



LAPORAN PROYEK INTERNET OF THINGS

**SMART WASTE BIN WITH REAL TIME
MONITORING SYSTEM USING WEB
APPLICATION**

Kisron	(2020191006)
Saifudin Usman	(2020191016)
Kevin Ilham Apriyandi	(2020191018)
Mohamad Walid Asyhari	(2020191021)

DOSEN PENGAMPU:

M. Udin Harun Al Rasyid, S.Kom., Ph.D.

**PROGRAM PASCASARJANA TERAPAN
POLITEKNIK ELEKTRONIKA NEGERI SURABAYA
2019**

KATA PENGANTAR



Alhamdulillah Rabbil'aalamin, segala puji syukur kepada Allah SWT karena atas segala taufik, rahmat, dan hidayah-Nya penulis dapat menyelesaikan proyek akhir internet of things yang berjudul :

SMART WASTE BIN WITH REAL TIME MONITORING SYSTEM USING WEB APPLICATION

Laporan proyek akhir ini disusun untuk memenuhi persyaratan penyelesaian mata kuliah Internet of Things program studi pascasarjana terapan teknik informatika dan komputer di Politeknik Elektronika Negeri Surabaya (PENS).

Proses pengerjaan dan penulisan laporan proyek akhir ini menggunakan beberapa literatur, baik dari perkuliahan maupun tidak. Selain itu, tidak lepas dari bimbingan dosen pengampu mata kuliah yang telah memberikan bimbingan serta arahan dalam penyelesaiannya.

Penulis sangat menyadari bahwa laporan proyek akhir ini masih banyak kekurangan. Oleh karena itu penulis memohon maaf atas kekurangan yang terdapat pada buku ini. Penulis juga tidak lupa untuk mengharapkan kritik dan saran dari semua pihak demi perbaikan selanjutnya.

Besar harapan penulis agar proyek akhir yang telah dikerjakan dan ditulis dalam laporan ini dapat bermanfaat khususnya bagi penulis sendiri, dan bermanfaat bagi pembaca lain untuk menambah pengetahuan dan wawasan.

Surabaya, 12 Desember 2019

Penyusun

BAB I

PENDAHULUAN

1.1 LATAR BELAKANG

Tingginya jumlah penduduk di Indonesia memicu tingkat konsumsi produk berupa barang kebutuhan seperti: makanan, minuman, dll yang sangat tinggi. Dampak nyata dari tingkat konsumsi yang tinggi tersebut akan menghasilkan sisa produk/barang yang dikategorikan sebagai sampah yang sangat tinggi. Sayangnya hal ini tidak diikuti dengan kesadaran masyarakatnya untuk membuang sampah pada tempatnya.

Kurangnya kesadaran masyarakat akan pentingnya membuang sampah pada tempatnya ini sebenarnya telah teridentifikasi dari tidak adanya pendidikan khusus dan pembiasaan membuang sampah pada tempatnya sejak usia dini. Diperparah lagi dengan kondisi masyarakat pada umumnya yang tidak peduli dengan kondisi lingkungan di sekitarnya. Hal ini bisa jadi disebabkan kurangnya implementasi dari pendidikan karakter dan moral di masyarakat, dimana kebiasaan-kebiasaan baik dan buruk biasanya hanya sebatas teori, tidak sampai pada tahap implementasi atau praktek.



Gambar 1. TPA Sampah di Benowo, Surabaya

Sumber : <http://www.lensaindonesia.com>

Di beberapa daerah di Indonesia, permasalahan pengelolaan sampah telah menjadi perhatian utama dari pemerintah daerah, diantaranya di Surabaya, Jambi, Balikpapan, dll yang mengeluarkan Perda (Peraturan Daerah) yang bertujuan untuk

mengatur pengelolaan sampah dan pemberian sanksi. Sebenarnya sampah memiliki potensi digunakan sebagai pembangkit listrik, barang daur ulang dsb. Namun, karena tidak adanya kesadaran dari anggota masyarakat secara umum, maka Perda yang telah dibuat dan disosialisasikan tersebut seolah tidak berdaya untuk menanggulangi permasalahan sampah.

Untuk mengurangi permasalahan tersebut akan dikembangkan sebuah SMART WASTE BIN WITH REAL-TIME MONITORING SYSTEM USING WEB APPLICATION. Pada sistem ini digunakan sensor ultrasonic untuk menghitung volume sampah dan wifi modul untuk pengiriman data ke cloud server. Cloud server yang digunakan adalah Amazon Web Service. Dari data yang disimpan di server nanti akan dibuat sebuah website untuk melakukan monitoring realtime. Ditambahkan juga shortest path untuk mengetahui jalur tercepat yang harus dilalui dalam mengambil sampah.

1.2 PERUMUSAN PERMASALAHAN

Dari latar belakang tersebut terdapat permasalahan bagaimana merancang sistem yang akan dibuat. Berikut rumusan masalah :

1. Bagaimana sistem dapat mendeteksi volume sampah pada tempat sampah?
2. Bagaimana sistem dapat memantau data volume tempat sampah secara real time?
3. Bagaimana sistem dapat memberikan jalur pengambilan tempat sampah tercepat?

1.3 BATASAN MASALAH

Batasan masalah untuk proyek yang akan dibuat adalah sebagai berikut :

1. Sensor Ultrasonik digunakan untuk mengukur volume sampah.
2. Koordinat posisi tempat sampah sudah ditentukan.

1.4 TUJUAN

Tujuan dari proyek yang akan dibuat adalah sebagai berikut :

1. Sistem dapat mendeteksi volume sampah.
2. Sistem dapat memantau tempat sampah secara real time.
3. Sistem dapat memberikan jalur tercepat untuk pengambilan tempat sampah.

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 PENELITIAN TERKAIT

Robert Eko Noegroho Sisyanto. dkk (2017) Dalam *Cyber Physical Social System* (CPSS), kerja kolaborasi antara petani hidroponik sekarang mungkin. Dengan konsep baru ini, sistem pertanian cerdas hidroponik itu dapat dipantau secara online melalui Telegram Messenger dikembangkan. Desain yang dibuat dapat memonitor parameter penting di sistem hidroponik, seperti intensitas cahaya, ruangan suhu, kelembaban, pH, *nutrient temperature*, dan *Electrical Conductivity*(EC). Prototipe dirancang menggunakan Raspberry Pi 3 yang terhubung langsung dengan sensor seperti modul DHT11, LDR, modul sensor pH, dan sensor EC. BOT telegram itu memungkinkan untuk memonitor sensor online melalui Telegram juga dibuat. Dengan integrasi Sistem Fisik (Raspberry Pi, sensor) dan Sistem Sosial (Telegram Messenger) terhubung secara online melalui internet atau cyber, sistem pemantauan hidroponik menjadi lebih fleksibel[1].

Norfadzlia Mohd Yusof, dkk (2018) membahas tentang proyek inovasi IoT dari tempat sampah pintar dengan sistem pemantauan waktu nyata yang mengintegrasikan banyak teknologi seperti tata surya, sensor, dan teknologi komunikasi nirkabel. Tujuan dari proyek ini adalah untuk menyediakan sistem pengelolaan pengumpulan limbah yang efisien dan hemat biaya sehingga memberikan lingkungan yang bersih, sehat dan hijau. Studi ini mengusulkan kerangka kerja baru yang memungkinkan pemantauan jarak jauh dari tempat sampah padat secara real-time melalui koneksi Wi-Fi, untuk membantu kegiatan pengelolaan limbah. Kerangka kerja sistem didasarkan pada jaringan sensor nirkabel [WSN] yang terdiri dari tiga segmen: sumber energi terbarukan, WSN dan stasiun kontrol. Dalam kerangka kerja ini ada empat subsistem yang dikembangkan: sistem tenaga surya, tempat sampah pintar, sistem pemberitahuan pesan singkat [SMS] dan sistem pemantauan real-time yang saling terkait satu sama lain untuk berfungsi sebagai manajemen limbah yang efisien, hemat biaya sistem yang menghasilkan lingkungan hidup yang hijau dan sehat[2].

Zainal Hisham Che Soh, dkk (2018) membahas tentang Pemantauan konsumsi air dan sistem peringatan via Internet of Things (IoT) Ubidots Cloud adalah salah satu platform untuk memantau jumlah penggunaan air di rumah tangga domestik per hari, per minggu dan per tahun secara real-time di ponsel atau laptop. Itu Ubidots Dashboard menyediakan antarmuka untuk menampilkan air data tingkat konsumsi dan juga dapat diilustrasikan menggunakan grafik. Grafik menunjukkan konsumsi air per hari di rumah tangga, ketika air mulai mengalir melalui sensor, sensor itu mulai mengukur volume air dan mengirim data ke Ubidots IoT peron. Menggunakan Ubidots Event, ketika data mencapai yang ditentukan batas konsumsi air yang ditentukan, peringatan pada yang berlebihan penggunaan air dikirim ke pengguna rumah melalui telegram atau email sedang menelepon[3].

Zhou Tao (2008) Untuk TSP (Traveling Salesman Problem) dari nomor kota, ada $(n-1)!$ Skema permutasi dari rute Tutup, maka TSP adalah NP-Complete problem. Sulit untuk menemukan solusi pengoptimalan terbaik untuk masalah seperti ini menggunakan metode pencarian global. Studi tentang menemukan algoritma ketersediaan yang sesuai untuk mengetahui optimasi terbaik atau perkiraan solusi optimasi menjadi sangat penting. Kalau tidak, banyak masalah praktis, seperti masalah membosankan kartu sirkuit cetak, masalah pengiriman barang dari beberapa toko, semua menjadi masalah TSP melalui pendahuluan. Saat ini, metode untuk memecahkan masalah TSP adalah metode pencarian elitasi, algoritma anneal simulasi, algoritma Genetik, algoritma jaringan Hopfield Neural. GA adalah algoritma pencarian global yang relatif efektif untuk menyelesaikan masalah NP-complete semacam ini[4].

Ma Jingyan dan Zhang Kehong (2012) Ada banyak metode algoritma genetika untuk menyelesaikan masalah. Penelitian ini mengedepankan operator selektif baru yang dapat meningkatkan algoritma genetika dengan informasi heuristik berdasarkan perubahan karakter informasi kombinatorial dalam masalah TSP dan persamaan Logistik. Metode ini dapat membantu algoritma genetika untuk memecahkan masalah TSP. Penelitian ini juga berusaha memberikan beberapa cara untuk menghindari masalah optimasi sebagian dari algoritma genetika[5].

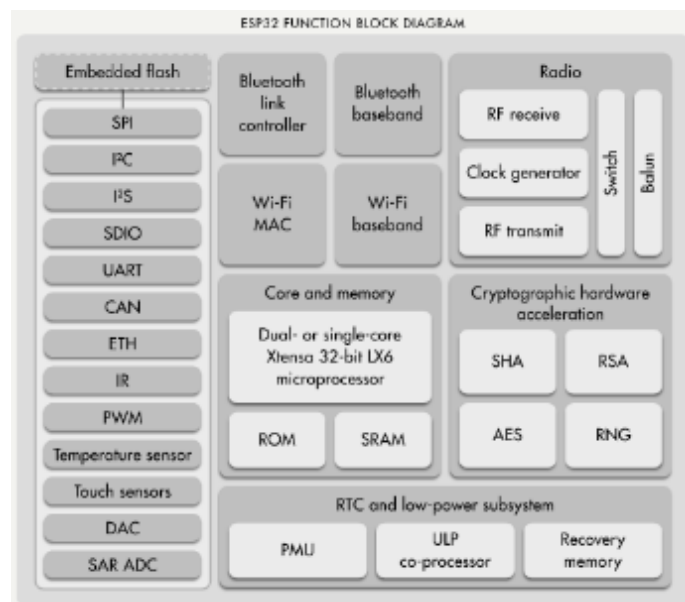
2.2 DASAR TEORI

2.2.1 NodeMCU ESP32

ESP32 dibuat oleh *Espressif Systems*, **ESP32** adalah sistem dengan biaya yang rendah, berdaya rendah pada seri chip (SoC) dengan Wi-Fi & kemampuan Bluetooth dua mode! Keluarga ESP32 termasuk chip ESP32-D0WDQ6 (dan ESP32-D0WD), ESP32-D2WD, ESP32-S0WD, dan sistem dalam paket (SiP) ESP32-PICO-D4. Pada intinya, ada mikroprosesor Tensilica Xtensa LX6 *dual-core* atau *single-core* dengan *clock rate* hingga 240 MHz. ESP32 sudah terintegrasi dengan *built-in antenna switches, RF balun, power amplifier, low-noise receive amplifier, filters, and power management modules*. Didesain untuk perangkat seluler, perangkat elektronik yang dapat dipakai, dan aplikasi IoT, ESP32 juga bekerja dengan konsumsi daya sangat rendah melalui fitur hemat daya termasuk *fine resolution clock gating, multiple power modes, and dynamic power scaling*.

Module ESP32 merupakan penerus dari module ESP8266 yang cukup populer untuk Aplikasi IoT. Pada ESP32 terdapat inti CPU serta Wi-Fi yang lebih cepat, GPIO yang lebih, dan mendukung *Bluetooth Low Energy*.

Berikut ini adalah Arsitektur dan Block Diagram dari ESP32 [6]:



Gambar 2. Arsitektur dan Block Diagram ESP32

Sumber : www.edukasi elektronik a.com

Fitur dan Spesifikasi ESP32 :

1. Processors:

- > Main processor: Tensilica Xtensa 32-bit LX6 microprocessor
 - > Cores: 2 or 1 (depending on variation)
 - > Clock frequency: up to 240 MHz
 - > Performance: up to 600 DMIPS
- > Ultra low power co-processor: allows you to do ADC conversions, computation, and level thresholds while in deep sleep.

2. Wireless connectivity:

- > Wi-Fi: 802.11 b/g/n/e/i (802.11n @ 2.4 GHz up to 150 Mbit/s)
- > Bluetooth: v4.2 BR/EDR and Bluetooth Low Energy (BLE)

3. Memory:

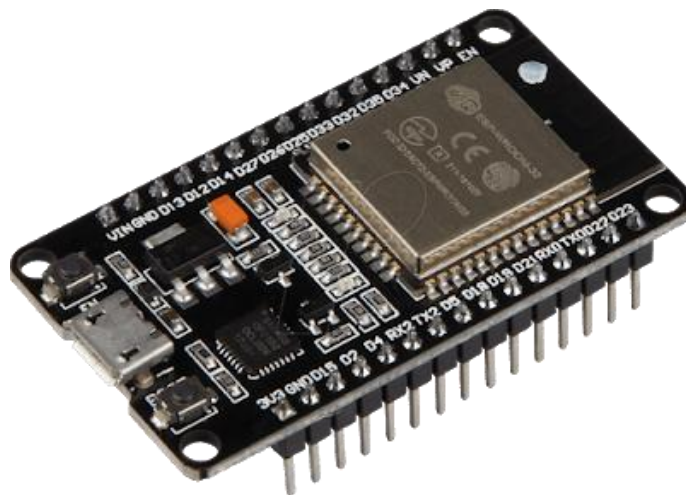
- > Internal memory:
 - > ROM: 448 KiB
 - > SRAM: 520 KiB
 - > RTC fast and slowSRAM: 8 KiB
 - > eFuse: 1 Kibit
- > Embedded flash:
 - > 0 MiB (ESP32-D0WDQ6, ESP32-D0WD, and ESP32-S0WD chips)
 - > 2 MiB (ESP32-D2WD chip)
 - > 4 MiB (ESP32-PICO-D4 SiP module)
- > External flash & SRAM: supports up to four 16 MiB external QSPI flashes and SRAMs with hardware encryption based on AES to protect developers' programs and data. ESP32 can access the external QSPI flash and SRAM through high-speed caches.
- > Up to 16 MiB of external flash memory-mapped onto the CPU code space, supporting 8-bit, 16-bit and 32-bit access. Code execution is supported.
- > Up to 8 MiB of external SRAM memory are mapped onto the CPU data space, supporting 8-bit, 16-bit and 32-bit access. Data-read is supported on the flash and SRAM. Data-write is supported on the SRAM.

4. *Peripheral input/output*: Rich peripheral interface with DMA that includes capacitive touch, ADCs (analog-to-digital converter), DACs (digital-to-analog converter), I²C (Inter-Integrated Circuit), UART (universal asynchronous receiver/transmitter), CAN 2.0 (Controller Area Network), SPI (Serial Peripheral Interface), I²S (Integrated Inter-IC Sound), RMII (Reduced Media-Independent Interface), PWM (pulse width modulation), and more.

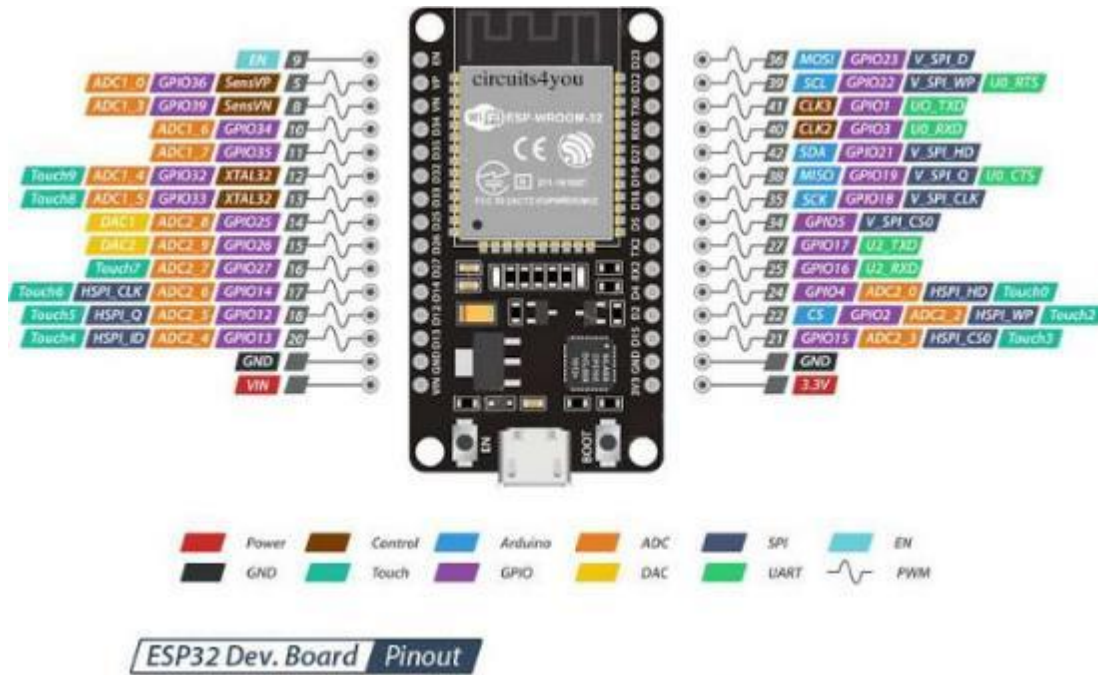
5. *Security*:

- > IEEE 802.11 standard security features all supported, including WPA, WPA/WPA2 and WAPI
- > Secure boot
- > Flash encryption
- > 1024-bit OTP, up to 768-bit for customers
- > Cryptographic hardware acceleration: AES, SHA-2, RSA, elliptic curve cryptography (ECC), random number generator (RNG).

Berikut ini adalah bentuk Fisik Module ESP32 :



Berikut ini adalah Pin Out Module ESP32 :

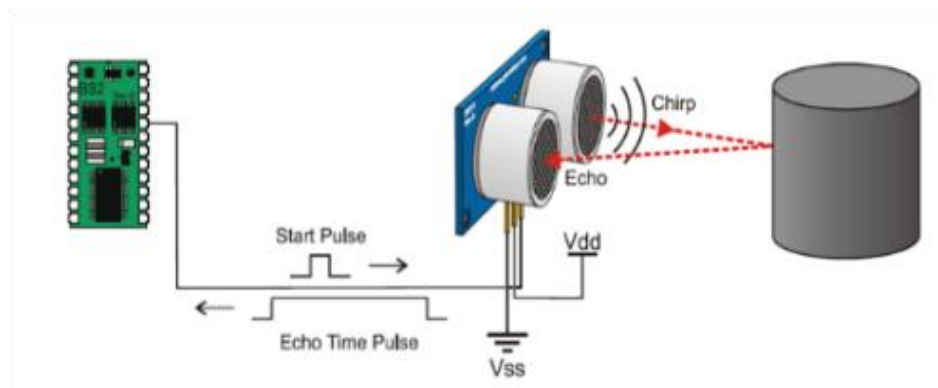


Gambar 4. Pinout Module ESP32

Sumber : www.edukasiaelektronika.com

2.2.2 Sensor Ultrasonik HY-SR05

Sensor ultrasonik adalah sensor yang bekerja berdasarkan prinsip pantulan gelombang suara dan digunakan untuk mendeteksi keberadaan suatu objek tertentu di depannya[7]. Sensor ultrasonik terdiri dari dua unit, yaitu unit pemancar dan unit penerima. Untuk lebih jelas tentang prinsip kerja dari sensor ultrasonik dapat dilihat prinsip dari sensor ultrasonik pada gambar 5.



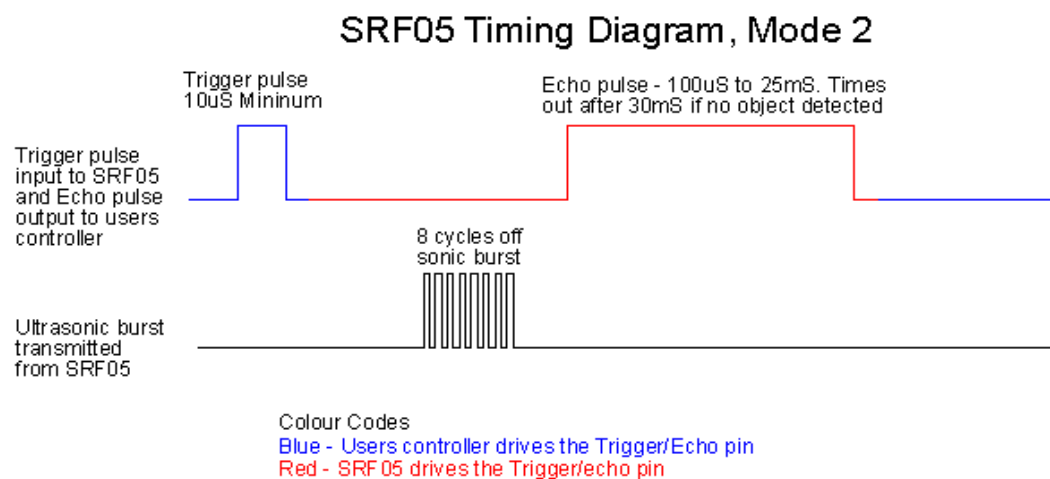
Gambar 5. Prinsip Kerja Sensor Ultrasonik

Sumber : www.arduinosensors.com

Besar amplitudo sinyal elektrik yang dihasilkan sensor penerima tergantung dari jauh dekatnya objek yang dideteksi serta kualitas sensor. Proses sensing yang dilakukan pada sensor ini menggunakan metode pantulan untuk menghitung jarak antara sensor dengan objek. Jarak antara sensor tersebut dihitung dengan cara mengalikan 1/2 waktu yang digunakan oleh sinyal ultrasonik dalam perjalanannya dari rangkaian pengirim sampai diterima oleh rangkaian penerima, dengan kecepatan rambat dari sinyal ultrasonik tersebut dengan media udara. Timing diagram (gambar 6) digunakan untuk mengatur pembacaan data. Semakin cepat timing-nya maka pembacaan data juga semakin cepat.

Karakteristik sensor ultrasonik SRF-05 :

- Tegangan sumber operasi tunggal 5 VDC arus 30 – 50 mA
- Frekuensi operasi 40 KHz
- *Trigger input* minimal 10uS
- Mampu mendeteksi jarak dari 0.03 – 4 m
- Sudut pantul gelombang pengukuran 15 derajat
- Minimum waktu penyulutan 10 mikrodetik dengan pulsa berlevel TTL
- Pulsa deteksi berlevel TTL dengan durasi yang bersesuaian dengan jarak deteksi
- Dimensi 45 x 20 x 15 mm

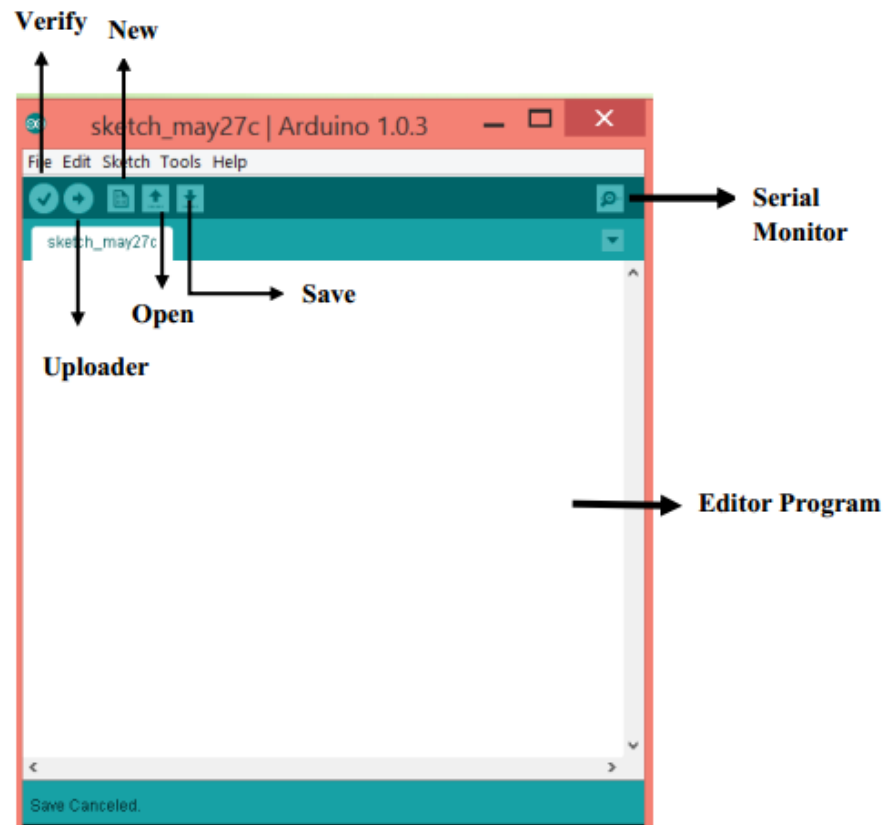


Gambar 6. Timing Diagram Sensor SRF05

Sumber : www.robot-electronics.co.uk

2.2.3 Arduino IDE

IDE atau *Integrated Development Environment* merupakan suatu program khusus untuk suatu komputer agar dapat membuat suatu rancangan atau sketsa program untuk papan Arduino. IDE arduino merupakan *software* yang sangat canggih ditulis dengan menggunakan java. IDE arduino terdiri dari :



Gambar 7. Tampilan Arduino IDE

Sumber : Dokumentasi Penulis

Keterangan :

1. Editor Program

Sebuah *window* yang memungkinkan pengguna menulis dan mengedit program dalam bahasa *processing*.

2. *Verify*

Mengecek kode sketch yang *error* sebelum mengupload ke *board* arduino.

3. *Uploader*

Sebuah modul yang memuat kode biner dari komputer ke dalam memori di dalam papan arduino.

4. *New*

Membuat sebuah sketch baru.

5. *Open*

Membuka daftar sketch pada *sketchbook* arduino.

6. *Save*

Menyimpan kode sketch pada *sketchbook*.

7. Serial Monitor

Menampilkan data serial yang dikirimkan dari *board* arduino.

2.2.4 Notepad++

Notepad++ adalah sebuah *text editor* yang sangat berguna bagi setiap orang dan khususnya bagi para developer dalam membuat program. Notepad++ menggunakan komponen *Scintilla* untuk dapat menampilkan dan menyuntingan teks dan berkas kode sumber berbagai bahasa pemrograman yang berjalan diatas sistem operasi Microsoft Windows. Selain manfaat dan kemampuannya menangani banyak bahasa pemrograman, Notepad++ juga dilisensikan sebagai perangkat *free*. Jadi, setiap orang yang menggunakannya tidak perlu mengeluarkan biaya untuk membeli aplikasi ini karena sourceforge.net sebagai layanan yang memfasilitasi Notepad++ membebaskannya untuk digunakan. Beberapa daftar bahasa program yang didukung oleh Notepad++ adalah C, C++, Java, C#, XML, HTML, PHP, Javascript. Sebenarnya masih banyak lagi bahasa program yang didukung, namun penulis baru mencoba Notepad++ dengan bahasa program yang diatas.

2.2.5 MySQL

Database adalah suatu kumpulan data-data yang disusun sedemikian rupa sehingga membentuk informasi yang sangat berguna. *Database* terbentuk dari sekelompok data-data yang memiliki jenis/sifat yang sama. Ambil contoh, data-data berupa nama-nama, kelas-kelas, alamat-alamat. Semua data tersebut dikumpulkan menjadi satu menjadi kelompok data baru, sebut saja sebagai data-data mahasiswa. Demikian juga, kumpulan dari data-data mahasiswa, data-data dosen, data-data keuangan dan lainnya dapat dikumpulkan lagi menjadi kelompok besar, misalkan data-data politeknik elektronika. Bahkan dalam perkembangannya, data-data tersebut dapat berbentuk berbagai macam data, misalkan dapat berupa

program, lembaran-lembaran untuk *entry* (memasukkan) data, laporan-laporan. Kesemuanya itu dapat dikumpulkan menjadi satu yang disebut dengan database. Salah satu bahasa database yang populer adalah SQL. MySQL biasa dibaca *mai-es-ki-el* atau *mai-se-kuel* adalah suatu perangkat lunak *database* relasi (*Relational Database Management System* atau RDBMS) seperti halnya Oracle, Postgresql, MS SQL dan sebagainya.

SQL atau singkatan dari *Structured Query Language* ialah suatu sintaks perintah-perintah tertentu atau bahasa pemrograman yang digunakan untuk mengelola suatu *database*. Jadi, MySQL dan SQL tidaklah sama. Singkatnya, MySQL ialah perangkat lunaknya dan SQL adalah bahasa perintahnya. Ketika dibandingkan antara MySQL dengan sistem *database* yang lain, maka perlu difikirkan apa yang paling penting sesuai kebutuhan. Apakah tampilan, *support*, fitur-fitur SQL, kondisi keamanan dalam lisensi, atau masalah harga. Dengan pertimbangan tersebut, MySQL memiliki banyak hal yang bisa ditawarkan, antara lain :

1. Berdasarkan kecepatannya, banyak ahli memberikan pendapat bahwa MySQL merupakan *server* tercepat.
2. MySQL memiliki performa tinggi namun merupakan *database* yang simpel sehingga mudah di-*setup* dan dikonfigurasi.
3. MySQL cenderung gratis untuk penggunaan tertentu.
4. MySQL mengerti bahasa SQL (*Structured Query Language*) yang merupakan pilihan sistem *database* moderen.
5. Banyak klien dapat mengakses *server* dalam satu waktu. Mereka dapat menggunakan banyak *database* secara simultan.
6. *Database* MySQL dapat diakses dari semua tempat di internet dengan hak akses tertentu.
7. MySQL dapat berjalan dalam banyak varian Unix dengan baik, sebaik seperti saat berjalan di sistem non-Unix.
8. MySQL mudah didapatkan dan memiliki *source code* yang boleh disebarluaskan sehingga bisa dikembangkan lebih lanjut.
9. Dapat dikoneksikan pada bahasa C, C++, Java, Perl, PHP dan Python

Jika hal-hal diatas ialah kelebihan yang dimiliki oleh MySQL, maka MySQL juga memiliki kekurangan seperti :

1. Untuk koneksi ke bahasa pemrograman visual seperti visual basic, delphi, dan foxpro, MySQL kurang mendukung. Karena koneksi ini menyebabkan *field* yang dibaca harus sesuai dengan koneksi dari program visual tersebut. Dan ini yang menyebabkan MySQL jarang dipakai dalam program visual.
2. Data yang ditangani belum begitu besar

2.2.6 HTTP POST

Dalam komputasi, POST adalah metode permintaan yang didukung oleh HTTP yang digunakan oleh World Wide Web. Secara desain, metode permintaan POST meminta server web menerima data yang disertakan dalam isi pesan permintaan, kemungkinan besar untuk menyimpannya. Ini sering digunakan saat mengunggah file atau saat mengirimkan formulir web yang sudah diisi.

Sebaliknya, metode permintaan GET HTTP mengambil informasi dari server. Sebagai bagian dari permintaan GET, beberapa data dapat dikirimkan dalam string kueri URL, menentukan (misalnya) istilah pencarian, rentang tanggal, atau informasi lain yang mendefinisikan kueri.

Sebagai bagian dari permintaan POST, jumlah data jenis apa pun yang sewenang-wenang dapat dikirim ke server di badan pesan permintaan. Bidang tajuk dalam permintaan POST biasanya menunjukkan jenis media Internet badan pesan.

POST digunakan untuk mengirim data ke server untuk membuat / memperbarui resource. Data yang dikirim ke server dengan POST disimpan di body request of the HTTP request :

```
POST /test/demo_form.php HTTP/1.1
```

```
Host: w3schools.com
```

```
name1=value1&name2=value2
```

POST adalah salah satu metode HTTP yang paling umum. Beberapa catatan lain tentang POST request:

- POST request tidak pernah di-cache

- POST request tidak tetap dalam riwayat browser
- POST request tidak dapat di-bookmark
- POST request tidak memiliki batasan panjang data

2.2.7 AWS EC2 Instance

Amazon Elastic Compute Cloud (Amazon EC2) adalah layanan web yang menyediakan kapasitas komputasi yang aman dan dapat diubah ukurannya di cloud. Ini dirancang untuk membuat komputasi awan skala web lebih mudah bagi pengembang.

Antarmuka layanan web sederhana Amazon EC2 memungkinkan Anda untuk mendapatkan dan mengkonfigurasi kapasitas dengan gesekan minimal. Ini memberi Anda kendali penuh atas sumber daya komputasi Anda dan memungkinkan Anda berjalan di lingkungan komputasi Amazon yang telah terbukti. Amazon EC2 mengurangi waktu yang diperlukan untuk mendapatkan dan mem-boot instance server baru menjadi beberapa menit, memungkinkan Anda untuk dengan cepat meningkatkan kapasitas, baik naik maupun turun, seiring perubahan kebutuhan komputasi Anda. Amazon EC2 mengubah ekonomi komputasi dengan memungkinkan Anda membayar hanya untuk kapasitas yang sebenarnya Anda gunakan. Amazon EC2 memberikan pengembang alat untuk membangun aplikasi yang tahan terhadap kegagalan dan mengisolasi dari skenario kegagalan yang umum.

Fitur dari Amazon EC2 :

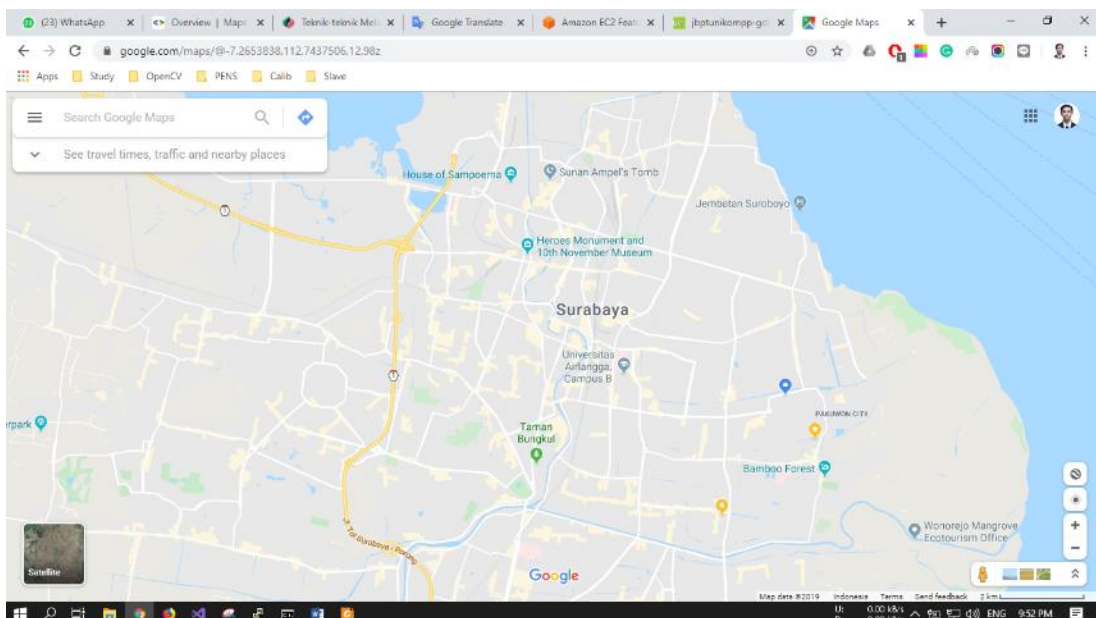
- Bare Metal instances
- Optimize Compute Performance and Cost with Amazon EC2 Fleet
- Pause and Resume Your Instances
- GPU Compute Instances
- GPU Graphics Instances
- High I/O Instances
- Dense HDD Storage Instances

- Optimized CPU Configurations
- Flexible Storage Options

2.2.8 API Google Maps

Maps Static API memungkinkan Anda menyematkan gambar Google Maps di halaman web Anda tanpa memerlukan JavaScript atau pemuatan halaman dinamis. Layanan Maps Static API membuat peta Anda berdasarkan parameter URL yang dikirim melalui permintaan HTTP standar dan mengembalikan peta sebagai gambar yang dapat Anda tampilkan di halaman web Anda.

GoogleMaps adalah peta online atau membuka peta secara online, dapat dilakukan secara mudah melalui layanan gratis dari Google. Bahkan layanan ini menyediakan API (Application Programming Interface) yang memungkinkan developer lain untuk memanfaatkan aplikasi ini di aplikasi buatannya. Tampilan GoogleMaps pun dapat dipilih, berdasarkan foto asli atau peta gambar rute saja. GoogleMaps adalah layanan gratis yang diberikan oleh Google dan sangat populer. GoogleMaps adalah suatu peta dunia yang dapat kita gunakan untuk melihat suatu daerah seperti pada gambar 2. Dengan kata lain, GoogleMaps merupakan suatu peta yang dapat dilihat dengan menggunakan suatu browser.



Gambar 8. Tampilan Google Maps

Sumber : maps.google.com

Kita dapat menambahkan fitur GoogleMaps dalam web yang telah kita buat atau pada blog kita yang berbayar maupun gratis sekalipun dengan GoogleMaps API. GoogleMaps API adalah suatu library yang berbentuk JavaScript. Cara membuat GoogleMaps untuk ditampilkan pada suatu web atau blog sangat mudah hanya dengan membutuhkan pengetahuan mengenai HTML serta JavaScript, serta koneksi Internet yang sangat stabil. Dengan menggunakan GoogleMaps API, kita dapat menghemat waktu dan biaya untuk membangun aplikasi peta digital yang handal, sehingga kita dapat fokus hanya pada data-data yang akan ditampilkan. Dengan kata lain, kita hanya membuat suatu data sedangkan peta yang akan ditampilkan adalah milik Google sehingga kita tidak dipusingkan dengan membuat peta suatu lokasi, bahkan dunia.

Pada GoogleMaps API terdapat 4 jenis pilihan model peta yang disediakan oleh Google, diantaranya adalah:

- a. ROADMAP, untuk menampilkan peta biasa 2 dimensi.
- b. SATELLITE, untuk menampilkan foto satelit.
- c. TERRAIN, untuk menunjukkan relief fisik permukaan bumi dan menunjukkan seberapa tingginya suatu lokasi, contohnya akan menunjukkan gunung dan sungai.
- d. HYBRID, akan menunjukkan foto satelit yang diatasnya tergambar pula apa yang tampil pada ROADMAP (jalan dan nama kota).

2.2.10 GA for TSP

Algoritma Genetika adalah teknik metaheuristik yang terinspirasi oleh prinsip seleksi alam. Algoritma ini digerakkan oleh tiga operator utama: seleksi, crossover dan mutasi. Reproduksi individu dilakukan oleh crossover dan operator mutasi. Operator seleksi adalah operator penggerak utama di GA untuk memilih individu terbaik dari populasi. GA berfungsi sebagai berikut: Jumlah individu yang dihasilkan secara acak yang membentuk populasi awal algoritma. Individu dapat direpresentasikan dalam bilangan biner, bilangan real, bilangan integer, karakter tergantung pada masalahnya. TSP menggunakan representasi permutasi. Pengukuran kebaikan untuk individu dilaksanakan melalui fungsi fitness.

Fungsi fitness untuk TSP adalah untuk memaksimalkan kebalikan dari fungsi objektif. Setiap individu dipindahkan dari satu generasi ke generasi berikutnya berdasarkan nilai fitness, lebih banyak nilai fitness, peluang untuk berpartisipasi dalam generasi yang akan datang adalah tinggi. Operator crossover memfasilitasi eksplorasi sementara mutasi memberikan eksploitasi. Jumlah operasi crossover dikendalikan oleh tingkat crossover.

Dalam operasi mutasi, titik acak dipilih dalam kromosom dan kemudian ganti nilai pada titik ini dengan beberapa nilai acak yang layak. Jumlah operasi mutasi dikendalikan oleh tingkat mutasi. Satu generasi algoritma genetika terdiri dari penggunaan operasi GA yaitu reproduksi, crossover dan kemudian mutasi. Setiap generasi menghasilkan jumlah individu. Pengakhiran algoritma mengembalikan individu dengan nilai fitness terbaik sebagai solusi. Rincian operator seleksi, crossover dan mutasi algoritma genetika dijelaskan dalam menemukan ruang optimal lokal untuk menemukan global optimal[8].

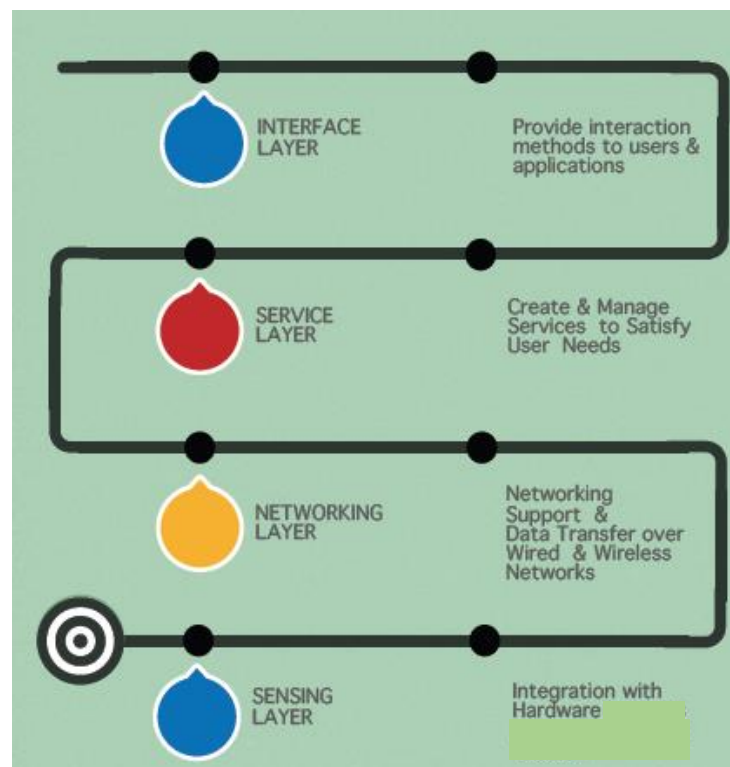
BAB III

DESAIN SISTEM

Pada bagian ini akan dijelaskan mengenai desain sistem layer IoT, Topologi, flowcart, pseudo code/source code utama, database, dan penjelasan metode pengerjaan detail.

3.1 Desain Sistem Layer IOT

Sistem layer IoT yang digunakan dalam proyek ini ada 4 seperti gambar 9 berikut ini.



Gambar 9. Sistem Layer IoT

Sumber : Dokumentasi Penulis

Berikut ini penjelasan masing – masing layer :

a. Sensing Layer

Pada layer ini dilakukan pembacaan sensor dan kalibrasi sampai mendapatkan nilai pembacaan dengan akurasi tinggi. Pada layer ini juga diatur penjadwalan untuk pembacaan sensor yaitu dilakukan setiap 5 menit sekali dan sekali pengambilan data dilakukan 5 kali pengambilan kemudian dilakukan rata rata.

b. Networking Layer

Layer ini digunakan dalam pengiriman data hasil pembacaan sensor kedalam cloud database yang sudah disediakan. Metode pengiriman data yang digunakan adalah menggunakan koneksi wireless.

c. Service Layer

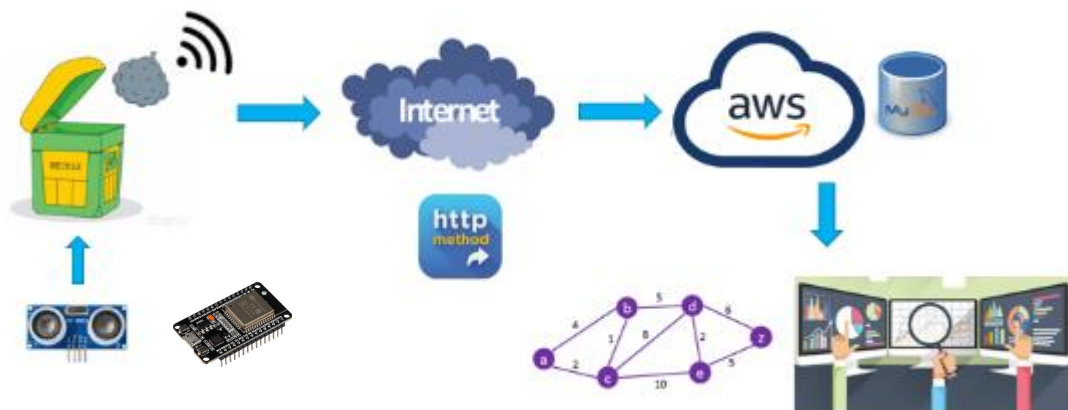
Service layer disini menyediakan sumber daya di cloud seperti database, web server. Pada proyek ini database yang digunakan adalah mysql database sedangkan web server yang digunakan adalah apache yang diinstall apa Amazon Web Service Instance EC2.

d. Interface Layer

Interface layer ini berisi visualisasi dari data yang telah dikirim dari node ke database. Visualisasi yang digunakan yaitu menggunakan website dengan menggunakan API google maps sehingga dapat memantau lokasi menggunakan maps tersebut. Disediakan juga fitur TSP untuk mendapatkan jalur tercepat dalam pengambilan sampah oleh petugas sampah.

3.2 Topologi

Topologi yang digunakan dalam proyek ini ditunjukkan pada gambar 10.



Gambar 10. Topologi IoT

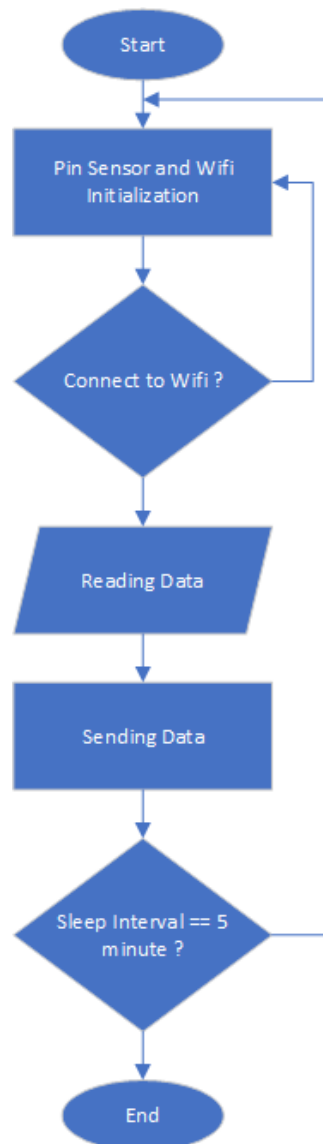
Sumber : Dokumentasi Penulis

Pada sistem ini digunakan sensor ultrasonik untuk mendapatkan volume sampah, setelah mendapatkan informasi tentang volume sampah sistem akan mengirimkan informasi tersebut ke cloud server. Pengiriman data pada sistem yang dibuat menggunakan koneksi wireless dengan HTTP Method. Cloud server yang

digunakan adalah Google Cloud Platform dengan database MySQL. Selanjutnya data tersebut akan diolah dan ditampilkan dalam aplikasi web untuk memudahkan pengguna untuk memantau volume sampah pada tempat sampah. Ditambahkan shortest path juga untuk memberikan informasi jalur tercepat dalam pengambilan sampah.

3.3 Flowchart

Berikut ini merupakan flowchart yang digunakan untuk melakukan pembacaan sensor dan pengiriman data sensor ke cloud :



Gambar 11. Flowchart Pembacaan dan Pengiriman Sensor

Sumber : Dokumentasi Penulis

3.4 Pseudo Code

Pseudo code ini berisi program yang digunakan untuk menyelesaikan proyek mulai dari program esp32 mengirim ke database sampai menampilkan kedalam website.

```
const int trigPin = 26;
const int echoPin = 27;

// Buat object Wifi
const char* ssid = "ssid";
const char* password = "password";

HTTPClient http;
String url = "http://3.89.128.93/smarttrash/Api?id=";
String payload;

float d1, d2, d3, d4, d5;
long tinggi, level, average;
long deviasi = 10;
long tinggiBaks = 150;
int sleepInterval = 300; //DEEP_SLEEP interval = 5m

void setup(){
  USE_SERIAL.begin(115200);
  USE_SERIAL.setDebugOutput(false);
  for(uint8_t t = 4; t > 0; t--) {
    USE_SERIAL.printf("[SETUP] Tunggu %d...\n", t);
    USE_SERIAL.flush();
    delay(1000);
  }
  konekToWifi();
  Serial.println("Levelling...");
  tinggi = ukurTinggi();
  average = ukurAverage();
  Serial.println(average);
  level = ukurLevel(average);
  Serial.print("#Tinggi Sampah (CM) = ");
  Serial.println(tinggiBaks - average);
  Serial.print("#Level (%) = ");
  Serial.println(level);
  kirimData();
  Serial.println("Goto sleep for 5 minute...");
  gotoSleep();
  delay(100);
}
```

```

float ukurTinggi(){
    pinMode(trigPin, OUTPUT);
    pinMode(echoPin, INPUT);
    long duration, distance;
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(1000);
    digitalWrite(trigPin, LOW);
    duration = pulseIn(echoPin, HIGH);
    distance = (duration / 2) / 29.1;
    return distance;
}

float ukurAverage(){
    d1 = ukurTinggi();
    delay(2000);
    d2 = ukurTinggi();
    delay(2000);
    d3 = ukurTinggi();
    delay(2000);
    d4 = ukurTinggi();
    delay(2000);
    d5 = ukurTinggi();
    return (d1 + d2 + d3 + d4 + d5)/ 5;
}

float ukurLevel(float ukurAverage){
    long persen;
    persen = 100-(((ukurAverage-deviasi)*100)/tinggiBaks);
    return persen;
}

void konekToWifi(){
    Serial.println("- Coba Koneksi WIFI SSID : " + String(ssid));
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println();
    Serial.println("- Koneksi Sukses");
}

void kirimData(){
    if((WiFi.status() == WL_CONNECTED)) {
        USE_SERIAL.print("[HTTP] Memulai...\n");
        http.begin( url + 2 + "&ketinggian=" + level);
    }
}

```



```

delay(2000);
USE_SERIAL.print("[HTTP] Melakukan GET ke server...\n");
int httpCode = http.GET();
if(httpCode > 0) {
    USE_SERIAL.printf("[HTTP] kode response GET: %d\n", httpCode);
    if(httpCode == HTTP_CODE_OK) {
        payload = http.getString();
        USE_SERIAL.println(payload);
    }
} else {
    USE_SERIAL.printf("[HTTP] GET gagal, error: %s\n", http.errorT
oString(httpCode).c_str());
}
http.end();
}
delay(5000);
}

void gotoSleep(){
    ESP.deepSleep(1000000 * sleepInterval);
}

```

Pseudo code Api.php untuk menerima data dari node dan disimpan kedalam database:

```

<?php
require APPPATH . '/libraries/REST_Controller.php';
Class Api Extends REST_Controller{
    function index_get(){
        $data=array(
            "id_sensor" =>$_GET['id'],
            "ketinggian" =>$_GET['ketinggian'],
        );
        $insert = $this->db->insert('data_ketinggian',$data);

        if($insert){
            $data = array ('status'=>'success insert');
            $this->response($data,200);
        }else{
            $data = array ('status'=>'gagal insert');
            $this->response($data,502);
        }
    }
}
}

```

Pseudo code untuk menampilkan api google maps :

```
<script type="text/javascript">
$(function() {
  var url;
  url= "<?php echo base_url('Data_ketinggian/getAlat'); ?>"

  $.ajax(
  {
    type: "GET",
    url: url,
  }
  ).done(function( data )
  {
    var jsonObj = JSON.parse(data);
    var count = Object.keys(jsonObj).length;
    setTimeout(function () {
      for (var i = 0; i < count; i++) {
        var lat_lng = new google.maps.LatLng(jsonObj[i].lt,jsonObj[i].lg);
        var nama =""+jsonObj[i].lokasi+"";
        var a = i+1;
        var sContent = '<h4>'+jsonObj[i].lokasi+'</h4>';

        addMarker(nama,lat_lng,sContent,a*200,jsonObj[i].id_sensor,jsonObj[i].sta
tus);
      }
    }, 3000);
  });

  .....
  .....
  .....
  .....

  * bagian ini sengaja dihilangkan
```

Pseudo code untuk melakukan pencarian jalur tercepat :

```
// Start GA
$('#find-route').click(function() {
  if (nodes.length < 2) {
    if (prevNodes.length >= 2) {
      nodes = prevNodes;
    } else {
      alert("Tempat sampah masih aman !!!");
      return;
    }
  }
}
```

```

if (directionsDisplay != null) {
    directionsDisplay.setMap(null);
    directionsDisplay = null;
}

$('#ga-buttons').hide();

// Get route durations
getDurations(function(){
    $('#ga-info').show();

    // Get config and create initial GA population
    ga.getConfig();
    var pop = new ga.population();
    pop.initialize(nodes.length);
    var route = pop.getFittest().chromosome;

    ga.evolvePopulation(pop, function(update) {
        $('#generations-passed').html(update.generation);
        $('#best-time').html((update.population.getFittest().getDistance() /
60).toFixed(2) + ' Mins');

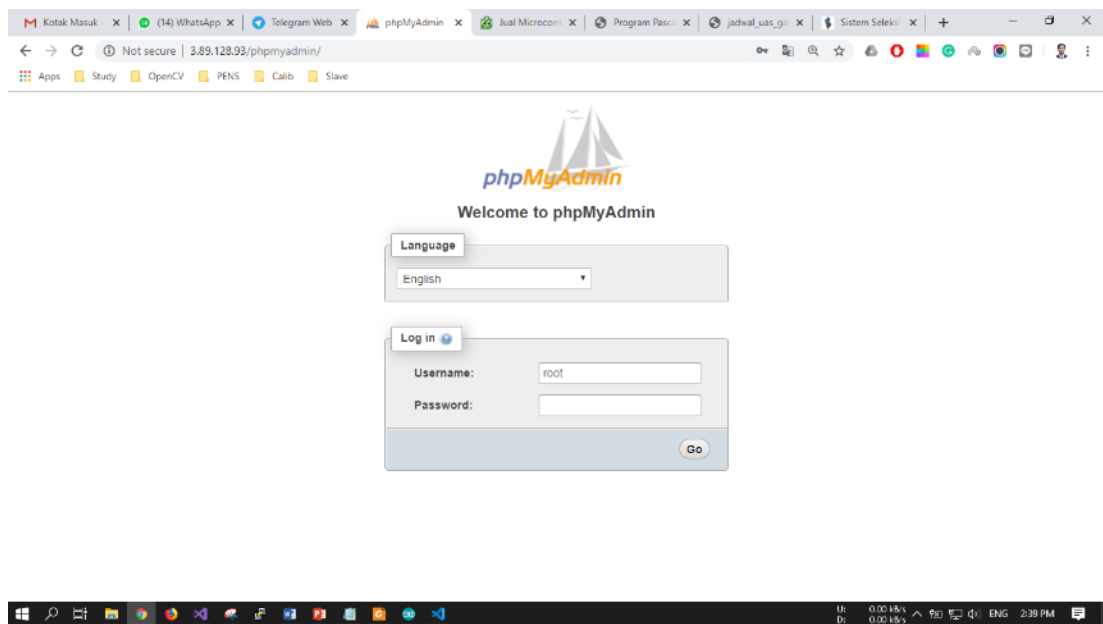
        // Get route coordinates
        var route = update.population.getFittest().chromosome;
        var routeCoordinates = [];
        for (index in route) {
            routeCoordinates[index] = nodes[route[index]];
        }
        routeCoordinates[route.length] = nodes[route[0]];
// Add route to map
        directionsService = new google.maps.DirectionsService();
        directionsDisplay = new google.maps.DirectionsRenderer();
        directionsDisplay.setMap(map);
        var waypts = [];
        for (var i = 1; i < route.length; i++) {
            waypts.push({
                location: nodes[route[i]],
                stopover: true
            });
        }

        .....
        .....
        .....
        .....
    * bagian ini sengaja dihilangkan

```

3.5 Perancangan Database

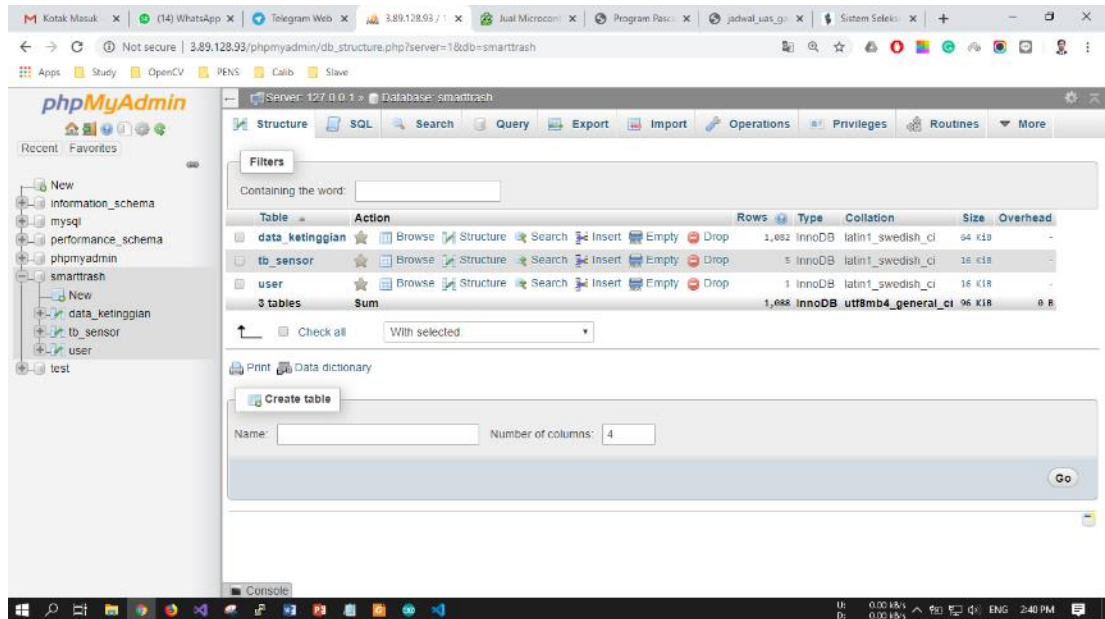
Database yang digunakan dalam project ini adalah mySQL. Untuk mengakses database tersebut digunakan phpMyAdmin. Gambar 12 merupakan tampilan pada saat membuka aplikasi phpmyadmin yaitu diminta memasukkan username dan password.



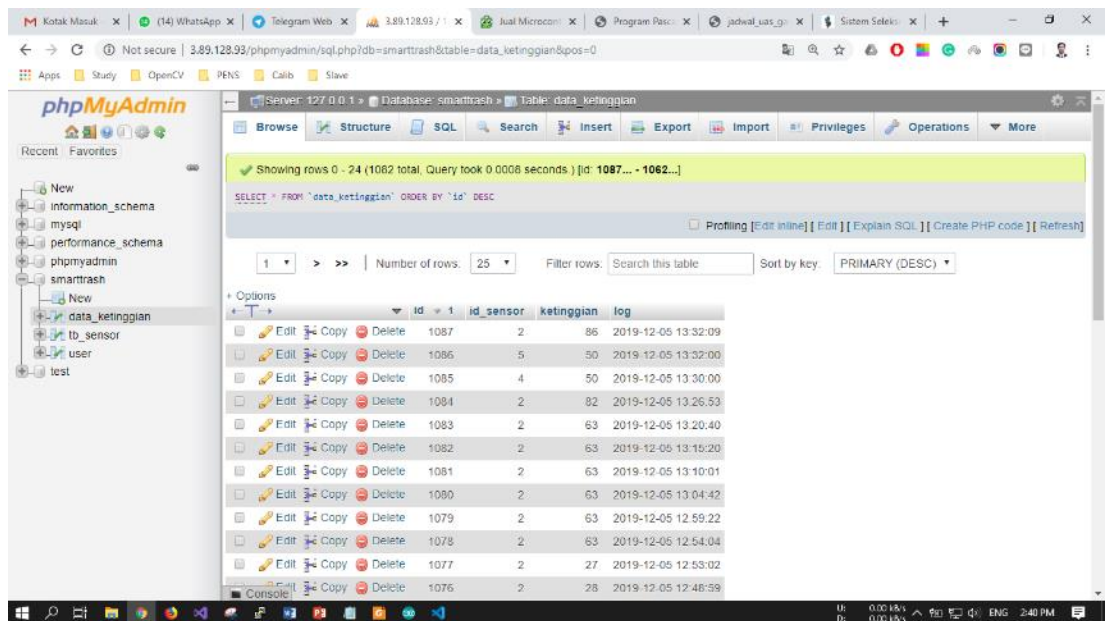
Gambar 12. Tampilan awal phpmyadmin

Sumber : Dokumentasi Penulis

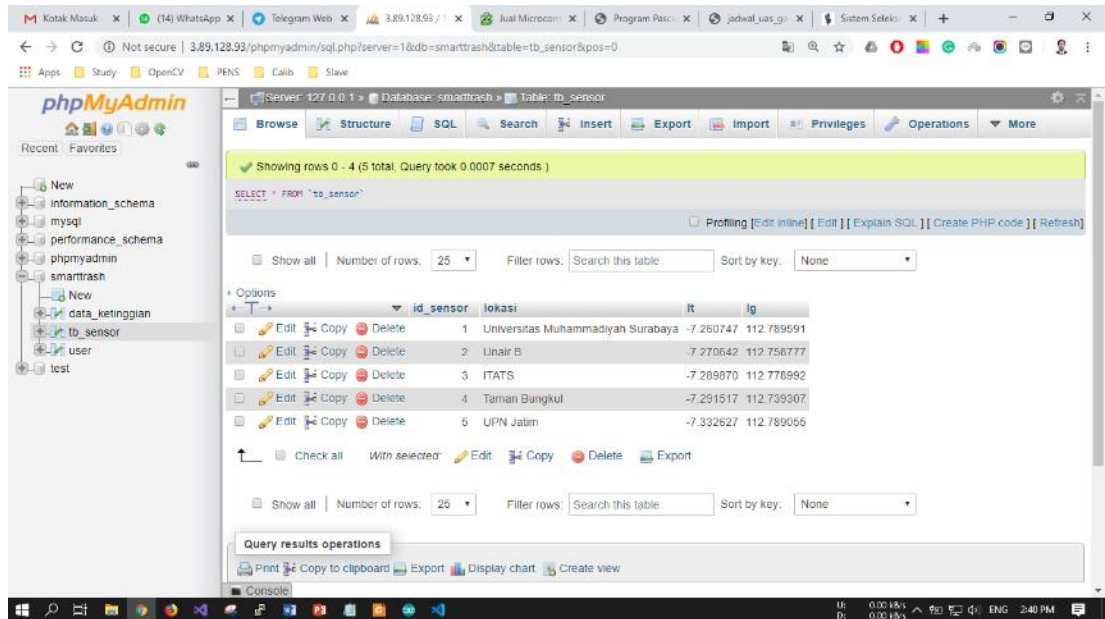
Database yang dipakai bernama smarttrash dan berisi 3 tabel yaitu data_ketinggian, tb_sensor, dan user seperti ditunjukkan pada gambar 13. Tabel data_ketinggian digunakan untuk menampung hasil pengukuran dari sensor yang didapatkan dari pengiriman node ke database. Tabel data_ketinggian memiliki beberapa kolom yaitu id_sensor, ketinggian dan log seperti pada gambar 14. Kemudian untuk tb_sensor ditunjukkan pada gambar 15 berisi nama lokasi alat berada dan koordinat lattitude longitude pada koordinat google maps. Kemudian untuk tabel user yang ditunjukkan gambar 16 berisi informasi tentang user yang dapat masuk ke website. Informasi tersebut terkait nama dan password yang dapat digunakan untuk menggunakan aplikasi.



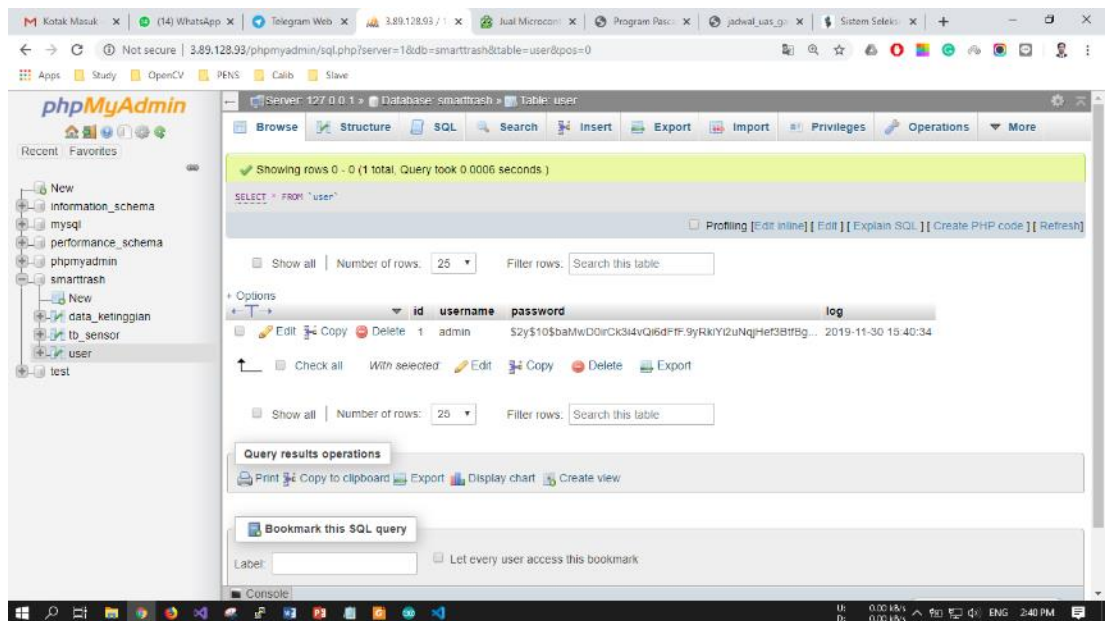
Gambar 13. Tabel dalam database
Sumber : Dokumentasi Penulis



Gambar 14. Tabel data_ketinggian
Sumber : Dokumentasi Penulis



Gambar 15. Tabel tb_sensor
Sumber : Dokumentasi Penulis



Gambar 16. Tabel user
Sumber : Dokumentasi Penulis

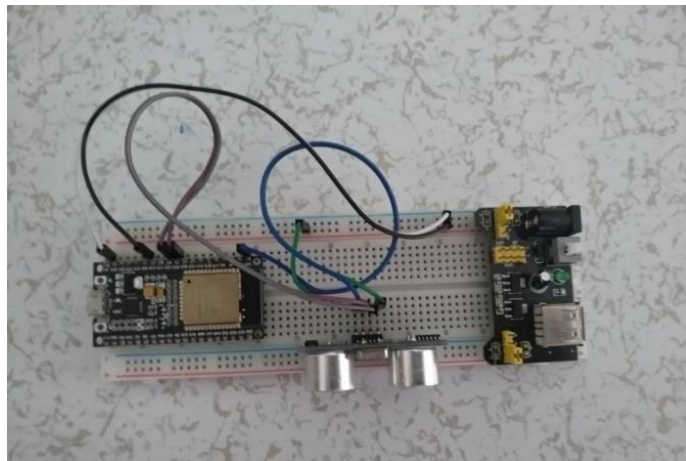
BAB IV

PENGUJIAN DAN ANALISA

Pada bab ini akan dibahas mengenai ujicoba hardware & software, Ujicoba pengiriman data sensor, ujicoba monitoring, ujicoba visualisasi web application dan analisa setiap pengujian.

4.1 Pengujian Hardware dan Sensor

Karena masih dalam bentuk prototipe pengujian sistem menggunakan project board. Rangkaian sistem ini ditunjukkan oleh gambar 17 dengan komponen esp32, sensor ultrasonik dan voltage regulator.



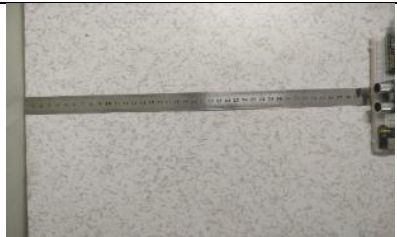



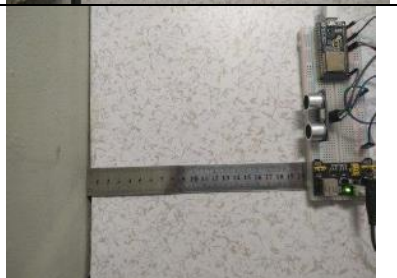
Gambar 17. Hardware Smarttrash
Sumber : Dokumentasi Penulis

Proses pengujian dilakukan dengan membandingkan antara pengukuran dengan alat ukur manual atau penggaris dengan hasil pembacaan pada sensor ultrasonic HY-SR04, hasil pengujian terdapat error atau deviasi dengan rata-rata 0,06%. Tabel 1 menunjukkan pengujian sensor ultrasonik HY-SR04 :

Tabel 1. Penjujian sensor ultrasonik

Pengukuran Ke :	D-Ruler (CM)	D-Sensor (CM)	Error (%)
1	40	38	0,050
2	35	32	0,086
3	30	28	0,067
4	25	23	0,080
5	20	19	0,050

Tabel 2. Pengukuran.


Pengujian 1 (40cm)	Levelling... 38.00 38.00 38.00 38.00 38.00 #Average (CM) = 38		
Pengujian 2 (35cm)	Levelling... 33.00 33.00 33.00 33.00 33.00 #Average (CM) = 33		
Pengujian 3 (30cm)	Levelling... 28.00 28.00 28.00 28.00 28.00 #Average (CM) = 28		
Pengujian 4 (25cm)	Levelling... 23.00 23.00 23.00 23.00 23.00 #Average (CM) = 23		
Pengujian 5 (20cm)	Levelling... 19.00 19.00 19.00 19.00 19.00 #Average (CM) = 19		

Pengukuran dilakukan sebanyak 5 kali dengan interval 2000ms, kemudian diambil nilai rata-rata (Avg). Tinggi bak sampah (H) di asumsikan 150 cm, dengan nilai deviasi (Dv) 10cm.

$$Level (\%) = 100 - \left(\frac{(Avg - Dv) \times 100}{H} \right)$$

4.2 Pengujian Pengiriman Data Sensor

Pengiriman data sensor dilakukan menggunakan koneksi internet wireless sehingga ketika tidak ada internet maka node tidak akan mengirim data ke database.



```
[SETUP] Tunggu 4...
[SETUP] Tunggu 3...
[SETUP] Tunggu 2...
[SETUP] Tunggu 1...
- Coba Koneksi WIFI SSID : didinx
..
- Koneksi Sukses
Levelling...
127.00
127.00
7.00
27.00
127.00
Average (CM) =
83
#Tinggi Sampah (CM) = 67
#Level (%) = 51
[HTTP] Memulai...
[HTTP] Melakukan GET ke server...
[HTTP] kode response GET: 200
{"status":"success insert"}
```

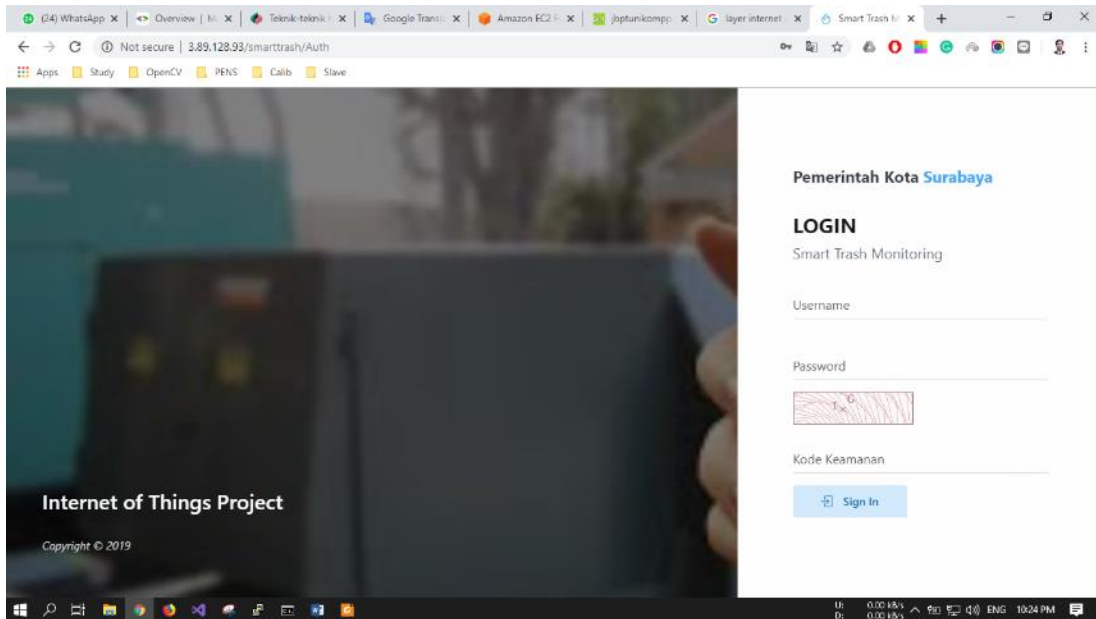
Gambar 18. Pengiriman Data

Sumber : Dokumentasi Penulis

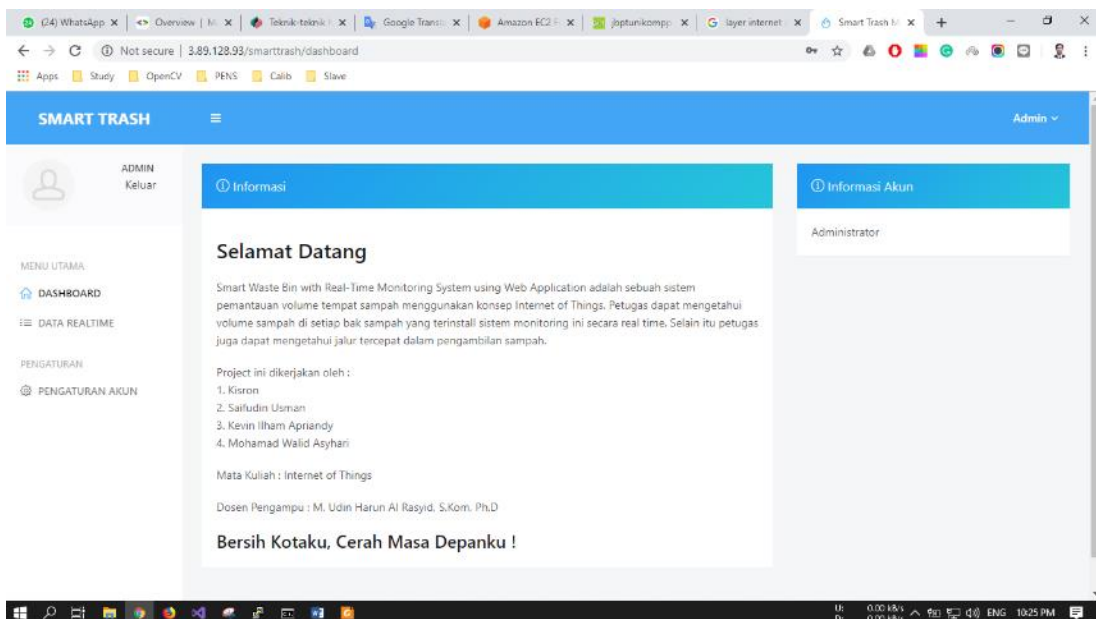
Gambar diatas menunjukkan proses pengiriman data hasil pengukuran sensor ke database. Setelah koneksi node terhubung ke internet, proses pengukuran level ketinggian sampah akan dilakukan sebanyak 5 kali, dan kemudian dilanjutkan proses pengiriman data dengan metode HTTP, dan Node akan menerima respon atau status sukses insert data jika data berhasil di kirimkan.

4.3 Pengujian Monitoring dan Visualiasi Aplikasi

Sistem ini menggunakan aplikasi web sebagai media untuk melakukan monitoring. Didalam web tersebut pengguna harus memiliki akun agar dapat masuk kedalam website. Halaman awal ditunjukkan pada gambar 19, untuk masuk ke aplikasi pengguna harus memasukkan username dan password sebagai keamanan. Untuk memastikan bahwa bukan robot yang memasuki website monitoring ini diberikan kode keamanan berupa captcha sehingga keamanan dari sistem ini akan terjaga.



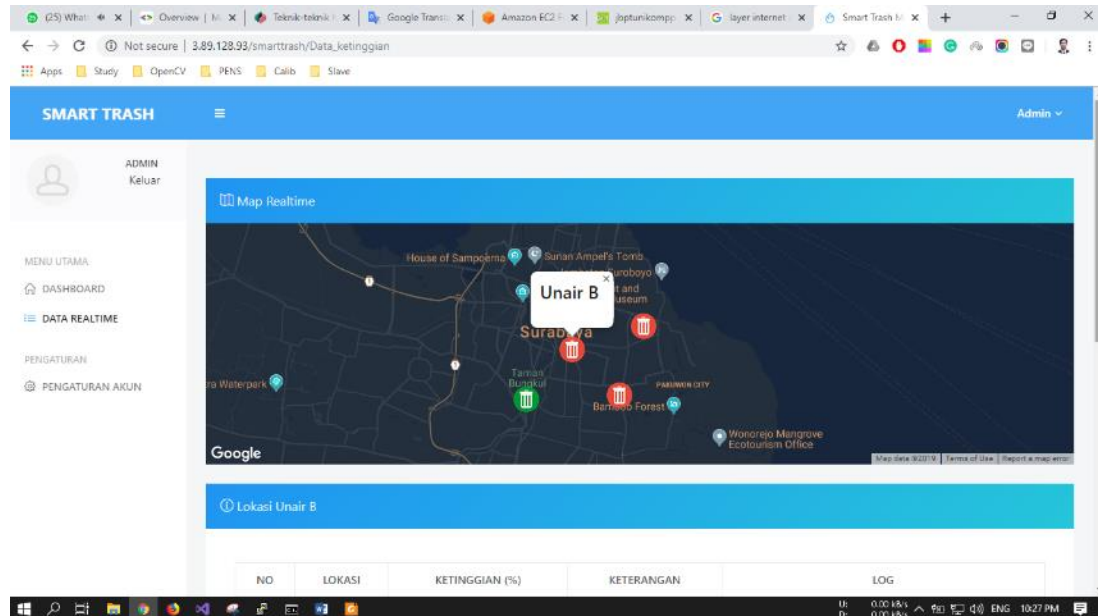
Gambar 19. Halaman Awal
Sumber : Dokumentasi Penulis



Gambar 20. Dashboard Aplikasi
Sumber : Dokumentasi Penulis

Setelah masuk ke dalam aplikasi pengguna akan mendapati halaman awal yang berisi informasi mengenai aplikasi ini dan bagaimana cara penggunaannya. Menu berikutnya yaitu data realtime, dalam menu tersebut pengguna dapat melihat data

tempat sampah dan juga volume secara update. Gambar 21 menunjukkan posisi tempat sampah dengan ikon merah untuk tempat sampah penuh dan hijau untuk tempat sampah yang belum penuh. Untuk mengetahui detail dari tempat sampah dengan cara klik pada ikon tempat sampah sehingga nanti akan keluar informasi berupa dimana tempat sampah itu berada, volume sampah dan statusnya seperti gambar 22.



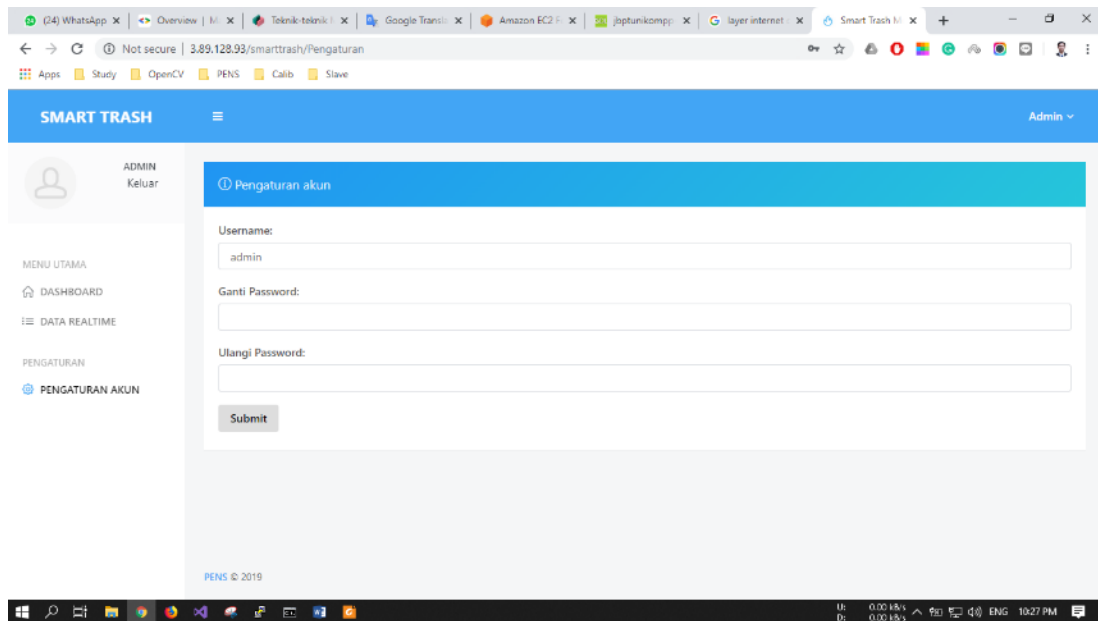
Gambar 21. Posisi Tempat Sampah di Maps

Sumber : Dokumentasi Penulis

NO	LOKASI	KETINGGIAN (%)	KETERANGAN	LOG
1	Unair B	86 %	Penuh	05 Desember 2019 pukul 13:32:09
2	Unair B	82 %	Penuh	05 Desember 2019 pukul 13:26:53
3	Unair B	63 %	Aman	05 Desember 2019 pukul 13:20:40
4	Unair B	63 %	Aman	05 Desember 2019 pukul 13:15:20
5	Unair B	63 %	Aman	05 Desember 2019 pukul 13:10:01
6	Unair B	63 %	Aman	05 Desember 2019 pukul 13:04:42
7	Unair B	63 %	Aman	05 Desember 2019 pukul 12:59:22
8	Unair B	63 %	Aman	05 Desember 2019 pukul 12:54:04
9	Unair B	27 %	Aman	05 Desember 2019 pukul 12:53:02
10	Unair B	28 %	Aman	05 Desember 2019 pukul 12:48:59

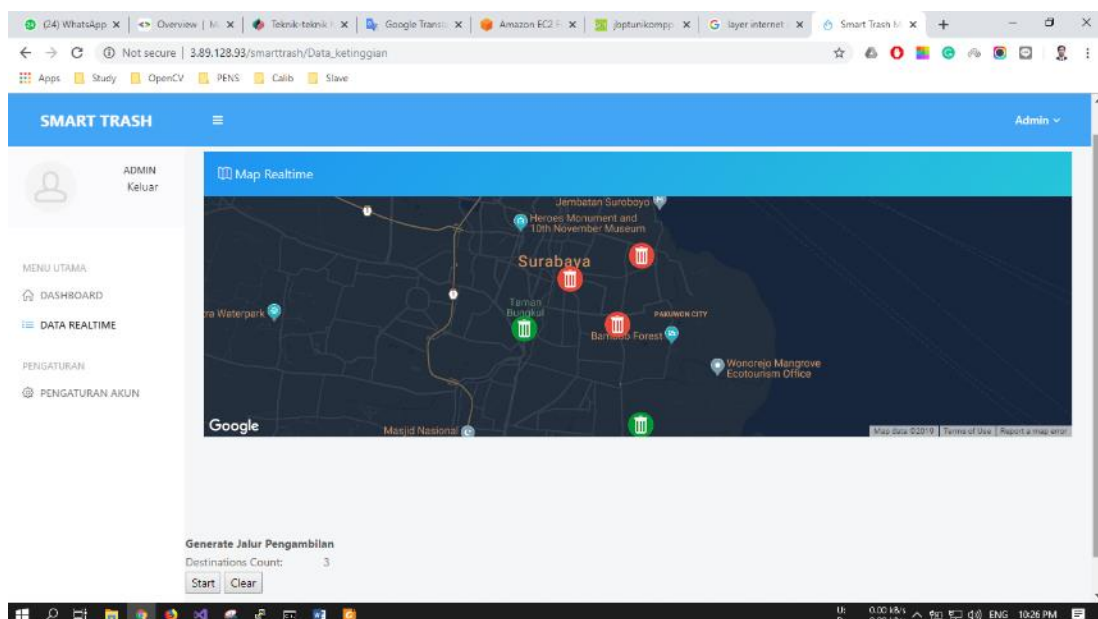
Gambar 22. Tabel Volume Sampah

Sumber : Dokumentasi Penulis



Gambar 23. Menu Pengaturan Akun

Sumber : Dokumentasi Penulis

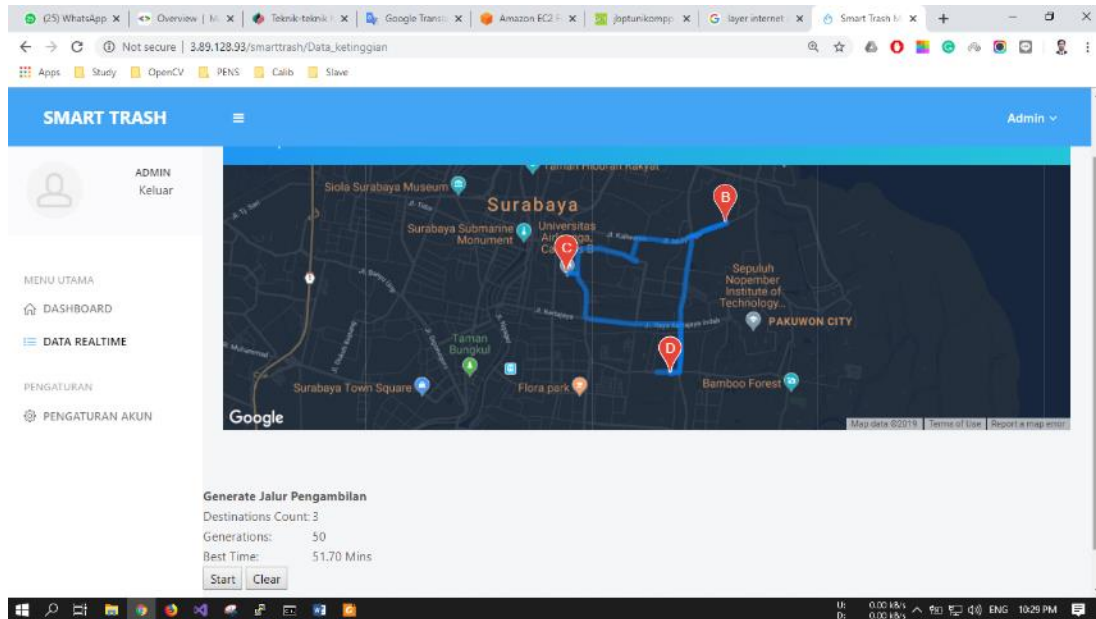


Gambar 24. Data Volume Sampah

Sumber : Dokumentasi Penulis

Pengguna juga dapat mengganti password pada menu pengaturan akun seperti pada gambar 23. Pengguna hanya diminta untuk memasukkan username dan password yang baru kemudian akan update otomatis ketika sudah disimpan.

Untuk membuat jalur tercepat dalam pengambilan sampah pengguna cukup menekan tombol Start dibawah maps. Dengan menggunakan genetika algoritma akan dibuat jalur sesuai jarak terdekat masing masing titik dan menandainya dengan titik B,C, dan D seperti gambar 25. Tempat sampah yang masuk dalam perhitungan jarak terdekat hanya tempat sampah yang sudah penuh saja.



Gambar 25. Pembuatan Jalur Tercepat

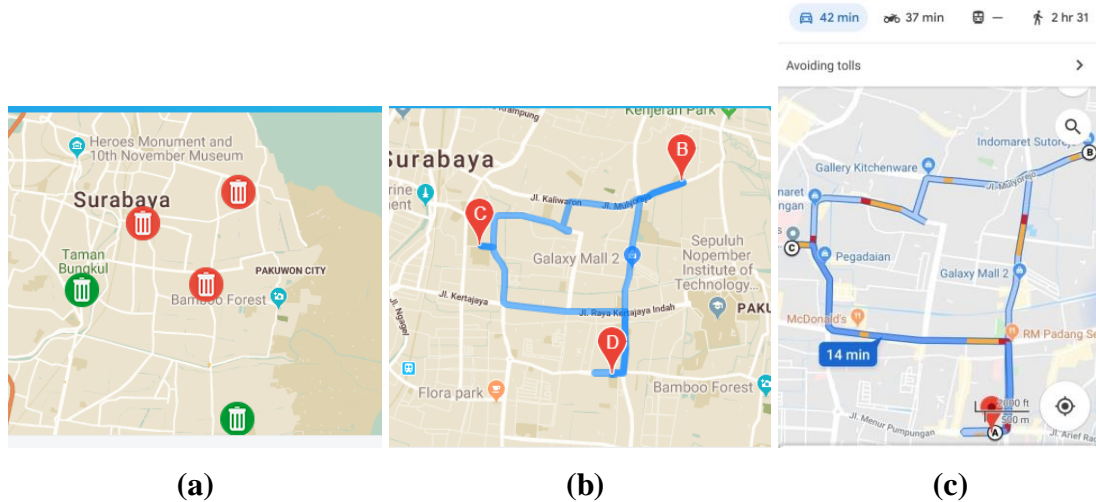
Sumber : Dokumentasi Penulis

4.4 Pengujian TSP

Pada pengujian TSP (Travel Salesman Problem) untuk membuat jalur tercepat sebagai bantuan dalam pengambilan sampah pada tempat sampah yang sudah penuh. Untuk melakukan pengujian dilakukan perbandingan antara TSP yang dibuat dengan menggunakan algoritma genetika dengan metode jalur tercepat dari google maps. Indikator yang dijadikan pembanding adalah waktu yang ditempuh.

Gambar 26 (a) menunjukkan ada 3 tempat sampah yang penuh. Kemudian dilakukan perencanaan jalur tercepat untuk mengambil sampah tersebut. Gambar 26 (b) menunjukkan hasil pembuatan jalur tercepat menggunakan algoritma genetika dengan perkiraan waktu 51.8 menit. Kemudian gambar 26 (c) adalah perhitungan waktu menggunakan aplikasi google maps sebagai pembanding.

Setelah dilakukan beberapa percobaan didapatkan hasil yang cukup baik dengan tingkat kesalahan kurang dari 10 menit. Tabel 3 menunjukkan hasil percobaan yang telah dilakukan.



Gambar 26. (a) Posisi Tempat Sampah (b) TSP GA (c) TSP Google Maps

Sumber : Dokumentasi Penulis

Tabel 3. Pengujian TSP

No	Route	TSP GA (menit)	Google Maps (menit)	Error (menit)
1	Unair B – ITATS – Unair B	37	30	7
2	Unair B – ITATS – UMS – Unair B	52	42	10
3	ITATS – UMS – Taman Bungkul – UPN - ITATS	85	82	3

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil pengujian dan analisa yang dilakukan maka dapat diambil kesimpulan sebagai berikut :

1. Dengan menggunakan sensor ultrasonik HC-SR04 sistem dapat mengetahui volume sampah dengan error pembacaan sebesar 0.06 %.
2. Sistem internet of things dengan metode HTTP dapat digunakan untuk memantau data volume sampah.
3. API dari Google Maps dan Algoritma Genetika dapat digunakan untuk membuat jalur tercepat berdasarkan volume sampah dan jarak antar tempat sampah dengan perbedaan waktu antara TSP GA dan TSP Google Maps paling besar 10 menit.

5.2 Saran

Dalam proyek ini terdapat beberapa kelebihan namun tidak terlepas dari kekurangan yang membutuhkan saran-saran untuk mendukung kesempurnaannya. Hingga saat ini saran-saran yang dapat diberikan adalah sebagai berikut :

1. Sistem yang sudah diuji diharapkan bisa direalisasikan dengan memperhatikan konsumsi power dari sistem.
2. Penggunaan metode pengiriman data selain HTTP POST untuk membandingkan performa dari pengiriman data yang sudah digunakan.
3. Penambahan fitur pada website agar informasi yang diberikan lebih menarik.

DAFTAR PUSTAKA

- [1] R. E. N. Sisyanto, Suhardi and N. B. Kurniawan, "Hydroponic smart farming using cyber physical social system with telegram messenger," 2017 International Conference on Information Technology Systems and Innovation (ICITSI), Bandung, 2017, pp. 239-245.
- [2] Norfadzlia Mohd Yusof, Mohd Faizal Zulkifli, Nor Yusma Amira Mohd Yusof, Azziana Afifie Azman.. "Smart Waste Bin with Real Time Monitoring System". (2018)
- [3] Zainal Hisham Che Soh, Muhammad S. Shafie, dkk.. "IoT Water Consumption Monitoring & Alert System". (2018)
- [4] Zhou Tao.. "TSP Problem solution based on improved Genetic Algorithm". Fourth International Conference on Natural Computation. (2008)
- [5] Ma Jingyan, Zhang Kehong.. "Research on TSP Solution Based on Genetic Algorithm of Logistic Equation". 2nd International Conference on Computer Science and Network Technology. (2012)
- [6] Elga Aris Prastyo, Arsitektur dan Fitur ESP32 (Modul EPS32) IoT, Edukasi Elektronika, available : <https://www.edukasielektronika.com/2019/07/arsitektur-dan-fitur-esp32-module-esp32.html> [online], diakses pada 13 Desember 2019
- [7] Kison, B. S. Bayu Dewantara and F. Ardilla, "Early Warning and IoT-based Reporting System for Mobile Trash Bin Robot Application," 2018 *International Electronics Symposium on Knowledge Creation and Intelligent Computing (IES-KCIC)*, Bali, Indonesia, 2018, pp. 341-348. doi: 10.1109/KCIC.2018.8628550
- [8] I. K. Gupta, A. Choubey and S. Choubey, "Randomized bias genetic algorithm to solve traveling salesman problem," 2017 *8th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, Delhi, 2017, pp. 1-6. doi: 10.1109/ICCCNT.2017.8204127

DATA KELOMPOK

Data Anggota 1:

Nama Mahasiswa : Kisron
NRP : 2020191006
Jurusan / Program Studi : Magister Terapan Teknik Informatika dan Komputer

Data Anggota 2:

Nama Mahasiswa : Saifudin Usman
NRP : 2020191016
Jurusan / Program Studi : Magister Terapan Teknik Informatika dan Komputer

Data Anggota 3:

Nama Mahasiswa : Kevin Ilham Apriyandi
NRP : 2020191018
Jurusan / Program Studi : Magister Terapan Teknik Informatika dan Komputer

Data Anggota 4:

Nama Mahasiswa : Mohamad Walid Asyhari
NRP : 1020191021
Jurusan / Program Studi : Magister Terapan Teknik Informatika dan Komputer

LAMPIRAN

Source code lengkap dapat diakses melalui github berikut ini :

<https://github.com/kisron86/smarttrash>

URL website bisa diakses yaitu :

<http://3.89.128.93/smarttrash>

username : admin

password : admin