR PENELITIAN

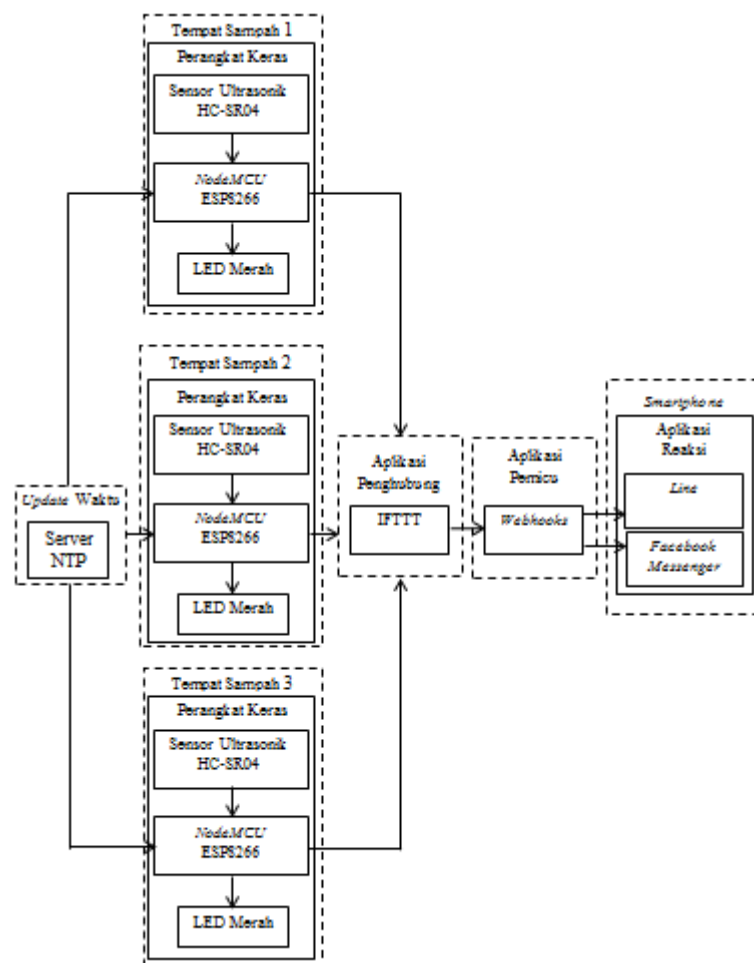


Gambar 3.1 *Flowchart* Alur Penelitian

3.2.1 STUDI LITERATUR

Merupakan metode pengumpulan data dengan mencari referensi berupa buku, artikel, jurnal, dan pembahasan literatur yang terkait dengan bidang penelitian yang dilakukan. Dengan metode ini maka akan ditemukan gambaran mengenai suatu topik dan bisa diketahui perbedaan penelitian antara yang satu dengan penelitian lainnya.

3.2.2 BLOK DIAGRAM SISTEM



Gambar 3.2 Blok Diagram Perancangan Tempat Sampah Pintar

Pada Gambar 3.2 merupakan blok diagram sistem tugas akhir. Dalam sistem ini terbagi menjadi lima bagian utama, yaitu *update* waktu, perangkat keras, aplikasi penghubung, aplikasi pemicu, dan aplikasi reaksi. Pada bagian *update* waktu terdapat server NTP dan bagian perangkat keras terdiri dari sensor

ultrasonik, *NodeMCU*, dan LED. Kemudian bagian aplikasi penghubung terdapat IFTTT, bagian aplikasi pemicu terdiri dari aplikasi *Webhooks*, dan bagian aplikasi reaksi terdiri dari aplikasi *Line* dan *Facebook Messenger*.

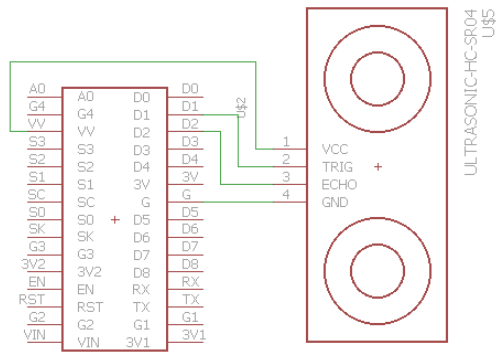
Sensor ultrasonik berfungsi untuk mendeteksi ketinggian sampah yang akan diproses pada *NodeMCU*, kemudian *NodeMCU* akan meng-*update* waktu secara *real time* tiap satu detik sekali melalui server NTP. Jika sesuai dengan jadwal yang sudah ditentukan pada server NTP yaitu jam 8 pagi maka data akan dikirimkan ke *Applets* pada IFTTT sesuai identitas tempat sampah pada *coding* program, dengan menggunakan alamat *key* dan *event* yang sudah dibuat. Maka *event* akan terjadi pada *Applets* IFTTT, kemudian *Webhooks* menerima permintaan *web* dari IFTTT.

IFTTT akan mengirimkan pesan ke aplikasi *Line* dan *Facebook Messenger* berupa pemberitahuan ketinggian sampah sesuai dengan identitas tempat sampah. Kemudian LED merah berfungsi sebagai parameter sensor ultrasonik. Jika tempat sampah penuh yaitu dengan ketinggian sampah ≥ 16 cm maka LED merah akan menyala, selanjutnya jika tempat sampah tidak penuh yaitu dengan ketinggian sampah < 16 cm maka LED merah akan mati.

3.2.3 PERANCANGAN PERANGKAT KERAS

1. Skema Rangkaian *NodeMCU* dengan Sensor Ultrasonik (HC-SR04)

Sensor ultrasonik merupakan sensor yang dapat membangkitkan dan mendeteksi gelombang ultrasonik. Sensor ultrasonik yang mempunyai dua elemen, yaitu elemen pendeteksi gelombang ultrasonik dan elemen pembangkit gelombang ultrasonik. Pada Gambar rangkaian 3.3 *pin* D1 pada *NodeMCU* terhubung ke sensor ultrasonik pada *pin* trig (*trigger*), *pin* D2 terhubung ke *pin* echo (*recive*), *pin* VV (+5v) terhubung ke VCC, dan *pin* GND terhubung ke *pin* gnd.



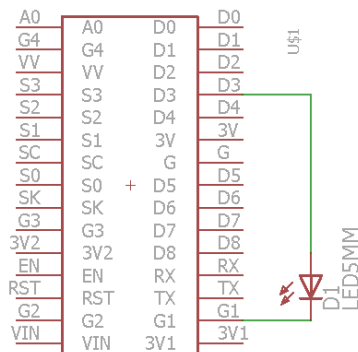
Gambar 3.3 Rangkaian *NodeMCU* dengan Sensor Ultrasonik (HC-SR04)

Tabel 3.3 Konfigurasi Rangkaian *NodeMCU* dengan Sensor Ultrasonik (HC-SR04)

| No | NodeMCU | Sensor Ultrasonik |
|----|---------|-------------------|
| 1 | D1 | Trig |
| 2 | D2 | Echo |
| 3 | VV | VCC |
| 4 | GND | Gnd |

2. Skema Rangkaian *NodeMCU* dengan LED Merah

LED merupakan komponen elektronika yang bisa memancarkan cahaya non monokromatik ketika diberikan tegangan maju. Warna cahaya pada LED tergantung dari jenis bahan semikonduktor yang digunakan. LED terdiri dari sebuah *chip* semikonduktor yang didoping sehingga menciptakan *junction* P dan N. Pada Gambar rangkaian 3.4 *pin* D3 pada *NodeMCU* terhubung ke *pin* positif (+) pada LED merah dan *pin* GND terhubung ke *pin* negatif (-).



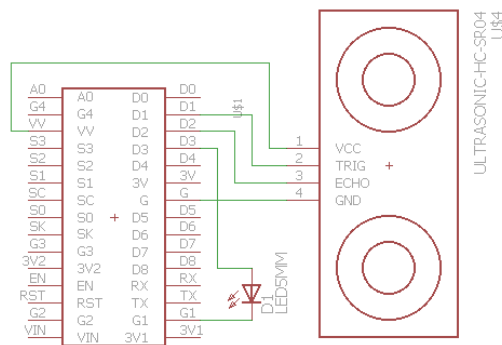
Gambar 3.4 Rangkaian *NodeMCU* dengan LED Merah

Tabel 3.4 Konfigurasi Rangkaian *NodeMCU* dengan LED Merah

| No | NodeMCU | LED |
|----|---------|-----|
| 1 | D3 | + |
| 2 | GND | - |

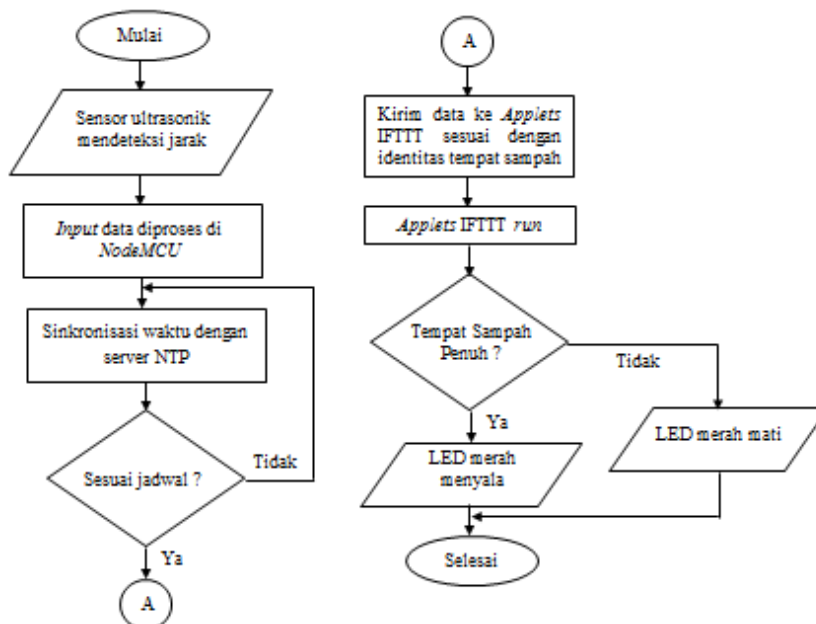
3. Skema Rangkaian Perangkat Keras Secara Keseluruhan

Pada Gambar rangkaian 3.5 menunjukkan rangkaian perangkat keras secara keseluruhan dari analisis yang sudah dilakukan. Pada skema rangkaian ini terdapat sensor ultrasonik dan LED merah yang sudah diintegrasikan dengan *NodeMCU*.



Gambar 3.5 Rangkaian Perangkat Keras Secara Keseluruhan

3.2.4 PERANCANGAN PERANGKAT LUNAK



Gambar 3.6 Flowchart Perancangan Tempat Sampah Pintar

Pada Gambar 3.6 dapat dijelaskan bahwa proses pertama yaitu sensor ultrasonik mendeteksi jarak dengan cara membangkitkan gelombang ultrasonik melalui penyulut, saat gelombang ultrasonik mengenai objek akan dipantulkan dengan sudut 15 derajat. Pada *Receiver* akan ditangkap gelombang ultrasonik tersebut. Kemudian gelombang pantulan tersebut akan diproses untuk menghitung jarak benda tersebut dengan rumus :

$$S = \frac{t}{29,1} \quad (3.1)$$

Proses selanjutnya *input* data diproses di *NodeMCU* yaitu dari data masukan sensor ultrasonik pada *pin I/O digital 1* dan *pin I/O digital 2*.

Setelah itu terjadi proses sinkronisasi waktu dari server NTP, yaitu *NodeMCU* akan meng-*update* waktu secara *real time* tiap satu detik sekali melalui server NTP. Kemudian jika sesuai dengan jadwal yang sudah ditentukan pada server NTP yaitu jam 8 pagi, maka selanjutnya akan terjadi proses kirim data ke *Applets* IFTTT sesuai dengan identitas tempat sampah. Pada proses ini data dikirimkan ke *applets* pada IFTTT sesuai identitas tempat sampah pada *coding* program, dengan menggunakan alamat *key* dan *event* yang sudah dibuat. Kemudian pada proses *Applets* IFTTT *run*, *event* akan terjadi pada *Applets* IFTTT.

Kemudian *Webhooks* menerima permintaan *web* dari IFTTT. Maka IFTTT akan mengirimkan pesan ke aplikasi *Line* dan *Facebook Messenger* berupa pemberitahuan ketinggian sampah sesuai dengan identitas tempat sampah. Selanjutnya jika tempat sampah penuh yaitu dengan ketinggian sampah ≥ 16 cm, maka akan terjadi proses LED merah menyala karena merupakan *output* dari sensor ultrasonik yang terhubung ke *pin I/O digital 3* pada *NodeMCU*. Jika tempat sampah tidak penuh yaitu dengan ketinggian sampah < 16 cm maka akan terjadi proses LED merah mati.

3.2.5 PERANCANGAN PENGUJIAN SISTEM

1. Perancangan Pengujian Akurasi Sensor Ultrasonik

Pada pengujian ini menggunakan jenis sampah kering. Untuk cara menguji akurasi sensor ultrasonik ini yaitu menggunakan pembanding berupa penggaris, apakah pengukuran jarak dengan sensor ultrasonik sesuai dengan jarak yang ada pada penggaris 30 cm. Sensor ultrasonik dan penggaris 30 cm disini digunakan untuk mengukur tinggi ruang kosong tempat sampah. Untuk dapat mengetahui tinggi sampah yang dideteksi sensor ultrasonik dan penggaris 30 cm dilakukan dengan cara :

$$\text{Tinggi tempat sampah} - \text{Tinggi ruang kosong tempat sampah} \quad (3.2)$$

Data yang diambil pada pengujian ini yaitu jarak deteksi sensor ultrasonik dan jarak sebenarnya pada penggaris 30 cm, dengan melakukan percobaan sebanyak 10 kali.

Untuk satu kali pengujian dilakukan dengan mengukur pada jarak ketinggian 5 cm, kemudian 10 cm, terus 15 cm. Setelah sampai pada ketinggian 15 cm, kemudian ulangi dari 5 cm lagi sampai dengan 10 kali pengujian. Dengan ini maka akurasi pengukuran sensor ultrasonik dengan jarak sebenarnya pada penggaris 30 cm dapat diketahui. Pada pengujian ini setelah data pengujian akurasi sensor ultrasonik didapatkan, selanjutnya menghitung nilai rata-rata selisih hasil pengukuran sensor ultrasonik dengan penggaris 30 cm dengan cara yaitu :

$$\text{Rata - rata} = \frac{\text{Jumlah selisih}}{\text{Jumlah Pengujian}} \quad (3.3)$$

Kemudian hitung varian dengan cara yaitu :

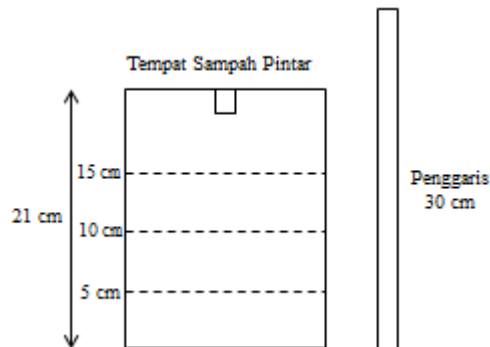
$$s^2 = \frac{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2}{n(n-1)} \quad (3.4)$$

Setelah nilai varian diketahui selanjutnya hitung nilai standar deviasi dengan cara yaitu :

$$s = \sqrt{\frac{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2}{n(n-1)}} \quad (3.5)$$

Pada pengujian ini juga dihitung nilai rata-rata *error* pengukuran dengan sensor ultrasonik dengan cara yaitu :

$$\text{Rata - rata} = \frac{\text{Jumlah Error}}{\text{Jumlah Pengujian}} \quad (3.6)$$



Gambar 3.7 Perancangan Pengujian Akurasi Sensor Ultrasonik

2. Perancangan Pengujian Identitas Tempat Sampah

Pada pengujian identitas tempat sampah dilakukan pengujian dengan menggunakan tiga tempat sampah pintar, dengan memasukan sampah pada ketiga tempat sampah tersebut. Apakah identitas tempat sampah yang penuh sesuai dengan identitas notifikasi di *smartphone*. Data yang diambil pada penelitian ini yaitu kesesuaian identitas tempat sampah yang penuh dengan notifikasi di *smartphone*, dengan melakukan pengujian sebanyak 5 kali. Pengujian ini dilakukan dengan memberi sampah berupa potongan kertas pada tempat sampah pintar. Untuk satu kali pengujian dilakukan sebanyak 7 kali, setelah sampai 7 kali kemudian ulangi pengujian dari awal kembali sampai 5 kali pengujian. Untuk lebih jelasnya dalam satu kali pengujian dapat dilihat pada Tabel 3.5 dibawah ini.

Tabel 3.5 Satu Kali Pengujian Identitas Tempat Sampah

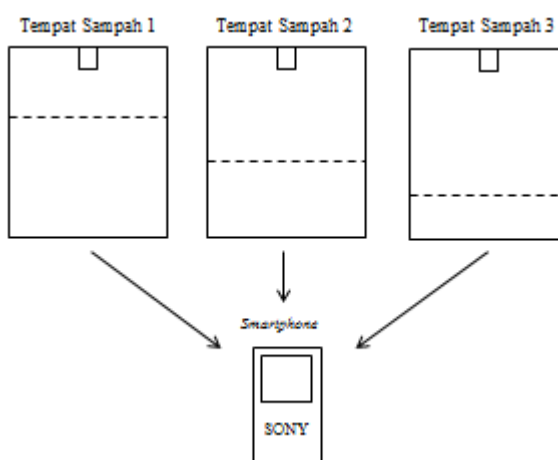
| No | Tempat Sampah 1 | Tempat Sampah 2 | Tempat Sampah 3 |
|----|-----------------|-----------------|-----------------|
| 1 | 1 | 0 | 0 |
| 2 | 0 | 1 | 0 |

| | | | |
|---|---|---|---|
| 3 | 0 | 0 | 1 |
| 4 | 1 | 1 | 0 |
| 5 | 1 | 0 | 1 |
| 6 | 0 | 1 | 1 |
| 7 | 1 | 1 | 1 |

Keterangan :

1 = Penuh

0 = Kosong



Gambar 3.8 Perancangan Pengujian Identitas Tempat Sampah

3. Perancangan Pengujian Waktu Pengiriman Notifikasi

Pada pengujian waktu pengiriman notifikasi dilakukan dengan menggunakan *timer* pada *smartphone* untuk mengetahui lamanya waktu pengiriman notifikasi dari tempat sampah pintar ke *smartphone*. Data yang diambil pada pengujian ini yaitu lama waktu pengiriman notifikasi dari tempat sampah pintar ke *smartphone*, dengan melakukan percobaan sebanyak 30 kali. Pengujian ini dilakukan dengan memberi sampah berupa potongan kertas pada tempat sampah pintar. Kemudian sensor ultrasonik akan mendeteksi, saat sensor ultrasonik mendeteksi ketinggian sampah 16 cm pada *timer* di pencet mulai. Saat notifikasi muncul pada *smartphone* maka pada *timer* dipencet berhenti. Dengan ini maka lama waktu pengiriman notifikasi tempat sampah pintar hingga sampai ke *smartphone* dapat diketahui.

Pada pengujian ini dihitung besarnya data yang dikirimkan tempat sampah pintar hingga sampai ke *smartphone*. Yaitu dengan menghitung jumlah karakter pada pesan yang dikirimkan tempat sampah pintar ke *smartphone*. Untuk menghitungnya menggunakan kode ASCII, untuk 1 kode atau karakter ASCII berukuran 1 Byte (8 bit) [22]. Setelah data pengujian waktu pengiriman notifikasi didapatkan dan besar data diketahui, selanjutnya menghitung nilai rata-rata waktu pengiriman notifikasi dengan cara yaitu :

$$\text{Rata - rata} = \frac{\text{Jumlah Waktu Pengiriman Notifikasi}}{\text{Jumlah Pengujian}} \quad (3.7)$$

Kemudian hitung varian dengan cara yaitu :

$$s^2 = \frac{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2}{n(n-1)} \quad (3.8)$$

Setelah nilai varian diketahui selanjutnya hitung nilai standar deviasi dengan cara yaitu :

$$s = \sqrt{\frac{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2}{n(n-1)}} \quad (3.9)$$

Pada pengujian ini juga menghitung nilai *Throughput* pada data yang dikirim tempat sampah pintar ke *Smartphone* dengan cara yaitu :

$$\text{Throughput} = \frac{\text{Jumlah data yang dikirim}}{\text{Waktu pengiriman data}} \quad (3.10)$$

Setelah nilai *Throughput* untuk 10 kali pengujian didapatkan, selanjutnya menghitung nilai rata-rata *Throughput* dengan cara yaitu :

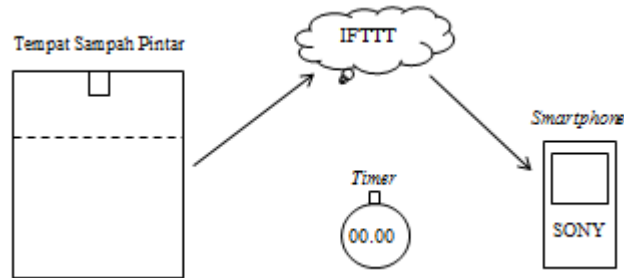
$$\text{Rata - rata} = \frac{\text{Jumlah Throughput}}{\text{Jumlah Pengujian}} \quad (3.11)$$

Kemudian hitung varian dengan cara yaitu :

$$s^2 = \frac{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2}{n(n-1)} \quad (3.12)$$

Setelah nilai varian diketahui selanjutnya hitung nilai standar deviasi dengan cara yaitu :

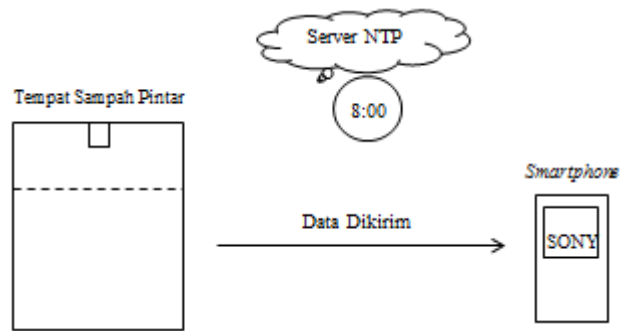
$$s = \sqrt{\frac{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2}{n(n-1)}} \quad (3.13)$$



Gambar 3.9 Perancangan Pengujian Waktu Penerimaan Notifikasi

4. Perancangan Pengujian Sinkronisasi Server NTP

Pada pengujian sinkronisasi server NTP dilakukan dengan membandingkan waktu pengiriman data pada tempat sampah pintar (sesuai waktu yang sudah ditentukan pada server NTP) dan saat notifikasi tersebut sampai di *Smartphone*. Apakah waktu saat tempat sampah pintar mengirim data sama dengan waktu saat notifikasi tersebut sampai di *Smartphone*. Data yang diambil pada pengujian ini yaitu waktu saat tempat sampah pintar mengirim data dengan waktu saat notifikasi sampai di *Smartphone*, dengan melakukan pengujian sebanyak 10 kali. Pengujian ini dilakukan setiap 1 jam sekali dan setiap 1 hari 2 kali. Untuk pengujian tiap 1 jam sekali dilakukan pada 10 waktu yang berbeda yaitu dari jam 8:00 sampai jam 17:00, sedangkan untuk pengujian tiap 1 hari 2 kali dilakukan pada 10 waktu berbeda pada jam 8:00 dan jam 16:00 selama 5 hari. Dengan ini maka kesesuaian antara waktu saat tempat sampah pintar mengirim data dan waktu saat notifikasi sampai di *Smartphone* dapat diketahui.



Gambar 3.10 Perancangan Pengujian Sinkronisasi Server NTP