

BAB II TINJAUAN PUSTAKA

A. Penelitian Terkait

Kebersihan lingkungan merupakan suatu kewajiban bagi setiap individu untuk menciptakan lingkungan yang bersih, indah, dan menyenangkan secara visual. Untuk mencapai hal tersebut, diperlukan adanya tempat pembuangan akhir yang berfungsi sebagai tempat penyimpanan sementara berbagai jenis sampah sebelum diproses untuk didaur ulang atau diolah menjadi produk lainnya. Menjaga kebersihan lingkungan adalah tanggung jawab bersama masyarakat, karena selain mengganggu pemandangan, tumpukan sampah yang tidak diangkat oleh petugas sampah juga akan menyebabkan bau yang tidak sedap. Akumulasi sampah yang berlebihan dapat menimbulkan bau yang tidak nyaman dan menjadi sumber penyakit [5].

Oleh karena itu, diperlukan penggunaan teknologi untuk mengatasi permasalahan lingkungan ini dan menciptakan lingkungan yang bersih dan sehat. Teknologi tersebut akan mempermudah petugas sampah dalam melaksanakan tugasnya dengan tepat waktu, terutama ketika tempat sampah sudah penuh. Dalam penelitian ini, dirancang sebuah sistem informasi berbasis website yang menggunakan sistem kendali NodeMCU. NodeMCU merupakan papan elektronik yang menggunakan chip ESP8266 dengan fungsi mikrokontroler dan koneksi internet (WiFi). Papan ini memiliki beberapa pin input/output yang dapat dikembangkan menjadi aplikasi monitoring dan kontrol pada proyek *Internet of Things* (IoT) [6]. Berdasarkan permasalahan yang telah diuraikan, dilakukan penelitian dengan judul "Sistem Monitoring Tempat Sampah Pintar Secara Real Time Menggunakan Metode *Fuzzy Logic*". Penelitian ini menggunakan studi literatur terkait dengan penelitian sebelumnya sebagai acuan dalam perancangan sistem yang akan diteliti.

Penelitian [4], [7] membuat tempat sampah pintar yang dapat mengendalikan dan mengirimkan data ketika tempat sampah mencapai kapasitas penuh. Inovasi ini memudahkan petugas kebersihan dalam mengelola pembuangan sampah secara efisien tanpa memakan waktu berlebihan. Pertukaran data difasilitasi dengan menggunakan protokol MQTT, yang khusus digunakan untuk implementasi Internet of Things (IoT), serta sensor pengukur jarak, yakni sensor ultrasonik HY-SRF05, Arduino Uno, dan modul WiFi ESP8266. Penelitian ini bertujuan untuk menghasilkan nilai delay, packet loss, dan throughput selama pengiriman data, memberikan wawasan berharga terhadap kinerja tempat sampah pintar yang telah dikembangkan.

Penelitian [4] memberikan kontribusi bagi kampus dalam menangani permasalahan sampah secara lebih efektif dan efisien di Era Industri 4.0 dengan mengembangkan aplikasi Sistem Pemantauan Tempat Sampah Cerdas berbasis

Internet of Things. Untuk mendukung fungsionalitas aplikasi tersebut, diperlukan penggunaan NodeMcu dan Arduino Uno sebagai mikrokontroler, sensor ultrasonik (HCSR-04), serta motor servo sebagai penggerak penutup tempat sampah. Selain itu, sensor ping juga digunakan dalam rangka memperoleh data ketinggian sampah yang kemudian disampaikan kepada petugas sampah melalui sebuah aplikasi. Metode pengembangan aplikasi ini mengadopsi model waterfall dan diilustrasikan dalam Unified Modeling Language (UML) agar dapat menjadi penanda apakah tempat sampah sudah mencapai kapasitas penuh atau belum.

Penelitian [3] mengembangkan digitalisasi sistem pemantauan sampah rumahan berbasis Internet of Things (IoT) dengan tujuan mencegah kelebihan kapasitas tempat sampah dan mengubah sampah menjadi sumber ekonomi tambahan bagi masyarakat. Desain sistem ini melibatkan penggunaan modul mikrokontroler Arduino Mega, sensor infrared, sensor induktif proximity, modul HX711, sensor load cell, dan NodeMCU untuk mengirimkan data melalui local server. Berdasarkan hasil pengujian, sensor load cell untuk sampah logam menunjukkan kesalahan rata-rata sebesar 2,89% dengan selisih berat rata-rata 0,83 gram dibandingkan dengan timbangan konvensional. Sementara itu, sensor load cell untuk sampah non logam memiliki kesalahan rata-rata sebesar 2,47% dengan selisih berat rata-rata 0,9 gram dibandingkan dengan timbangan konvensional. Pengujian pada perangkat lunak dan perangkat keras membuktikan bahwa keduanya dapat beroperasi dengan baik dan sesuai dengan perencanaan, memungkinkan data pengujian dari perangkat keras untuk didokumentasikan secara digital di dalam perangkat lunak.

Penelitian yang dilakukan oleh [8] membahas permasalahan dalam sistem pendataan sampah di TPA Kota Tangerang. Latar belakang penelitian menyoroti perubahan struktur organisasi dari Dinas Kebersihan dan Pertamanan Kota Tangerang menjadi Dinas Lingkungan Hidup Kota Tangerang pada tahun 2017. Pada tahun 2015, meskipun telah menggunakan timbangan, sistem pendataan hanya mencakup berat sampah, sedangkan volume dan ritasinya masih dicatat secara manual. Metode penelitian melibatkan wawancara dengan pimpinan dan pegawai TPA Kota Tangerang yang terlibat dalam pendataan manual menggunakan ms. excel. Pengembangan program dilakukan dengan menerapkan metode SDLC. Hasil penelitian menunjukkan bahwa sistem manual menyulitkan pegawai dalam memasukkan data volume dan ritase harian, karena bergantung pada pendataan manual dengan kertas kerja. Selain itu, pegawai Dinas Lingkungan Hidup Kota Tangerang harus secara manual mentransfer data volume dan ritase sampah dari komputer timbangan ke media penyimpanan seperti flashdisk.

Penelitian oleh [2] membahas pengelolaan Tempat Penampungan Sampah Sementara (TPS) di lima wilayah DKI Jakarta, khususnya TPS 3R yang memiliki kompleksitas tinggi. TPS 3R belum optimal karena fasilitas yang kurang memadai.

Penelitian ini membangun sistem pendukung keputusan untuk menentukan prioritas bantuan dana bagi TPS 3R, dengan menggunakan metode optimasi Fuzzy Logic dan Hill Climbing. Hasilnya diharapkan membantu DLH DKI Jakarta dalam menentukan TPS 3R yang berhak mendapatkan bantuan dana dengan cepat dan tepat, meningkatkan efektivitas pengelolaan sampah di wilayah tersebut.

Penelitian [9] mengembangkan suatu sistem berbasis Internet of Things (IoT) yang dapat memantau lokasi dan ketinggian permukaan sampah di Tempat Penampungan Sampah (TPS) melalui sebuah website. NodeMCU ESP32 berfungsi sebagai pengatur utama untuk seluruh komponen perangkat keras dan perangkat lunak. Sensor ultrasonik digunakan untuk mengukur ketinggian sampah, sedangkan Global Positioning System (GPS) digunakan untuk menentukan titik lokasi TPS. Penelitian ini menggunakan website sebagai platform pemantauan lokasi dan ketinggian permukaan sampah di TPS. Sistem yang dikembangkan dapat menampilkan hasil pengukuran ketinggian sampah dan titik lokasi pada antarmuka website. Hasil pengujian menunjukkan bahwa pada TPS 1, sensor 1 memiliki error 0%, sensor 2 memiliki error 0,067%, dan sensor 3 memiliki error 0,075%. Pada TPS 2, sensor 1 memiliki error 0,053%, sensor 2 memiliki error 0,081%, dan sensor 3 memiliki error 0,017%. Selain itu, nilai rata-rata error GPS TPS terhadap Google Maps pada TPS 1 adalah 64,565 m, sedangkan pada TPS 2 adalah 68,795 m.

Tabel 1. Penelitian Terkait

No	Nama (Tahun)	Judul
(1)	(2)	(3)
1	(Zhafira et al., 2019)	Analisis dan Rancang Bangun Sistem Monitoring Tempat Sampah Berbasis IOT menggunakan Protokol MQTT
2	(Safitri & Zulfian, 2021)	Aplikasi smart trash bin monitoring system berbasis <i>Internet of Things</i> (IoT)
3	(Nur Pasha et al., 2022)	Digitalisasi sistem monitoring sampah rumahan berbasis <i>Internet of Things</i>
4	(Romlah et al., 2020)	Monitoring Volume & Ritase Pengangkutan Sampah Pada Tempat Pembuangan Akhir (TPA) Kota Tangerang
5	(Arbi et al., 2021)	Pemanfaatan <i>Fuzzy Logic</i> dan <i>Hill Climbing</i> untuk Optimasi Pengelolaan Tempat Pengolahan Sampah Reduce-Reuse-Recycle

	(TPS 3R) pada Dinas Lingkungan Hidup (DLH) DKI Jakarta
6	(Visa Bilin Nur, Dedi Triyanto, 2021) Sistem pemantauan tempat penampungan sampah secara realtime dengan memanfaatkan location tracking menggunakan antarmuka website

F. Landasan Teori

1. Profil Desa Kuripan

Desa Kuripan merupakan salah satu desa yang terletak di kecamatan kesugihan Kabupaten Cilacap. Desa Kuripan Kabupaten Cilacap memiliki luas wilayah 8.230,62 Ha atau 82,31 Km² dengan batas wilayah:

- i. Utara: Kabupaten Banyumas Timur: Kecamatan Maos, dan Kecamatan Adipala Kabupaten Cilacap
- ii. Selatan: Samudera Hindia
- iii. Barat: Kecamatan Cilacap Utara, Kabupaten Cilacap

Peta Desa Kuripan, Kecamatan Kesugihan, Kabupaten Cilacap ditunjukkan pada Gambar 1 berikut.



Gambar 1. Peta Desa Kuripan Kecamatan Kesugihan [10]

Identitas Desa Kuripan, Kecamatan Kesugihan, Kabupaten Cilacap yaitu:

- i. Nama Desa : Kuripan
- ii. Kode Desa : 3301022006
- iii. Kecamatan : Kesugihan
- iv. Kode Kecamatan : 330102
- v. Kabupaten : Cilacap
- vi. Kode Kabupaten : 3301
- vii. Provinsi : Jawa Tengah
- viii. Kode Provinsi : 33
- ix. Kode Pos : 53274

2. Sampah

Sampah merupakan residu atau sisa dari produk atau barang yang tidak lagi digunakan. Sampah merupakan material yang tidak diinginkan dalam suatu proses. Terdapat dua klasifikasi sampah berdasarkan sifatnya, yaitu sampah organik dan sampah anorganik. Selain itu, sampah juga dapat diklasifikasikan berdasarkan bentuknya, seperti sampah padat, cair, alam, konsumsi, manusia, dan sampah radioaktif [11].

Dalam konteks undang-undang, sampah didefinisikan sebagai sisa dari aktivitas sehari-hari manusia dan/atau proses alam yang berbentuk padat. Sampah memiliki dampak negatif, seperti mencemari sungai dan menghambat pergerakan air tanah. Selain itu, sampah juga dapat mencemari tanah dan mengancam kesehatan manusia. Akumulasi sampah yang menumpuk juga dapat menghasilkan bau yang mengganggu aktivitas dan kenyamanan manusia.

Sampah sering kali menjadi masalah lingkungan karena penumpukannya yang berlebihan dapat mencemari lingkungan, menghasilkan bau yang tidak sedap, dan mengancam kesehatan manusia serta kehidupan makhluk lain. Selain itu, pengolahan dan pembuangan sampah yang tidak tepat dapat menyebabkan dampak negatif terhadap ekosistem, polusi tanah, air, dan udara.

Pengelolaan sampah yang baik dan bertanggung jawab menjadi sangat penting untuk menjaga kebersihan lingkungan, mencegah penyebaran penyakit, dan mengurangi dampak negatif terhadap ekosistem. Upaya pengelolaan sampah meliputi tindakan seperti pengurangan, daur ulang, pengomposan, pemilahan, pengangkutan, dan pengolahan sampah secara aman dan efisien.

Dalam konteks yang lebih luas, masalah sampah juga menimbulkan kebutuhan akan kesadaran masyarakat untuk mengurangi produksi sampah, melakukan daur ulang, dan memilih penggunaan bahan yang ramah lingkungan. Tujuan akhir adalah menciptakan lingkungan yang bersih, sehat, dan berkelanjutan untuk kehidupan manusia dan ekosistem di sekitarnya.

Untuk mengatasi dampak dari sampah, beberapa upaya yang dapat dilakukan antara lain adalah pemisahan sampah berdasarkan jenisnya, pengolahan sampah organik dan anorganik, serta pengangkutan sampah secara tepat waktu oleh petugas sampah yang bertanggung jawab.

3. *Internet of Things* (IoT)

Internet of Things (IoT) adalah gagasan di mana berbagai objek sehari-hari, seperti perangkat elektronik, kendaraan, dan peralatan rumah tangga, dilengkapi dengan sensor, perangkat lunak, dan jaringan komunikasi agar dapat terhubung dan berkomunikasi melalui internet. Pada dasarnya, IoT memungkinkan objek-objek tersebut mengumpulkan data dan berinteraksi dengan lingkungan sekitarnya. Mereka juga dapat saling berkomunikasi dan

bertindak secara otomatis berdasarkan data yang dikumpulkan. Dalam lingkup bisnis dan industri, konsep IoT dapat digunakan untuk meningkatkan efisiensi produksi dan mengoptimalkan kinerja peralatan. Di sisi konsumen, IoT dapat meningkatkan kenyamanan dan keamanan di rumah tangga [6].

Cara kerja IoT melibatkan interaksi otomatis antara perangkat atau mesin yang terhubung tanpa campur tangan pengguna dan dapat berjalan dalam jarak berapa pun. Untuk mencapai cara kerja tersebut, internet berperan sebagai penghubung antara kedua perangkat yang saling berinteraksi, sedangkan pengguna bertugas sebagai pengatur dan pengawas bagi perangkat tersebut. Manfaat yang diperoleh dari konsep IoT adalah memungkinkan pekerjaan menjadi lebih cepat, mudah, dan efisien [12]. Sistem dasar dari IoT melibatkan lima hal, yaitu:

- i. *Hardware/fisik (Things)*: Merujuk pada benda-benda yang terhubung ke internet dan mampu berkomunikasi dengan perangkat lain.
- ii. *Koneksi Internet/Gateway*: Merupakan sarana utama yang memungkinkan benda-benda tersebut berkomunikasi antara satu sama lain melalui jaringan internet.
- iii. *Cloud Data Center*: Tempat penyimpanan data atau menjalankan aplikasi yang terhubung dengan perangkat IoT.
- iv. *Analytic*: Tempat analisis dari sebuah data yang dihasilkan oleh *device*.
- v. *User Interface*: Antarmuka dan interaksi dari sebuah sistem IoT.

Secara ringkas, Internet of Things adalah konsep di mana benda-benda di sekitar kita dapat berkomunikasi satu sama lain melalui jaringan seperti internet. Implementasi konsep IoT memerlukan komponen-komponen seperti sensor untuk mengumpulkan data dari lingkungan sekitar, aktuator untuk bertindak berdasarkan data tersebut, jaringan komunikasi untuk menghubungkan objek-objek tersebut ke internet, dan platform pengolahan data untuk menganalisis dan memproses data yang telah dikumpulkan.

Selain itu, IoT juga membutuhkan protokol komunikasi dan keamanan yang kuat untuk melindungi data yang dikirim dan diterima melalui internet. Integrasi IoT dengan teknologi lain seperti kecerdasan buatan (AI) dan analisis data (data analytics) juga dapat meningkatkan fungsionalitas dan efektivitas dari sistem IoT

Tujuan utama dari IoT adalah untuk menciptakan sistem yang lebih cerdas dan terhubung di mana objek-objek dapat berkomunikasi dan berinteraksi satu sama lain secara otonom. Melalui konektivitas internet, data yang dikumpulkan oleh perangkat IoT dapat diakses dan dianalisis untuk memberikan informasi yang berguna, mengoptimalkan kinerja, dan memungkinkan pengambilan keputusan yang lebih baik

Contoh penerapan IoT dapat ditemukan dalam berbagai bidang, seperti smart home (rumah pintar), smart city (kota pintar), smart agriculture (pertanian pintar), smart healthcare (pelayanan kesehatan pintar), dan masih banyak lagi.

Dalam konteks IoT, perangkat-perangkat terhubung dapat saling berkomunikasi, mengumpulkan data, dan mengambil tindakan secara otomatis untuk meningkatkan efisiensi, kenyamanan, keamanan, dan produktivitas. Arsitektur IoT ditunjukkan pada Gambar 2.



Gambar 2. Arsitektur IoT [4]

4. Sistem Monitoring

Sistem monitoring adalah sebuah sistem yang dirancang untuk mengawasi, memantau, dan mengumpulkan data atau informasi secara berkala tentang suatu proses, perangkat, lingkungan, atau aktivitas tertentu. Tujuan utama dari sistem monitoring adalah untuk memastikan bahwa segala sesuatu berjalan dengan baik dan sesuai dengan standar yang ditetapkan, serta untuk memberikan informasi yang diperlukan untuk mengambil tindakan yang tepat jika terjadi masalah atau perubahan kondisi yang signifikan [13].

Sistem monitoring Internet of Things (IoT) adalah jenis sistem monitoring yang memanfaatkan teknologi IoT untuk mengumpulkan, mentransmisikan, dan menganalisis data dari berbagai perangkat yang terhubung ke internet. IoT adalah jaringan perangkat fisik yang terhubung secara online dan dilengkapi dengan sensor, perangkat keras, perangkat lunak, dan kemampuan komunikasi yang memungkinkan mereka berinteraksi dan berbagi data secara otomatis. Sistem ini berfungsi untuk mengumpulkan data secara real-time dan melakukan monitoring jarak jauh pada aset atau proses. Data yang dikumpulkan bisa dianalisis untuk menentukan tindakan apa yang cocok untuk dilakukan nantinya.

5. Android Studio

Android Studio adalah lingkungan pengembangan terintegrasi (IDE) yang dikembangkan oleh Google untuk memudahkan para pengembang dalam membuat aplikasi berbasis Android. IDE ini menyediakan berbagai fitur dan alat yang dibutuhkan untuk mengembangkan, menguji, dan menerapkan aplikasi Android. Android menyediakan platform untuk Developer Junior maupun senior mengembangkan aplikasi yang sendiri atau kelompok. Android

terdapat fitur fitur canggih sesuai dengan developer [3]. Fitur utama dari Android Studio meliputi:

- i. Editor Kode
Android Studio menyediakan editor kode yang kuat dengan fitur seperti penyorotan sintaks, penyelesaian kode otomatis, pemformatan kode, dan kemampuan navigasi yang baik.
- ii. Designer Antarmuka Pengguna (UI)
IDE ini memiliki alat yang memungkinkan pengembang untuk dengan mudah merancang antarmuka pengguna aplikasi Android dengan menggunakan tata letak visual, serta mengedit XML tata letak.
- iii. Pemeriksaan Kode
Android Studio memeriksa kode Anda untuk menemukan kesalahan potensial, memberikan saran perbaikan, dan memastikan kualitas kode yang lebih baik.
- iv. Emulator
IDE ini menyediakan emulator Android bawaan yang memungkinkan Anda menguji aplikasi Anda pada berbagai perangkat dan konfigurasi Android yang berbeda.
- v. Debugger
Android Studio memiliki alat pemecah masalah (debugger) yang membantu Anda dalam menganalisis dan memperbaiki masalah pada aplikasi.
- vi. Integrasi dengan Android SDK
IDE ini terintegrasi dengan Android Software Development Kit (SDK), yang mencakup berbagai pustaka, alat, dan sumber daya yang diperlukan untuk mengembangkan aplikasi Android.
- vii. Dukungan untuk Bahasa Pemrograman
Android Studio mendukung berbagai bahasa pemrograman, termasuk Java dan Kotlin. Kotlin kini telah menjadi bahasa pemrograman resmi yang disarankan oleh Google untuk pengembangan aplikasi Android.
- viii. Dukungan untuk Gradle
Android Studio menggunakan sistem build Gradle yang kuat untuk mengelola proyek dan dependensi aplikasi.
- ix. Dukungan untuk Pengembangan Aplikasi Berbasis Android
Android Studio mendukung pengembangan berbagai jenis aplikasi Android, termasuk aplikasi seluler, aplikasi tablet, aplikasi TV, dan bahkan aplikasi wearable.
- x. Pustaka dan Template
Android Studio menyediakan pustaka dan template yang dapat membantu Anda mempercepat pengembangan aplikasi dengan menyediakan kode dan komponen yang umum digunakan.

Pada penelitian ini memiliki Proses dan mekanisme pembuatan aplikasi pada android studio versi 2022,2.1.20 sebagai berikut:

- i. Analisis Kebutuhan:
 - a) Identifikasi kebutuhan pengguna dan tujuan aplikasi.
 - b) Analisis fitur dan fungsionalitas yang diperlukan dalam aplikasi monitoring daya listrik.
 - c) Penentuan integrasi dengan *Firestore* untuk penyimpanan dan sinkronisasi data.
- ii. Desain Sistem:
 - a) Perancangan arsitektur sistem aplikasi, termasuk antarmuka pengguna dan fitur-fitur utama.
 - b) Desain basis data untuk menyimpan data volume sampah dan waktu pengambilan terakhir.
 - c) Penentuan skema penggunaan algoritma *Fuzzy Logic* Mamdani untuk pengambilan keputusan berdasarkan data yang diterima.
- iii. Pengembangan Aplikasi Android:
 - a) Pembuatan antarmuka pengguna menggunakan Android Studio.
 - b) Implementasi logika aplikasi untuk mengambil data volume sampah dan waktu pengambilan terakhir dari perangkat sensor.
 - c) Integrasi dengan *Firestore* untuk penyimpanan data dan sinkronisasi dengan aplikasi.
- iv. Implementasi Algoritma *Fuzzy Logic* Mamdani
 - a) Penentuan variabel masukan dan keluaran yang relevan untuk sistem monitoring tempat sampah.
 - b) Pembuatan aturan-aturan *Fuzzy* berdasarkan kondisi volume sampah dan pengambilan terakhir.
 - c) Implementasi metode *defuzzyfikasi* untuk menghasilkan output yang dapat dimengerti.
- v. Pengujian dan Evaluasi:
 - a) Pengujian aplikasi untuk memastikan keakuratan pengambilan data volume sampah dan kinerja algoritma *Fuzzy*.
 - b) Evaluasi respons dan pengalaman pengguna terhadap antarmuka dan fungsionalitas aplikasi.
 - c) Pembaruan dan perbaikan berdasarkan umpan balik dan hasil pengujian.

6. *Firestore*

Firestore adalah platform pengembangan aplikasi yang menawarkan berbagai layanan yang dapat membantu para pengembang dalam membangun aplikasi yang lebih baik, cepat, dan mudah diakses. *Firestore* menyediakan berbagai alat dan layanan yang mencakup berbagai aspek pengembangan aplikasi, mulai dari penyimpanan data, otentikasi pengguna, analitik, hingga hosting. Menggunakan *Firestore* menjadi pilihan banyak pengembang, karena platform yang dirancang oleh Google ini memiliki fitur yang cukup handal untuk dimanfaatkan dalam berbagai konteks. Basis data *Firestore* secara

otomatis akan menyelaraskan informasi dengan aplikasi klien yang terhubung, tanpa perlu campur tangan secara manual. Aplikasi yang mendukung berbagai platform seperti Android, iOS, dan JavaScript dapat mengakses pembaruan data terbaru secara otomatis ketika terhubung ke server *Firebase*. Tujuan utama dari penggunaan database real-time di *Firebase* adalah untuk meningkatkan kinerja waktu akses data. Database real-time ini dioptimalkan untuk meminimalkan waktu akses data hingga dalam rentang waktu yang sangat cepat, bahkan hingga hitungan mikrodetik atau nanodetik, sehingga biaya akses data dapat diminimalkan [14]. Beberapa fitur utama dari *Firebase* meliputi:

i. *Cloud Firestore*

Layanan penyimpanan database yang bersifat real-time dan fleksibel. *Firebase Firestore* memungkinkan Anda menyimpan dan mengambil data dalam waktu nyata, baik untuk aplikasi seluler, *web*, atau desktop.

ii. *Authentication*

Firebase menyediakan sistem otentikasi pengguna yang aman dan mudah diintegrasikan dalam aplikasi Anda. Ini termasuk pilihan autentikasi dengan email, media sosial, serta integrasi dengan layanan lain.

iii. *Cloud Functions*

Layanan ini memungkinkan Anda menulis dan menjalankan kode di server *Firebase* saat terjadi peristiwa tertentu, seperti penyimpanan data atau otentikasi.

iv. *Cloud Storage*

Firebase menyediakan penyimpanan cloud yang dapat diakses dengan mudah untuk menyimpan berbagai jenis data, seperti gambar, video, dan file lainnya.

v. *Real-time Database*

Firebase menyediakan basis data real-time yang dapat disinkronkan dengan cepat antara klien dan server. Ini sangat berguna untuk aplikasi yang memerlukan pembaruan data instan.

vi. *Hosting*

Firebase memungkinkan Anda untuk meng-host aplikasi *web* dan menyediakan domain khusus secara gratis.

vii. *Cloud Messaging*

Layanan ini memungkinkan Anda mengirim pesan push ke pengguna perangkat Android, iOS, dan *web*.

viii. *Remote Config*

Firebase memungkinkan Anda mengubah parameter dalam aplikasi tanpa perlu merilis pembaruan. Ini membantu dalam mengelola variasi fitur atau tampilan tanpa perlu mengubah kode.

ix. *Crashlytics*

Firebase menyediakan alat untuk melacak dan menganalisis laporan kecelakaan dalam aplikasi, yang membantu Anda memahami masalah yang dialami pengguna.

x. Analitik

Firebase Analytics membantu Anda melacak dan menganalisis perilaku pengguna dalam aplikasi Anda, memberikan wawasan tentang kinerja dan interaksi pengguna.

7. *Visual Studio Code*

Visual Studio Code adalah editor kode yang dapat diaplikasikan pada semua jenis komputer meja. Perangkat lunak ini juga berlaku untuk segala jenis sistem operasi, termasuk Windows, Linux, dan Mac. Editor kode ini dibentuk serta dikembangkan oleh Microsoft. Dari sisi fungsinya, text editor ini memiliki kemampuan yang mengesankan namun tetap memiliki kinerja yang ringan saat digunakan. Software ini juga memegang peranan signifikan dalam menghasilkan dan mengedit kode sumber dari berbagai bahasa pemrograman, termasuk contohnya penggunaan *Python* untuk penerapan logika *Fuzzy* [15].

Peneliti menggunakan *Visual Studio Code* untuk melaksanakan implementasi proses logika *Fuzzy* dengan bahasa pemrograman *Python* yang telah diinstal di dalam *Visual Studio Code* (VSC). Peneliti juga menggunakan data yang telah diolah sebelumnya di *Firebase*, lalu mengirimkan data tersebut ke antarmuka Android dalam bentuk tabel dan grafik. *Visual Studio Code* memiliki kemampuan untuk mengelola proyek, menulis kode dalam bahasa *Python*, melakukan tahap debugging, serta mengintegrasikan dengan platform *Firebase* dan Android. Semua ini digunakan untuk mengembangkan sistem pemantauan volume sampah dan waktu pengambilan terakhir.

8. NodeMCU ESP8266

NodeMCU adalah sebuah papan elektronik yang menggunakan chip ESP8266 sebagai dasarnya. Papan ini memiliki kemampuan untuk berfungsi sebagai mikrokontroler dan memiliki koneksi internet melalui Wi-Fi. NodeMCU dilengkapi dengan beberapa pin Input/Output (I/O) yang memungkinkan pengembangan aplikasi monitoring dan pengendalian pada proyek IoT. Untuk memprogram NodeMCU ESP8266, Anda dapat menggunakan compiler Arduino dengan menggunakan Arduino IDE. Bentuk fisik NodeMCU ESP8266 memiliki port USB (mini USB) yang memudahkan dalam proses pemrograman. Papan NodeMCU didesain untuk mudah digunakan dan diprogram, sehingga telah mendapatkan popularitas yang luas di kalangan pembuat dan komunitas elektronika [16]. Berikut adalah beberapa fitur kunci dari papan NodeMCU ESP8266:

i. Modul Wi-Fi ESP8266

ESP8266 adalah chip Wi-Fi dengan biaya rendah dan dilengkapi dengan stack TCP/IP, sehingga mampu terhubung ke jaringan Wi-Fi dan berkomunikasi dengan perangkat lain melalui internet.

ii. Mikrokontroler (MCU)

Papan NodeMCU dilengkapi dengan mikrokontroler yang kuat, memungkinkan Anda menjalankan kode dan mengendalikan berbagai komponen perangkat keras.

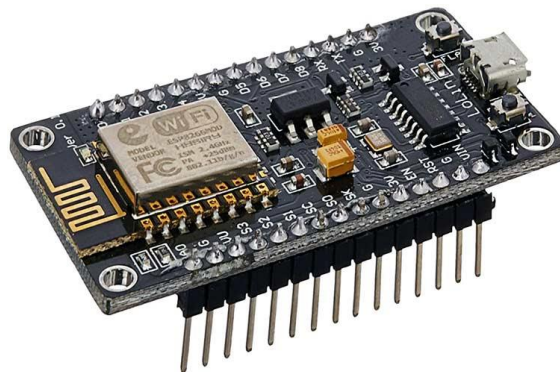
iii. Pin GPIO

Papan ini memiliki sejumlah pin Input/Output Umum (GPIO) yang dapat digunakan untuk berinteraksi dengan sensor, aktuator, dan komponen elektronik lainnya.

iv. Antarmuka USB

Terdapat port USB mikro untuk pasokan daya dan pemrograman, yang memudahkan untuk menghubungkannya ke komputer untuk keperluan pemrograman dan debugging.

Dengan ukurannya yang kecil, kemampuan Wi-Fi, dan kemudahan dalam pemrograman, NodeMCU ESP8266 banyak digunakan dalam berbagai aplikasi IoT, termasuk otomatisasi rumah, perangkat pintar, pemantauan sensor, dan banyak lagi. NodeMCU ESP8266 ditunjukkan pada Gambar 3 berikut.



Gambar 3. Modul NodeMCU ESP8266 [16]

Berikut adalah beberapa spesifikasi umum dari NodeMCU ESP8266:

i. Chipset

NodeMCU ESP8266 menggunakan chip ESP8266 sebagai inti sistemnya. Chip ini memiliki kemampuan Wi-Fi dan berfungsi sebagai mikrokontroler.

ii. CPU

ESP8266 pada NodeMCU biasanya berjalan pada kecepatan 80 MHz. Namun, ada beberapa varian yang dapat diatur ke kecepatan yang lebih tinggi atau lebih rendah.

iii. Memori

NodeMCU ESP8266 memiliki RAM dan Flash internal. Biasanya, sekitar 80KB hingga 96KB RAM tersedia untuk program dan data, serta 4MB Flash untuk menyimpan kode program.

- iv. Wi-Fi
NodeMCU ESP8266 mendukung standar Wi-Fi 802.11 b/g/n, yang memungkinkan perangkat ini untuk terhubung ke jaringan Wi-Fi.
- v. GPIO
Papan ini dilengkapi dengan beberapa pin General Purpose Input/Output (GPIO) yang dapat digunakan untuk berkomunikasi dengan berbagai perangkat dan sensor.
- vi. Analog Input
NodeMCU ESP8266 memiliki beberapa pin yang dapat digunakan sebagai masukan analog untuk membaca nilai tegangan analog dari sensor.
- vii. USB Interface
Papan ini biasanya memiliki port USB mikro (Micro USB) yang dapat digunakan untuk pasokan daya dan pemrograman.
- viii. Tegangan Operasional
NodeMCU ESP8266 biasanya beroperasi pada tegangan 3.3V. Penting untuk memperhatikan bahwa jangan memberikan tegangan lebih dari 3.3V ke pin GPIO atau pin lainnya, karena dapat merusak papan.
- ix. Firmware dan Pemrograman
NodeMCU ESP8266 dapat diprogram menggunakan berbagai Bahasa pemrograman seperti Arduino IDE (C++), Lua, atau *MicroPython*. Karena kemampuannya untuk menggunakan Arduino IDE, membuatnya sangat populer dalam komunitas Arduino.
- x. Komunikasi Serial
NodeMCU ESP8266 mendukung komunikasi serial melalui pin TX dan RX, yang memungkinkan perangkat ini berkomunikasi dengan perangkat lain melalui protokol UART.
- xi. Sumber Daya
Papan ini dapat diberi daya melalui port USB atau melalui pin VIN dengan tegangan 5V. Namun, perlu diingat untuk menggunakan tegangan 3.3V untuk menyuplai daya ke pin VIN agar sesuai dengan spesifikasi papan.
NodeMCU ESP8266 memiliki beberapa pin yang dapat dikonfigurasi sebagai Input atau Output (I/O) dengan berbagai fungsi tergantung pada kebutuhan proyek Anda. Berikut adalah beberapa pin konfigurasi yang umum pada NodeMCU ESP8266:
 - i. GPIO (General Purpose Input/Output) Pins
Papan NodeMCU ESP8266 memiliki banyak pin GPIO yang dapat digunakan sebagai pin input atau output untuk berkomunikasi dengan sensor, aktuator, atau perangkat lainnya.
 - ii. Analog Input Pins
Beberapa pin GPIO pada NodeMCU ESP8266 juga dapat diatur sebagai pin masukan analog untuk membaca data dari sensor-sensor yang menghasilkan nilai analog.

iii. I2C Pins

NodeMCU ESP8266 memiliki pin SDA (Serial Data) dan SCL (Serial Clock) yang dapat digunakan untuk menghubungkan perangkat melalui bus I2C (Inter-Integrated Circuit).

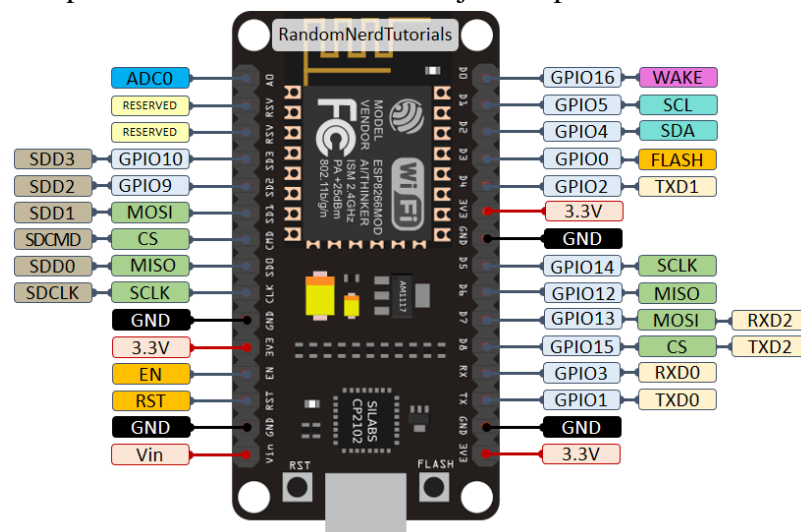
iv. SPI Pins

Papan ini memiliki pin MISO (Master In Slave Out), MOSI (Master Out Slave In), SCLK (Serial Clock), dan SS (Slave Select) yang dapat digunakan untuk berkomunikasi melalui bus SPI (Serial Peripheral Interface).

v. UART Pins

NodeMCU ESP8266 juga memiliki pin TX (Transmitter) dan RX (Receiver) untuk komunikasi serial UART.

Konfigurasi pin NodeMCU ESP8266 ditunjukkan pada Gambar 4 berikut.



Gambar 4. Konfigurasi Pin NodeMCU ESP8266 [16]

9. Sensor Ultrasonik

Sensor ultrasonik HC-SR04 adalah sebuah sensor jarak yang menggunakan gelombang ultrasonik untuk mengukur jarak antara sensor dan objek di depannya. Sensor ini terdiri dari dua bagian utama, yaitu modul pemancar dan modul penerima.

Modul pemancar menghasilkan gelombang ultrasonik yang dikirimkan ke objek. Gelombang ini kemudian dipantulkan kembali oleh objek dan diterima oleh modul penerima. Waktu yang diperlukan gelombang untuk pergi dan kembali dapat digunakan untuk menghitung jarak antara sensor dan objek dengan menggunakan prinsip kecepatan suara.

Sensor ultrasonik HC-SR04 umumnya digunakan dalam berbagai proyek elektronika, robotika, dan otomasi. Sensor ini dapat memberikan informasi jarak yang akurat dalam rentang pengukuran tertentu, biasanya dari beberapa sentimeter hingga beberapa meter.

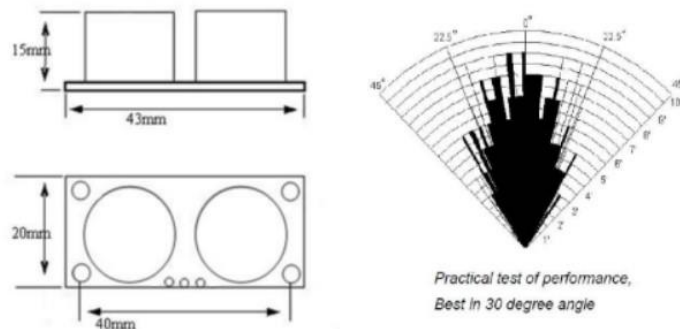
Dengan antarmuka yang sederhana dan dukungan dari berbagai platform mikrokontroler, sensor ultrasonik HC-SR04 sangat populer di kalangan pengembang dan hobiis elektronika. Sensor ini dapat digunakan untuk mendeteksi objek, mengukur jarak, dan mengaktifkan tindakan atau respons berdasarkan hasil pengukuran jarak yang diperoleh.

Sensor ultrasonik memiliki dua buah elemen, yaitu elemen pembangkit gelombang ultrasonik (TX, Transmitter) dan elemen pendeteksi gelombang ultrasonik (RX, Receiver). Transmitter memancarkan 8 gelombang dengan frekuensi 40KHz dalam waktu 10 μ S. Sensor ultrasonik dapat ditunjukkan pada Gambar 5.



Gambar 5. Sensor Ultrasonik HC-SR04 [15]

Sensor ultrasonik berfungsi untuk mengukur jarak suatu objek dengan menggunakan gelombang suara ultrasonik. Prinsip kerja dasar sensor ultrasonik yaitu dengan cara gelombang ultrasonik dipancarkan dari transceiver menuju objek yang berada di depannya dan dipantulkan kembali oleh benda ke arah sensor dan diterima oleh receiver dan benda akan menghitung jarak dengan memperkirakan waktu (saat gelombang dipancarkan dan saat gelombang pantul diterima) dan kecepatan [15]. Uji praktis performa terbaik dalam sudut 30 derajat ditunjukkan pada Gambar 6 [17].



Gambar 6. Uji Praktis Performa Terbaik Dalam Sudut 30 Derajat

Sinyal yang diterima akan diubah menjadi bentuk digital, lalu diolah menjadi bentuk jarak oleh Arduino dengan menuliskan program yang telah

dibuat pada mikrokontroller. Untuk mengukur jarak antara objek dan sensor dapat menggunakan rumus sebagai berikut:

$$S = \frac{v.t}{2} \dots\dots\dots(2.1)$$

Dimana:

s = Jarak antara objek ke sensor (m)

v = Kecepatan suara (340m/s)

t = Waktu

Sensor ultrasonik HC-SR04 menggunakan empat pin untuk konfigurasi dan koneksi. Berikut adalah konfigurasi pin yang umum digunakan:

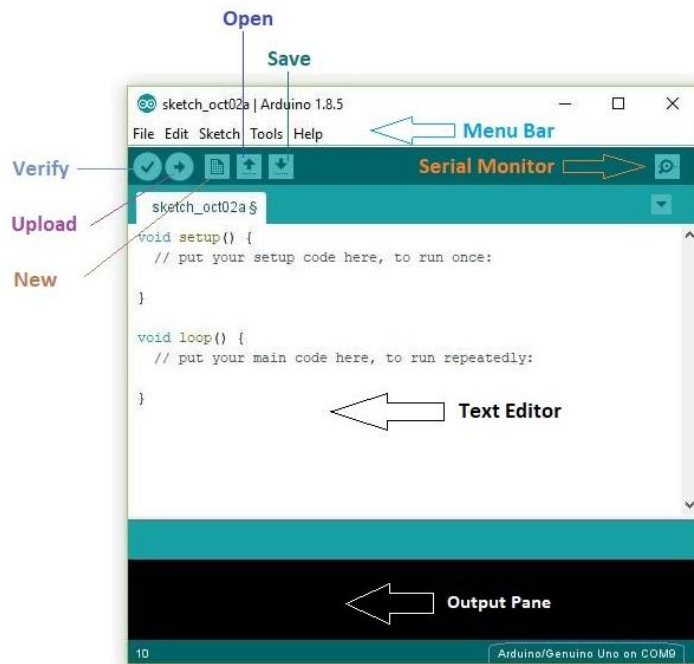
- i. VCC: Pin ini digunakan untuk menyediakan catu daya +5V untuk sensor ultrasonik. Biasanya dihubungkan ke sumber daya +5V pada mikrokontroler atau modul mikrokontroler yang digunakan.
- ii. Trig (Trigger): Pin ini digunakan untuk mengirimkan sinyal trigger ke sensor ultrasonik. Saat pin ini diberi pulsa tinggi (biasanya 10 mikrodetik), sensor akan memancarkan gelombang ultrasonik.
- iii. Echo: Pin ini digunakan untuk menerima sinyal balik atau pantulan gelombang ultrasonik dari objek yang terdeteksi. Kriteria sinyal pada pin ini menunjukkan waktu yang diperlukan gelombang untuk pergi dan kembali, yang dapat digunakan untuk menghitung jarak.
- iv. GND (Ground): Pin ini terhubung ke ground atau titik referensi negatif pada sumber daya.

Dalam pengaturan koneksi, pin Trig pada sensor harus dihubungkan ke pin output pada mikrokontroler yang akan mengirimkan sinyal trigger. Pin Echo pada sensor harus dihubungkan ke pin input pada mikrokontroler yang akan membaca sinyal balik. VCC dihubungkan ke sumber daya +5V, dan GND dihubungkan ke ground pada mikrokontroler.

10. Arduino IDE

Arduino IDE merupakan kependekan dari Integrated Development Environment merupakan perangkat lunak resmi yang diperkenalkan oleh Arduino.cc. Digunakan untuk menulis, menyusun, dan mengunggah kode di hampir semua modul/papan Arduino. Arduino IDE adalah perangkat lunak open source dan mudah tersedia untuk diunduh & dipasang dari situs resmi Arduino. Arduino IDE membuat kompilasi kode terlalu mudah sehingga bahkan orang biasa yang tidak memiliki pengetahuan teknis sebelumnya pun dapat mengikuti proses pembelajaran. Tersedia untuk semua sistem operasi yaitu MAC, Windows, Linux dan berjalan di Platform Java yang dilengkapi dengan fungsi dan perintah bawaan yang memainkan peran penting dalam

debugging, pengeditan, dan kompilasi kode. Terdapat tiga bagian dari Arduino IDE yaitu Menu, Teks, dan Output [18]. Halaman tampilan Arduino IDE ditunjukkan pada Gambar 7.

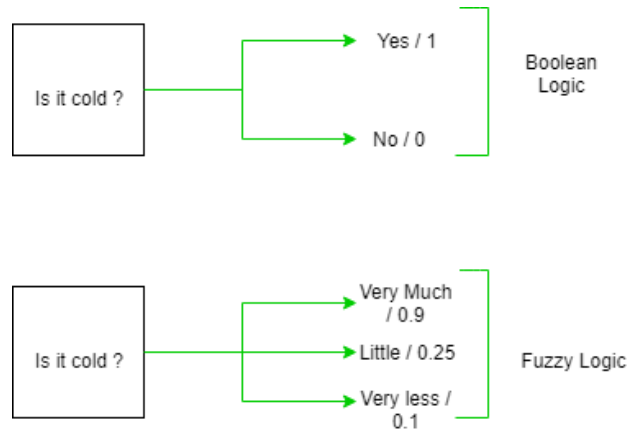


Gambar 7. Arduino IDE [18]

11. Logika *Fuzzy*

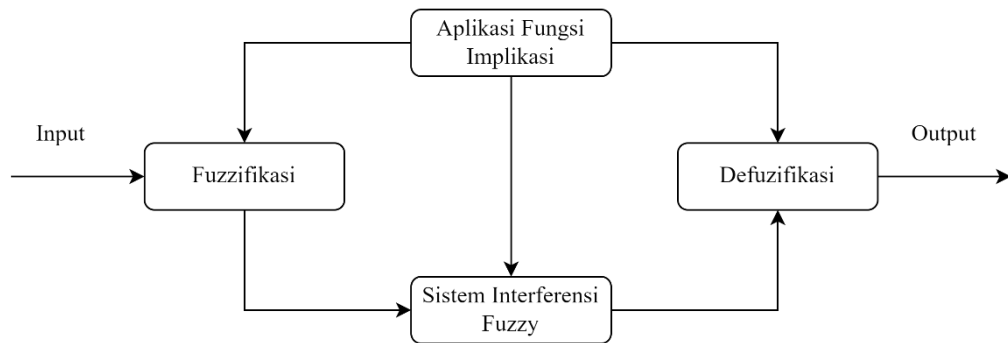
Istilah *fuzzy* mengacu pada hal-hal yang tidak jelas atau kabur. Di dunia nyata sering kali kita menghadapi situasi ketika kita tidak dapat menentukan apakah keadaan itu benar atau salah, logika *fuzzy* memberikan fleksibilitas yang baik untuk penalaran. Dengan cara ini, kita dapat mempertimbangkan ketidakakuratan dan ketidakpastian situasi apa pun [16].

Dalam nilai kebenaran sistem boolean 1 mewakili nilai kebenaran absolut dan 0 mewakili nilai salah absolut. Namun dalam sistem *fuzzy*, tidak ada logika untuk kebenaran mutlak dan nilai salah mutlak. Tetapi dalam logika *fuzzy*, ada nilai tengah yang hadir sebagian benar dan sebagian salah. Hal tersebut dapat dipahami melalui ilustrasi yang ditunjukkan pada Gambar 8.



Gambar 8. Konsep Logika *Fuzzy* [16]

Arsitektur dari logika *fuzzy* terdiri dari 4 bagian penting seperti yang ditunjukkan pada Gambar 9.



Gambar 9. Konsep Arsitektur Logika *Fuzzy* [16]

Keterangan:

i. Fuzifikasi

Fuzzifikasi merupakan pembentukan himpunan variabel input dan output menjadi satu atau lebih himpunan *fuzzy*. Himpunan *fuzzy* terdiri dari dua komponen, yaitu numeris dan linguistik. Komponen numeris ialah suatu nilai yang menunjukkan ukuran sebuah variabel. Sedangkan komponen linguistik ialah atribut yang digunakan dalam penamaan grup yang mewakili suatu keadaan tertentu menggunakan bahasa yang alami seperti dingin, hangat, dan panas. Beberapa hal yang perlu diketahui dalam memahami logika *fuzzy* yaitu:

- a. Variabel *fuzzy* yaitu variabel-variabel yang akan digunakan dalam sistem *fuzzy*.
- b. Himpunan *Fuzzy* berfungsi untuk mewakili suatu keadaan dalam suatu variabel *fuzzy*.

- c. Semesta pembicaraan ialah nilai-nilai yang terdiri dari bilangan real yang boleh digunakan untuk dioperasikan dalam variabel *fuzzy*. Nilai ini bisa berupa bilangan positif atau bilangan negatif.
- d. Domain pada himpunan *fuzzy* yaitu nilai-nilai yang boleh digunakan dalam semesta pembicaraan dan bisa dioperasikan dalam himpunan *fuzzy*. Nilai domain sama dengan nilai semesta pembicaraan yaitu berupa bilangan real positif atau negatif.

ii. Aplikasi Fungsi Implikasi

Aplikasi fungsi implikasi berisi seperangkat aturan dan kondisi IF-THEN untuk mengatur sistem pengambilan keputusan, berdasarkan informasi linguistik. Pada tahap aplikasi fungsi implikasi dilakukan penyusunan basis aturan, dimana aturan-aturan tersebut berwujud implikasi-implikasi *fuzzy* yang menyatakan hubungan antara variabel input dan variabel output.

iii. Sistem Inferensi *Fuzzy*

Sistem inferensi *fuzzy* menentukan tingkat pencocokan input *fuzzy* saat ini sehubungan dengan setiap aturan dan memutuskan aturan mana yang akan dipilih sesuai dengan bidang input. Selanjutnya, aturan yang dipilih digabungkan untuk membentuk tindakan kontrol. Jika sebuah sistem tersusun atas beberapa aturan, maka inferensi didapatkan dari kumpulan juga hubungan antar aturan. Inferensi sistem *fuzzy* dapat dilakukan dengan tiga metode berikut:

a) Metode Max (Maximum)

Dalam metode max, solusi himpunan *fuzzy* didapatkan dengan cara mengambil nilai maksimum dari aturan yang ada, lalu nilai tersebut digunakan untuk modifikasi daerah *fuzzy* lalu mengaplikasikan ke output dengan memakai operator OR (gabungan). Bila semua proporsi sudah dievaluasi, maka output akan berisi suatu himpunan *fuzzy* yang merefleksikan kontribusi dari tiap-tiap proporsi. Umumnya dapat ditulis dalam persamaan sebagai berikut:

$$\mu(x_i) = \max(\mu_{sf}(x_i), \mu_{kf}(x_i)) \dots\dots\dots (2.2)$$

Keterangan:

$\mu_{sf}(x_i)$ = nilai keanggotaan solusi *fuzzy* sampai aturan ke-*i*

$\mu_{kf}(x_i)$ = nilai keanggotaan konsekuen *fuzzy* aturan ke-*i*

b) Metode Additive (Sum)

Dalam metode additive, solusi himpunan *fuzzy* didapatkan dengan cara menjumlahkan semua hasil output dari daerah *fuzzy*.

c) Metode Probabilistik

Dalam metode probabilistic, solusi himpunan *fuzzy* didapatkan dengan cara mengalikan semua hasil output dari daerah *fuzzy*.

iv. Defuzifikasi

Defuzifikasi digunakan untuk mengubah himpunan *fuzzy* yang diperoleh oleh interferensi sistem menjadi nilai tegas. Ada beberapa metode defuzzifikasi yang tersedia dan yang paling cocok digunakan khusus untuk mengurangi kesalahan. Nilai input pada proses penegasan yaitu sebuah himpunan *fuzzy* yang didapat dari komposisi aturan-aturan *fuzzy*, sedangkan nilai output-nya ialah bilangan real yang tegas. Dengan demikian jika terdapat suatu himpunan *fuzzy* pada range tertentu, maka dapat diambil sebuah nilai tegas tertentu sebagai output-nya. Pada proses defuzzifikasi terdapat beberapa metode yang dapat digunakan seperti centroid, bisector, MOM, LOM, dan SOM. Salah satu metode defuzzifikasi yang dapat digunakan adalah weight average atau rata-rata bobot. Metode ini menghasilkan nilai keluaran tunggal tegas yang didapat dengan menjumlahkan perkalian antara masing-masing nilai *fuzzy* dibagi dengan jumlah nilai-nilai *fuzzy* (α -predikat). Pendekatan matematis dari metode weight average dinyatakan dalam persamaan sebagai berikut:

$$z = \frac{[(\mu v1 \times Z1) + (\mu v2 \times Z2) + \dots + (\mu v27 \times Z27)]}{\mu v1 + \mu v2 + \dots + \mu v27} \dots\dots\dots (2.9)$$

Dimana:

μv = α -predikat (hasil operasi *fuzzy* pada bobot wilayah)

Z = nilai membership *output*

12. Unified Modeling Language

UML (*Unified Modeling Language*) adalah bahasa pemodelan visual yang digunakan untuk mendokumentasikan, merancang, dan memahami sistem perangkat lunak. UML menyediakan sejumlah notasi grafis yang digunakan untuk menggambarkan berbagai aspek dari sistem, termasuk struktur, fungsi, interaksi, dan perilaku. Ini adalah alat yang sangat umum digunakan oleh para pengembang perangkat lunak untuk menyusun dan mengkomunikasikan konsep desain serta arsitektur sistem secara lebih jelas dan terstruktur. UML membantu dalam menggambarkan hubungan antara elemen-elemen sistem serta memberikan pandangan visual yang kaya terkait dengan berbagai aspek dari proyek pengembangan perangkat lunak [7].

UML (*Unified Modeling Language*) adalah suatu metode visual dalam pemodelan yang dipergunakan sebagai alat perancangan sistem berorientasi objek. UML dijuluki sebagai bahasa pemodelan yang digunakan untuk mendokumentasikan, merancang, serta memahami sistem perangkat lunak. UML menghadirkan serangkaian simbol grafis yang dimanfaatkan untuk menggambarkan berbagai aspek sistem perangkat lunak, termasuk struktur, fungsi, interaksi, dan perilaku sistem.

Pada penelitian ini, peneliti melakukan UML untuk rancangan sitem aplikasi android dengan memenuhi semua kebutuhan pengguna dengan efektif, lengkap, dan tepat. Jenis yang diambil untuk penelitian ini yaitu *Use Case* diagram, *Squence Diagram* dan *Activity* diagram dari rancangan sistem aplikasi android.