**IMPLEMENTASI SISTEM PEMANTAUAN DAN PENGENDALI SIRKULASI AIR PADA AQUAPONIK DENGAN METODE FUZZY TSUKAMOTO**

SKRIPSI

Digunakan Sebagai Syarat Maju Ujian Diploma IV

Politeknik Negeri Malang

Oleh:

SYAHDANNY ALHAMDA NIM. 1741720081



PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

JULI 2021

**IMPLEMENTASI SISTEM PEMANTAUAN DAN PENGENDALI SIRKULASI AIR PADA AQUAPONIK DENGAN METODE FUZZY TSUKAMOTO**

SKRIPSI

Digunakan Sebagai Syarat Maju Ujian Diploma IV

Politeknik Negeri Malang

Oleh:

SYAHDANNY ALHAMDA NIM. 1741720081



PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

JULI 2021

# **HALAMAN PENGESAHAN**

**IMPLEMENTASI SISTEM PEMANTAUAN DAN PENGENDALI SIRKULASI AIR PADA AQUAPONIK DENGAN METODE FUZZY TSUKAMOTO**

Disusun oleh:

SYAHDANNY ALHAMDA NIM. 1741720081

**Skripsi ini telah diuji pada tanggal**

**Disetujui oleh:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1. | Pembimbing Utama | : | Luqman Affandi, S.Kom, MMSI NIP. 19821130 201404 1 001 | ........................... |
|  |  |  |  |  |
| 2. | Pembimbing Pendamping | : | Moch. Zawaruddin Abdullah, S.ST., M.Kom.  NIP. 19890210 201903 1 019 | ........................... |
|  |  |  |  |  |
| 3. | Penguji Utama | : |  | ........................... |
|  |  |  |  |  |
| 4. | Penguji Pendamping | : |  | ........................... |

Mengetahui,

|  |  |
| --- | --- |
| Ketua Jurusan  Teknologi Informasi | Ketua Program Studi  Teknik Informatika |
| Rudy Ariyanto, S.T., M.Cs. | Imam Fahrur Rozi, S.T., M.T. |
| NIP. 19711110 199903 1 002 | |  | | --- | | NIP. 19840610 200812 1 004 | |

# **PERNYATAAN**

Dengan ini saya menyatakan bahwa pada Skripsi ini tidak terdapat karya, baik seluruh maupun sebagian, yang sudah pernah diajukan untuk memperoleh gelar akademik di Perguruan Tinggi manapun, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini serta disebutkan dalam daftar sitasi/pustaka.

|  |  |
| --- | --- |
|  | Malang, 21 Juni 2021  Syahdanny Alhamda |

# **ABSTRAK**

**Alhamda, Syahdanny**. “IMPLEMENTASI SISTEM PEMANTAUAN DAN PENGENDALI SIRKULASI AIR PADA AQUAPONIK DENGAN METODE FUZZY TSUKAMOTO”. **Pembimbing: (1) Luqman Affandi, S.Kom, MMSI, (2) Moch. Zawaruddin Abdullah, S.ST., M.Kom.**

**Skripsi, Program Studi Teknik Informatika, Jurusan Teknologi Informasi, Politeknik Negeri Malang, 2021.**

Akuaponik adalah integrasi antara budidaya tanaman secara hidroponik dengan budidaya ikan (akuakultur). Proses keberlangsungan hidup ikan nila pada system aquaponic sangat dipengaruhi oleh kekeruhan dan keasaman pH dimana kekeruhan dan pH tersebut dapat mempengaruhi kualitas air, sehingga kekeruhan dan pH tersebut harus dijaga supaya tetap berada pada kondisi yang optimal. Untuk mendapatkan kualitas air yang baik, meka diperlukan pemantauan air secara berkala. Kriteria yang terdapat pada kualitas air yang baik adalah keasaman pH, Kekeruhan, dan Ketinggian air. Sistem monitoring air merupakan sistem yang menerapkan konsep internet of things (IOT) . Internet Of Things merupakan sebuah komunikasi antar perangkat dengan konsep yang bertujuan untuk memperluas sebuah konektivitas internet yang terhubung secara terus – menerus. Sistem monitoring air melakukan pemantauan kualitas air berdasarkan kriteria ph, kekeruhan, dan ketinggian air menggunakan Nodemcu dan Arduino uno. Selain itu sistem dapat melakukan pengisian air secara otomatis berdasarkan keputusan yang dihasilkan oleh metode fuzzy tsukamoto.

**Kata Kunci** : Internet of Thing, Sistem Monitoring Air, Fuzzy Tsukamoto

# ***ABSTRACT***

***Alhamda, Syahdanny****. “IMPLEMENTATION OF WATER MONITORING AND CIRCULATION CONTROL IN AQUAPONIC WITH FUZZY TSUKAMOTO METHOD”.* ***Counseling Lecturer: (1) Luqman Affandi, S.Kom, MMSI, (2) Moch. Zawaruddin Abdullah, S.ST., M.Kom.***

*Aquaponics is the integration of hydroponic plant cultivation with fish farming (aquaculture). The process of survival of tilapia in the aquaponic system is strongly influenced by turbidity and pH acidity where turbidity and pH can affect water quality, so that turbidity and pH must be maintained in order to remain in optimal conditions. To get good water quality, it is necessary to monitor water regularly. The criteria for good water quality are pH acidity, turbidity, and water level. The water monitoring system is a system that applies the internet of things (IOT) concept. Internet of Things is a communication between devices with a concept that aims to expand an internet connectivity that is connected continuously. The water monitoring system monitors water quality based on the criteria of pH, turbidity, and water level using Nodemcu and Arduino uno. In addition, the system can automatically fill water based on the decisions generated by the Tsukamoto fuzzy method.*

***Keywords****: Internet of Things, Water Quality, Fuzzy Tsukamoto*

# **KATA PENGANTAR**

Puji Syukur kami panjatkan kehadirat Allah SWT atas segala rahmat dan hidayah-Nya penulis dapat menyelesaikan skripsi dengan judul “IMPLEMENTASI SISTEM PEMANTAUAN DAN PENGENDALI SIRKULASI AIR PADA AQUAPONIK DENGAN METODE FUZZY TSUKAMOTO”. Skripsi ini penulis susun sebagai persyaratan untuk menyelesaikan studi program Diploma IV Program Studi Teknik Informatika, Jurusan Teknologi Informasi, Politeknik Negeri Malang.

Kami menyadari bahwasannya dengan tanpa adanya dukungan dan kerja sama dari berbagai pihak, kegiatan skripsi ini tidak akan dapat berjalan baik. Untuk itu, kami ingin menyampaikan rasa terima kasih kepada:

1. Bapak Rudy Ariyanto, ST., M.Cs., selaku Ketua Jurusan Teknologi Informasi.
2. Bapak Imam Fahrur Rozi, ST., MT., selaku Ketua Program Studi Teknik Informatika.
3. Bapak Luqman Affandi, S.Kom, MMSI., selaku pembimbing utama.
4. Bapak Moch. Zawaruddin Abdullah, S.ST., M.Kom. selaku pembimbing pendamping.
5. Kedua Orang Tua yang telah mendukung dan mendo’akan kelancaran dalam proses penyelesaian skripsi.
6. Teman-teman yang telah memberikan bantuan dan dukungan dalam membantu menyelesaikan skripsi ini.
7. Dan seluruh pihak yang telah membantu dan mendukung lancarnya pembuatan Skripsi dari awal hingga akhir yang tidak dapat kami sebutkan satu persatu.

Penulis menyadari bahwa dalam penyusunan laporan akhir ini, masih banyak terdapat kekurangan dan kelemahan yang dimiliki penulis baik itu sistematika penulisan maupun penggunaan bahasa. Untuk itu penulis mengharapkan saran dan kritik dari berbagai pihak yang bersifat membangun demi penyempurnaan laporan ini. Semoga laporan ini berguna bagi pembaca secara umum dan penulis secara khusus. Akhir kata, penulis ucapkan banyak terima kasih.

Malang, 21 Juni 2021

Syahdanny Alhamda

# **DAFTAR ISI**

Halaman

[HALAMAN PENGESAHAN iii](#_Toc78059403)

[PERNYATAAN iv](#_Toc78059404)

[ABSTRAK v](#_Toc78059405)

[*ABSTRACT* vi](#_Toc78059406)

[KATA PENGANTAR vii](#_Toc78059407)

[DAFTAR ISI ix](#_Toc78059408)

[DAFTAR GAMBAR xii](#_Toc78059409)

[DAFTAR TABEL xiii](#_Toc78059410)

[DAFTAR LAMPIRAN xiv](#_Toc78059411)

[BAB I. PENDAHULUAN 1](#_Toc78059412)

[1.1 Latar Belakang 1](#_Toc78059413)

[1.2 Rumusan Masalah 2](#_Toc78059414)

[1.3 Batasan Masalah 3](#_Toc78059415)

[1.4 Tujuan 3](#_Toc78059416)

[1.5 Sistematika Penulisan 3](#_Toc78059417)

[BAB II. LANDASAN TEORI 5](#_Toc78059418)

[2.1 Studi Literatur 5](#_Toc78059419)

[2.2 Dasar Teori 6](#_Toc78059420)

[2.2.1 *Internet of Things* (IoT) 6](#_Toc78059421)

[2.2.2 *Java* 6](#_Toc78059422)

[2.2.3 NodeMCU ESP8266 6](#_Toc78059423)

[2.2.4 Arduino Uno R3 7](#_Toc78059424)

[2.2.5 Sensor pH 7](#_Toc78059425)

[2.2.6 Sensor Ultrasonik 7](#_Toc78059426)

[2.2.7 Sensor Turbidity 7](#_Toc78059427)

[2.2.8 *Firebase Database* 7](#_Toc78059428)

[2.2.9 Logika *Fuzzy* 8](#_Toc78059429)

[BAB III. METODOLOGI PENELITIAN 10](#_Toc78059430)

[3.1. Teknik Pengumpulan Data 10](#_Toc78059431)

[3.2.1 Data 11](#_Toc78059432)

[3.2.2 Metode Pengambilan Data 11](#_Toc78059433)

[3.2.3 Teknik Pengolahan Data 11](#_Toc78059434)

[3.2 Metode Pengembangan Perangkat Lunak 12](#_Toc78059435)

[3.4 Uji Coba Sistem 14](#_Toc78059436)

[BAB IV. ANALISIS DAN PERANCANGAN SISTEM 15](#_Toc78059437)

[4.1 Analisis 15](#_Toc78059438)

[4.1.1 Simulasi Perhitungan Metode *Fuzzy Tsukamoto* 16](#_Toc78059439)

[4.1.2 Deskripsi Sistem 20](#_Toc78059440)

[4.1.3 Analisis Pengguna 21](#_Toc78059441)

[4.1.4 Kebutuhan Fungsional 21](#_Toc78059442)

[4.1.5 Kebutuhan Non-Fungsional 22](#_Toc78059443)

[4.1.6 Kebutuhan Perangkat Keras (*Hardware*) 22](#_Toc78059444)

[4.1.7 Kebutuhan Perangkat Lunak (*Software*) 23](#_Toc78059445)

[4.2 Perancangan 24](#_Toc78059446)

[4.2.1 Perancangan Usecase Diagram 24](#_Toc78059447)

[4.2.2 Perancangan *Activity Diagram* 25](#_Toc78059448)

[4.2.3 Designer Database 31](#_Toc78059449)

[4.2.4 Arsitektur Sistem 32](#_Toc78059450)

[4.2.5 Flowchart Sistem 33](#_Toc78059451)

[4.2.6 Flowchart Perhitungan Metode 34](#_Toc78059452)

[4.2.7 Desain Tampilan 34](#_Toc78059453)

[BAB V. IMPLEMENTASI DAN PENGUJIAN 37](#_Toc78059454)

[5.1 Implementasi 37](#_Toc78059455)

[5.1.1 Implementasi *Database* 37](#_Toc78059456)

[5.1.2 Implementasi Kode Program 38](#_Toc78059457)

[5.1.3 Implementasi Tampilan Sistem 72](#_Toc78059458)

[5.2 Pengujian 74](#_Toc78059459)

[5.2.1 Pengujian Fungsional Sistem 74](#_Toc78059460)

[5.2.2 Pengujian Validitas Perhitungan 89](#_Toc78059461)

[BAB VI. HASIL DAN PEMBAHASAN 91](#_Toc78059462)

[6.2 Hasil Penelitian 91](#_Toc78059463)

[6.2 Pembahasan 92](#_Toc78059464)

[BAB VII. KESIMPULAN DAN SARAN 94](#_Toc78059465)

[7.1 Kesimpulan 94](#_Toc78059466)

[7.2 Saran 94](#_Toc78059467)

[DAFTAR PUSTAKA 95](#_Toc78059468)

# **DAFTAR GAMBAR**

# **DAFTAR TABEL**

# **DAFTAR LAMPIRAN**

# **BAB I. PENDAHULUAN**

## **Latar Belakang**

Semakin berkembangnya teknologi pada zaman sekarang, semakin berkembang pula alternatif bercocok tanam. Salah satu teknologi yang tepat dikembangkan adalah teknologi akuaponik. Akuaponik adalah integrasi antara budidaya tanaman secara hidroponik dengan budidaya ikan (akuakultur). Dari hasil observasi bahwa di Desa Slamet yang berada di Kabupaten Malang, tepatnya berada di Kecamatan Tumpang, masih menggunakan sistem berkebun pada umumnya. Oleh karena itu dibutuhkan suatu inovasi baru yaitu pemanfaatan lahan yang ada dengan menggunakan sistem aquaponik.

Aquaponic memberikan alternatif bercocok tanam dilahan terbatas dengan menggabungkan akuakultur dan hidroponik dalam lingkungan yang simbiotik. Nutrisi akuaponik bisa didapat dengan mudah, yaitu diperoleh dari kotoran ikan. Umumnya pada akuakultur eksresi dari ikan yang dipelihara akan terakumulasi di air dan meningkatkan toksisitas jika tidak dibuang, dalam akuaponik, kotoran ikan akan dipecah menjadi nitrat dan nitrit melalui proses alami dan dimanfaatkan oleh tanaman sebagai sumber nutrisi. Dalam kegiatan ini sistem hidroponik berperan sebagai filter bagi lingkungan ikan(Rahayu et al., 2018).

Proses keberlangsungan hidup ikan nila pada system aquaponic sangat dipengaruhi oleh kekeruhan dan keasaman pH dimana kekeruhan dan pH tersebut dapat mempengaruhi kualitas air, sehingga kekeruhan dan pH tersebut harus dijaga supaya tetap berada pada kondisi yang optimal. Hal tersebut jika dilakukan secara konvensional atau datang ke tempat kolam aquaponik secara langsung akan membutuhkan waktu dan energi yang banyak karena harus melihat informasi kekeruhan dan pH secara langsung pada kolam. Untuk memaksimalkan waktu dan energi dalam mengukur kekeruhan, pH, dan ketinggian air maka dibuatlah sebuah alat otomatis untuk memonitoring kolam aquaponik yang terintegrasi dengan internet, melihat kebutuhan pemilik aquaponik yang tidak selalu dapat mengontrol kondisi air secara langsung. Sehingga pengguna dapat memperoleh informasi dari aplikasi yang telah dibangun untuk memonitoring kolam aquaponic tersebut dimanapun.

Terdapat banyak sensor yang dapat mengidentifikasi tingkat keasaman, kekeruhan, dan ketinggian air dengan memanfaatkan penggunaan *Internet of Things* (IoT). *Internet of Thing* merupakan sebuah teknologi yang menerapkan konsep memperluas manfaat sebuah konektivitas untuk berbagi data dalam jangkau yang sangat luas. Dengan menggunakan Arduino Uno sistem dapat digunakan untuk memonitoring air bersih dengan mengambil data kadar keasaman pH, kekeruhan, dan ketinggian kemudian menyimpan dan menampilkannya pada sebuah aplikasi android secara *realtime*. Sistem Monitoring air bersih ini bertujuan untuk memberikan sebuah keputusan dalam melakukan pergantian air melalui sebuah keputusan yang dianalisa menggunakan metode fuzzy tsukamoto.

Metode Fuzzy merupakan metode yang digunakan untuk data sensor yang bersifat bukan biner dan non linier yang bertujuan sebagai pengambil keputusan untuk memberikan perintah menambahkan air atau mengganti air berdasarkan kadar pH, kekeruhan dan ketinggian air. Metode Fuzzy Tsukamoto mempunyai keunggulan yaitu dapat memberikan keputusan berdasarkan yang bersifat kualitatif, tidak akurat dan ambigu. Berdasarkan hal tersebut peneliti ingin menbangun “Implementasi Sistem Pemantauan dan Pengendali Sirkulasi Air pada Aquaponik dengan Metode Fuzzy Tsukamoto” yang bertujuan untuk memantau dan mengontrol kejernihan air sehingga sistem aquaponik benerja dengan maksimal. Dengan penelitian ini dapat memberikan manfaat dalam melakukan pemantauan dan pengontrolan air bersih pada aquaponik.

## **Rumusan Masalah**

Berdasarkan latar belakang di atas, maka rumusan masalah yang dapat diambil adalah sebagai berikut:

1. Bagaimana cara untuk mengevaluasi kondisi kebersihan air dalam kolam aquaponik?
2. Bagaimana sistem aquaponik dapat membaca ph air, ketinggian air, dan kekeruhan sehingga dapat memberikan sebuah keputusan?
3. Bagaimana sistem aquaponic dapat dikontrol dan di awasi secara real time dengan *smartphone*?

## **Batasan Masalah**

Permasalahan yang dibahas pada penelitian ini memiliki beberapa batasan sebagai berikut :

1. Sistem hanya dapat memantau dan mengevaluasi kadar air yang ada di kolam.
2. Sistem hanya dapat memantau ph air, ketinggian air, dan kekeruhan air.
3. Sistem tidak dapat membersihkan secara otomatis ketika kolam kotor.
4. Sistem hanya dapat menambahkan air secara otomatis ketika volume air berkurang.

## **Tujuan**

Tujuan dari dilakukaannya penelitian dengan judul “Implementasi Sistem Pemantauan dan Pengendali Sirkulasi Air Pada Aquaponik Dengan Metode Fuzzy Tsukamoto”, adalah sebagai berikut:

1. Mewujudkan sistem aquaponic yang lebih efisien dan terotomatisasi menggunakan mikrokontroler arudino.
2. Membuat sebuah rancang bangun sistem aquaponic yang dapat di pantau langsung menggunakan *smartphone*.
3. Memberikan keputusan untuk mengganti air

## **Sistematika Penulisan**

BAB I PENDAHULUAN

Pada bab ini berisi latar belakang, rumusan masalah, tujuan , batasan masalah, serta sistematika penulisan yang merupakan dasar mengapa penelitian ini dilakukan.

BAB II LANDASAN TEORI

Pada bab ini membahas tentang teori-teori yang bersifat relevan dan mendukung dalam proses pengerjaan tugas proyek akhir, seperti konsep sistem pakar, konsep *artificial intelligence* / kecerdasan buatan, dan konsep *certainty factor*. Teori-teori ini merupakan pustaka yang bersumber dari jurnal ilmiah, prosiding dan lain-lain.

BAB III METODOLOGI

Pada bab ini membahas tentang deskripsi proyek akhir, alur pengerjaan proyek akhir, identifikasi data, serta proses perhitungan *certainty factor*.

BAB IV ANALISIS DAN PERANCANGAN

Pada bab ini membahas tentang simulasi dan analisis perencanaan. Melakukan perancangan dalam membuat sistem dalam penelitian dan menganalisis apa saja yang dibutuhkan dalam proses penelitian.

BAB V IMPLEMENTASI DAN PENGUJIAN

Pada bab ini membahas tentang kesimpulan dari pengerjaan proyek akhir dan saran untuk pembaca yang akan mengambil penelitian dengan topik yang sama.

BAB VI. HASIL DAN PEMBAHASAN

Pada bab ini membahas tentang hasil yang didapat dari penelitian.

BAB VII. KESIMPULAN DAN SARAN

Pada bab ini membahas tentang kesimpulan dari pengerjaan proyek akhir dan saran untuk pembaca yang akan mengambil penelitian dengan topik yang sama.

# **BAB II. LANDASAN TEORI**

## **2.1 Studi Literatur**

Beberapa penelitian yang dijadikan rujukan dalam penelitian ini adalah sebagai berikut:

Jurnal penelitian Arief Prasetyo dkk tahun 2018 yang berjudul “Implementasi IoT pada Sistem Monitoring dan Pengendali Sirkulasi Air Tanaman Hidroponik” menghasilkan kesimpulan bahwa penelitian telah berhasil membuat sistem yang digunakan untuk monitoring dan mengendalikan sirkulasi air hidroponik secara otomatis dengan fuzzy sugeno melalui antarmuka website. Sistem sangat sesuai dan berjalan dengan baik pada hidroponik *Deep Flow Technic* (DFT), dikarenakan sirkulasi air dapat berjalan dengan lebih efektif ketika kondisi tertentu. Dan berdasarkan perbandingan hasil pengamatan terhadap pertumbuhan tanaman sawi daging dalam dua kondisi menghasilkan perbedaan hasil pertumbuhan tinggi dan jumlah daun yang cukup signifikan.(Prasetyo et al., 2018)

Pada jurnal penelitian Nina Rahayu dkk tahun 2018 yang berjudul “Rancang Bangun Sistem Kontrol dan Pemantauan Aquaponik Berbasis *IoT* pada Kelurahan Kutajaya” menghasilkan kesimpulan bahwa sistem aquaponic dapat membaca suhu dan kelembaban disekitarnya dengan menggunakan sensor DHT22 lalu mengguakan sensor ultrasonik untuk mengukur ketinggian air yang ada dikolam ikan aquaponic. dan sensor ESP8266 sebagai modul wifi berguna untuk mengkoneksikan alat dengan *smartphone* agar dapat mengirimkan hasil scan sensor ke *smartphone*.(Rahayu et al., 2018)

Penelitian Kuat Indartono dkk tahun 2020 yang berjudul “Perancangan Sistem Pemantau Kualitas Air pada Budidaya Ikan Air Tawar” menghasilkan kesimpulan bahwa Prototype sistem pemantauan dan pengendalian kualitas air berhasil dibuat dan sensor bekerja normal pada saat dilakukan pengujian yang tampilan oleh LCD. Prototype yang dibuat bisa digunakan untuk memantau secara *real time*.(Indartono et al., 2020)

Penelitian ini terfokus pada pemantauan kualitas air terhadap perkembangan ikan nila dan tanaman hidroponik yang ada dengan menggunakan metode Fuzzy Tsukamoto. Pemantauan air tersebut akan menghasilkan sebuah keputusan untuk melakukan sebuah tindakan yang harus dilakukan untuk menambah air yang ada pada kolam aquaponik.

## **2.2 Dasar Teori**

Dasar teori merupakan bagian yang membahas beberapa teori-teori yang berkaitan dengan masalah yang diangkat.

### ***Internet of Things* (IoT)**

*Internet of Thing* merupakan era baru dalam bidang internet, secara baris besar IoT menggambarkan peralatan elektronik yang kita gunakan sehari-hari yang dihubungkan ke dalam jaringan computer untuk meningkatkan kegunaan internet dengan menghubungkan semua barangbarang elektronik untuk dapat berikteraksi melalui embedded sistem.(Adriantantri & Irawan, 2018)

*Internet of Things* (IoT) merupakan kumpulan bendabenda (*things*), berupa perangkat fisik (hardware/embedded sistem) yang mampu bertukar informasi antar sumber informasi, operator layanan ataupun perangkat lainnya yang terhubung kedalam sistem sehingga dapat memberikan kemanfaatan yang lebih besar.(Kuswinta et al., 2019)

### ***Java***

Java adalah suatu teknologi didunia software komputer, selain merupakan suatu bahasa pemrograman, java juga merupakan platform. Sedangkan Netbeans adalah sebuah IDE (*Integrated Development Environment*) *open source* yang seringkali diasosiasikan dengan Java. Akan tetapi bisa digunakan juga pada proyek seperti *web service* menggunakan PHP, Python, Ruby dan lain-lain .MySQL merupakan sistem database yang banyak digunakan untuk pengembangan aplikasi web.(Fatkhudin, 2016)

### **NodeMCU ESP8266**

NodeMCU ESP8266 adalah sebuah mikrokontroler yang dikhususkan untuk IoT ( *Internet of things* ) seperti arduino yang sudah dilengkapi dengan WiFi dan bersifat *open source*. Pengembangan alat ini didasarkan pada modul ESP8266. NodeMCU ini mengintegerasikan PWM (*Pulse Width Modulation*), GPIO, I2C, dan ADC (*Analog Digital Converter*) dimana semua ada pada satu board.(Kurniatuty & Widodo, 2015)

### **Arduino Uno R3**

Arduino Uno R3 merupakan sebuah papan pengembangan chip Atmega328P yang bersifat open source. Arduino Uno memiliki 14 digital pin input / *output* (I/O) dimana pin 14 merupakan pin *output* PWM. Arduino uno memiliki 6 input analog, menggunakan crystal 16 MHz antara lain pin A0 sampai A5, koneksi USB, jack listrik, header ICSP dan tombol reset.

### **Sensor pH**

Sensor pH merupakan sensor yang dipakai untuk mengukur derajat keasaman pada suatu larutan. Sensor ini mengkonversi besaran pH menjadi besaran listrik. Jenis sensor ph yang dipakai yaitu sensor ph yang memakai elektroda gelas.(Kurniatuty & Widodo, 2015)

### **Sensor Ultrasonik**

Menurut Kumar Ultrasonic sensors are devices that use electrical – mechanical energy transformation to measure distance from the sensor to the target object. Sedangkan menurut Santoso sensor ultrasonik adalah sebuah sensor yang berfungsi untuk mengubah besaran fisis (bunyi) menjadi besaran listrik dan sebaliknya. (Rahayu et al., 2018)

### **Sensor Turbidity**

Turbidimeter adalah alat yang dipakai sebagai uji standar untuk mengetahui tingkat kekeruhan air. Di alat sensor tersebut ada sejenis sensor sumber cahaya dan penangkap cahaya, yang kemudian dilewatkan ke 3 bagian air yang akan di lakukan pengukuran kekeruhan. Sensor ini dapat dihubungkan ke perangkat pengolah instrument pengukuran seperti ke mikrokontroller ataupun ke Arduino.(Kurniatuty & Widodo, 2015)

### ***Firebase Database***

Adalah *Backend as a Service* (BaaS) yang saat ini dimiliki oleh Google. Firebase merupakaan solusi yang ditawarkan oleh Google untuk mempermudah pengembangan aplikasi *mobile*. Dua fitur menarik dari Firebase adalah Firebase Remote Config dan Firebase Real Time Database. Selain itu juga terdapat fitur pendukung untuk aplikasi yang memerlukan push notification yaitu *Firebase Notification Console. Firebase Database* merupakan penyimpanan basis data nonSQL yang memungkinan untuk menyimpan beberapa tipe data. Tipe data itu antara lain String, Long, dan Boolean. Data pada Firebase Database disimpan sebagai objek JSON tree. Tidak seperti basis data SQL, tidak ada tabel dan baris pada basis data non-SQL. Ketika ada penambahan data, data tersebut akan menjadi node pada struktur JSON. Node merupakaan simpul yang berisi data dan bisa memiliki cabang-cabang berupa node lainnya yang berisi data pula. Proses pengisian suatu data ke Firebase Database dikenal dengan istilah push.(Sandy et al., 2017)

### **Logika *Fuzzy***

Logika *fuzzy* adalah salah satu cabang ilmu kecerdasan buatan untuk membangun sistem cerdas. Logika *fuzzy* sering digunakan dalam pemecahan masalah yang menjelaskan sistem bukan melalui angka-angka, melainkan secara linguistik, atau variabel - variabel yang mengandung ketakpastian/ketidaktegasan. Nilai-nilai yang bersifat tidak pasti ini berdasarkan penalaran yang mengkombinasikan variabel numerik, variabel linguistik, dan aturan-aturan. Didalam logika klasik menyatakan bahwa segala sesuatu bernilai biner, yang artinya hanya mempunyai dua kemungkinan yaitu “Ya” atau “Tidak”, “Benar” atau “Salah”, “Baik” atau “Buruk”, dan sebagainya. Oleh karena itu *fuzzy* mempunyai nilai keanggotaan 0 – 1. Tetapi, logika *fuzzy* dimungkinkan memiliki nilai keanggotaan berada pada antara 0 - 1. Artinya, satu keadaan dapat memiliki dua nilai “Ya” dan “Tidak” secara bersamaan, namun besar nilainya tergantung pada bobot keanggotaan yang dimilikinya (Rohadi, Apriyani, & Laili, 2019).

Logika fuzzy dapat digunakan dalam berbagai bidang, seperti pada sistem diagnosis penyakit, pemodelan sistem pemasaran, riset operasi, kendali pengairan, prediksi gempa, klasifikasi dan pencocokan pola (Qirom et al., 2019). Pada Penelitian ini menggunakan sebuah metode Fuzzy yang bernama Fuzzy Tsukamoto. Fuzzy Tsukamoto merupakan secara umum metode fuzzy tsukamoto terdapat 4 tahapan yaitu :

*Fuzzifikasi*

Fuzzifikasi merupakan suatu proses perubahan masukan yang memiliki nilai bersifat tegas (crisp) ke dalam bentuk fuzzy(variabel linguistik) yang disajikan dalam bentuk himpunan – himpunan fuzzy dalam fungsi keanggotaan.

*Inference*

Inference merupakan proses penalaran pada fuzzy input dengan fuzzy rules yang dibuat sebelumnya. Aturan fuzzy merupakan sebuah aturan yang melakukan pengondisian IF ancendence THEN consequent

*Agregasi*

Agregasi merupakan proses yang diperoleh dari kumpulan dan korelasi antar aturan – aturan yang melakukan penghitungan berdasarkan banyaknya aturan atau rules menggunakan rumus :

(2.1)

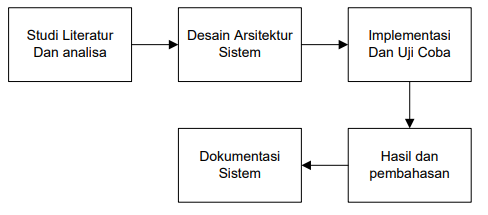
*Defuzzication*

Defuzzication merupakan sebuah proses pada fuzzy tsukamoto yang melakukan pengubahan nilai fuzzy *output* menjadi sebuah crisp value yang diperoleh dengan cara mengambil nilai rata – rata yang terpusat sebagai keputusan yang diberikan oleh sistem.

(2.2)

# **BAB III. METODOLOGI PENELITIAN**

Pada metode penelitian ini dijelaskan tentang metode yang digunakan dan konsep pembuatan keseluruhan sistem, serta melakukan analisa hasil yang diperoleh. Pada penelitian ini digunakan perangkat keras NodeMCU ESP8266 dengan menggunakan *Firebase Database* dan sistem berbasis *mobile*. Adapun konsep atau tahapan–tahapan yang dilakukan dalam penelitian ini agar dapat diperoleh hasil yang sistematis, akurat, dan sesuai yang diharapkan. Tahapan–tahapan metodologi digambarkan dalam bentuk diagram seperti pada Gambar 3.1.1.



Gambar 3.1 Tahapan-tahapan Metodologi

Pada tahapan studi literatur, peneliti melakukan pencarian referensi atau sumber terkait dengan akuaponik dan konsep–konsep *internet of thing* pada jurnal dan penelitian terdahulu yang nantinya akan digunakan sebagai acuan dalam melakukan pengembangan sistem dan melakukan analisa kebutuhan yang digunakan untuk membangun sistem. Lalu pada tahapan Desain Arsitektur Sistem, peneliti melakukan perancangan baik komunikasi data antar perangkat maupun rancangan antarmuka sistem. Lalu dilanjutkan pada tahap implementasi, dimana peneliti melakukan pengembangan sistem dan perangkat yang dibuat. Kemudian melakukan pengujian pada hasil yang didapatkan dan mendokumentasikan dalam bentuk laporan skripsi.

## **Teknik Pengumpulan Data**

Dalam penelitian ini terdapat beberapa cara teknik pengumpulan data mengenai data apa saja yang diolah dan bagaimana cara pengambilan serta pengumpulan data, sebagai berikut :

### **Data**

Sumber data yang diolah pada penelitian ini yaitu berdasarkan pada pengambilan data yang diperoleh dengan menggunakan sensor–sensor ketika penelitian berlangung dan dikirimkan pada firebase database menggunakan NodeMCU ESP8266. Data sensor tersebut berupa sensor pH, sensor kekeruhan, dan sensor ketinggian.

### **Metode Pengambilan Data**

Metode pengambilan data dilakukan melalui beberapa cara sebagai berikut :

1. Studi literarur tentang bagaiman cara melakukan perawatan pada kolam aquaponik yaitu mengumpulkan sebuah data dari jurnal–jurnal, penelitian, dan artikel yang terkait dengan penelitian.
2. Memahami prinsip–prinsip pada perangkat keras yang digunakan pada penelitian sebelumnya.
3. Memahami metode–metode fuzzy Tsukamoto yang digunakan pada penelitian sebelumnya dan menerapkannya pada penelitian.

### **3.2.3 Teknik Pengolahan Data**

Data yang didapatkan dari sensor dan dikirimkan oleh NodeMCU ESP8266 akan dianalisa menggunakan metode Fuzzy Tsukamoto dan menghasilkan sebuah keputusan yang harus dilakukan oleh sistem yaitu : Tambah Air Sedikit, Tambah Air Sedang, Tambah Air Banyak. Berdasarkan pengolahan data terdapat langkah-langkah sebagai berikut :

Fuzzyfication

Proses digunakan untuk mengubah informasi dari inputan data dari sensor ke data himpunan fuzzy linguistic.

Interference

Pada tahapan inteference melakukan proses penalaran pada fuzzy input dengan fuzzy rules . Pada Tabel 3.2 merupakan aturan – aturan fuzzy yang digunakan pada penelitian ini.

Agregasi

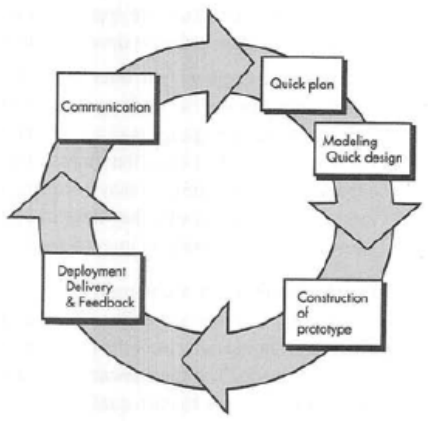
Setelah proses inferensi maka proses selanjutnya sebelum defuzzifikasi adalah proses aggregasi, yang mana menggabungkan semua *output* rule baik itu yang di scaling atau di clipping kedalam satu fuzzy set. Input dari proses agregasi ini adalah semua *output* dari consequent yang di scaling ataupun di clipping dan *output* dari proses ini adalah satu fuzzy set.

Defuzzification

Pada tahap ini merupakan proses pengubahan nilai fuzzy *output* menjadi sebuah crisp value yang diperoleh dengan cara mengambil nilai rata – rata yang terpusat sebagai keputusan yang diberikan oleh sistem

## **Metode Pengembangan Perangkat Lunak**

Metode pengembangan perangkat lunak yang diterapkan pada penelitian ini adalah dengan menggunakan Metode *Prototype*. *Prototype* merupakan salah satu metode pengembangan perangat lunak yang cukup banyak digunakan dalam pengembangan sistem karena prosesnya terbilang cepat. Dengan metode *prototype* ini *developer* dan *user* dapat saling berinteraksi selama proses pembuatan sistem. Metode *Prototype* dimulai dengan pengumpulan kebutuhan *user* terhadap perangkat lunak yang akan dibuat, mendefinisikan objektif keseluruhan dari software, mengidentifikasikan segala kebutuhan, kemudian dilakukan perancangan cepat yang difokuskan pada penyajian aspek yang diperlukan agar *user* lebih tergambarkan dengan apa yang sebenarnya diinginkan.(Mubarok et al., 2015) Diagram alur dari Metode *Prototype* yang terdapat pada Gambar 3.3.1 memiliki tahapan sebagai berikut :



Gambar 3.1.1 Metode Prototype (Sumber : (Syaddad, 2017)\*)

Tahapan dalam pengembangan perangkat lunak menggunakan Metode *Prototype* :

*Communication (*Komunikasi / identifikasi kebutuhan*).* Merupakan tahapan analisa terhadap kebutuhan pengembangan sistem. Studi literarur tentang bagaiman cara melakukan perawatan pada kolam aquaponik dengan mengumpulkan sebuah data dari jurnal–jurnal, penelitian, dan artikel yang terkait dengan penelitian. Memahami prinsip–prinsip pada perangkat keras yang digunakan pada penelitian sebelumnya serta memahami metode–metode fuzzy Tsukamoto yang digunakan pada penelitian sebelumnya dan menerapkannya pada penelitian. Pada tahap ini juga mengumpulkan informasi yang dibutuhkan sehingga sistem akan berjalan sesuai dengan keinginan atau tujuan dari pembangunan dan pengembangan sistem. Menentukan tujuan awal dari sistem yang akan dibangun, kebutuhan yang diinginkan dan hal-hal yang dibutuhkan.

*Quick Plan* (Perencanaan Cepat). Merupakan tahapan perancangan dilakukan dengan cepat dan mewakili keseluruhan sistem secara garis besar yang menjadi solusi pemecahan masalah serta dasar pembuatan *prototype*.

*Modelling Quick Design* (Pemodelan Desain Cepat). Merupakan tahapan desain dari keseluruhan sistem yang akan dikembangkan.

*Construction of Prototype* (Konstruksi Prototipe).Merupakan tahapan penerapan kode program pada proses pembuatan sistem. Sistem akan dibuat berdasarkan desain dan tujuan agar sistem berjalan sesuai dengan yang diinginkan.

*Deployment Delivery & Feedback* (Pengujian, Penyebaran & Umpan Balik). Merupakan tahapan akhir dari metode pengembangan perangkat lunak *Prototype.* Sistem yang telah selesai dibangun akan dilakukan proses pengujian, apakah sistem berjalan sesuai dengan yang direncanakan atau masih ada beberapa kesalahan. Setelah itu sistem akan disebar dan diberikan kepada *user* untuk dievaluasi, kemudian dapat memberikan feedback yang akan digunakan untuk merevisi kebutuhan software yang dibangun.

## **Uji Coba Sistem**

Sistem akan diuji hasil keputusannya pada aquarium berukuran 40x20x25. Proses pengujian Monitoring pada penelitian ini dilakukan dengan menggunakan Metode *Black Box* *Testing* dan Pengujian Validitas. Pengujian *Black Box* merupakan teknik pengujian yang dilakukan uji coba mengenai fungsionalitas keseluruhan fitur yang tersedia. Pengujian Validitas merupakan teknik pengujian untuk menguji hasil keputusan dari sistem yang dibandingkan dengan hasil pengujian perhitungan. Akurasi uji coba didapatkan dengan rumus persamaan 3.1, sebagai berikut :

(3.1)

Jumlah akurasi dihitung berdasarkan jumlah kesesuaian dan keberhasilan dibanding dengan total sampel ∑ 𝑇𝑜𝑡𝑎𝑙 𝑆𝑎𝑚𝑝𝑒𝑙 mengacu pada rumus yang direpresentasikan pada persamaan 3.1. Hasil akurasi dinyatakan dalam satuan persen (%) dapat dikatan baik jika besaran tingkat akurasi diatas 70%

# **BAB IV. ANALISIS DAN PERANCANGAN SISTEM**

Pada bab ini menjelaskan tentang perancangan mengenai analisis proses atau kinerja yang digunakan pada penelitian. Tahapan ini berisi tentang penjelasan– penjelasan umum sistem, dan proses yang dilakukan oleh sistem. Pada penelitian ini penggalian kebutuhan didapatkan menggunakan survey pada kolam aquaponik. Pada kolam aquaponik “Kampung Berseri” yang terletak di Kampung Rosella, Slamet Wiroto, Tumpang Kabupaten Malang. Terdapat beberapa kasus permasalahan yang terjadi yaitu diantaranya kekeruhan, pH, dan ketinggian. Pada penelitian ini alat dirancang sesuai kebutuhan yang ada dilapangan.

## **Analisis**

Berdasarkan sumber data yang telah didapatkan dari hasil studi literatur dan observasi, studi literatur digunakan untuk mengumpulkan informasi tambahan mengenai artikel tentang monitoring kolam aquaponik, memahami prinsip–prinsip pada perangkat keras yang digunakan pada penelitian sebelumnya, dan memahami metode–metode fuzzy Tsukamoto yang digunakan pada penelitian sebelumnya. Dari hasil studi literatur tersebut diperoleh beberapa variabel dan fungsi keanggotaan pada tiap variabel. Terdapat tiga jenis variabel yang digunakan dalam penelitian yaitu :

1. Kekeruhan Air
2. pH Air
3. Ketinggian Air

Ketiga variabel tersebut didapatkan dari sensor-sensor yang terhubung dengan mikrokontroller. Setelah data diterima oleh mikrokontroller, data tersebut akan dikirimkan ke *Firebase Database*. Data yang disimpan pada *Firebase Database* selanjutnya akan diproses menggunkan metode fuzzy Tsukamoto. Selanjutnya hasil dari proses tersebut nantinya akan dijalankan oleh mirkokontroller. Data dari sensor nantinya dapat dilihat pada aplikasi *mobile* berbasis android.

Setelah dilakukan analisis terhadap data yang digunakan dalam penelitian maka dibutuhkan analisis pengguna, kebutuhan *fungsional* dan *non-fungsional,* sebagai berikut :

### **4.1.1 Simulasi Perhitungan Metode *Fuzzy Tsukamoto***

Dalam melakukan perhitungan metode fuzzy tsukamoto terdapat contoh dan simulasi langkah-langkah perhitungan, sebagai berikut :

Fuzzyfication

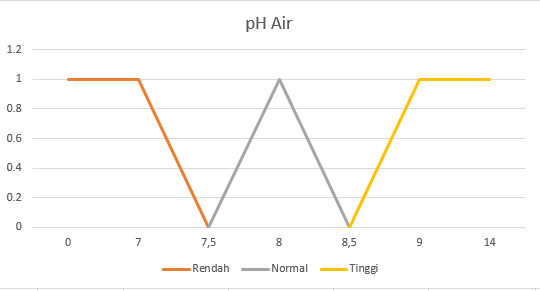
Proses digunakan untuk mengubah informasi dari inputan data dari sensor ke data himpunan fuzzy linguistic. Berikut adalah contoh inputan dari sensor yang diterima seperti pada Tabel 4.1.1.

Tabel 4.1 1 Tabel Aturan-aturan Fuzzy

|  |  |  |
| --- | --- | --- |
| Kekeruhan Air | pH Air | Ketinggian Air |
| 1 | 7.3 | 23 |

1. Himpunan keanggotaan pH

Pada himpunan keanggotaan keasaman pH , nilai input dari sensor diubah menjadi 3 bentuk himpunan yaitu himpunan pH Rendah, pH Normal, dan pH Tinggi.



Gambar 4.1.1 Fungsi Keanggotaan Himpunan Ph

Fungsi keanggotaan kadar pH Air membagi menjadi 3 bagian himpunan fuzzy diantaranya adalah ph\_rendah yaitu dengan nilai pH 0 sampai dengan 7,5. Ph\_normal dengan nilai pH 7,5 sampai dengan 8,5. Dan nilai ph\_tinggi dengan nilai pH 8,5 sampai dengan 14.

Berikut ini adalah contoh himpunan keanggotaan Keasaman air yang bernilai 7,3.

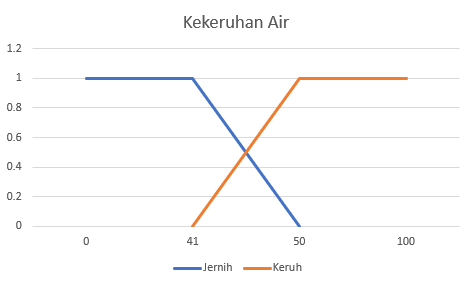
µ pH RENDAH[x] =

=

= 0,4

1. Himpunan keanggotaan kekeruhan

Pada fungsi keanggotaan kekeruhan air, sistem memberikan nilai input dari sensor berupa data digital tingkat kekeruhan air yang digolongkan menjadi 2 himpunan fuzzy yaitu : jernih, dan keruh.



Gambar 4.1.2 Fungsi Keanggotaan Himpunan Kekeruhan

Fungsi derajat kekeruhan air membagi menjadi 2 himpunan fuzzy dengan himpunan fuzzy dengan nilai keluaran jernih yaitu 1 NTU sampai dengan 50 NTU, dan himpunan fuzzy lainnya yaitu nilai keluaran keruh dengan nilai keluaran 41 NTU sampai dengan 100 NTU. Berikut adalah gambar himpunan keanggotaan kekeruhan air yang bernilai 1 NTU.

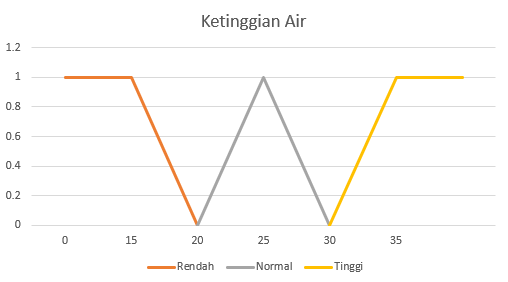
µ Kekeruhan RENDAH[x] =

=

= 0,98

1. Himpunan keanggotaan ketinggian air

Pada fungsi keanggotaan ketinggian air, sistem memberikan nilai input dari sensor berupa data digital tingkat kekeruhan air yang digolongkan menjadi 3 himpunan fuzzy yaitu : rendah ,normal, dan tinggi.



Gambar 4.1.2 Fungsi Keanggotaan Himpunan RENDAH, NORMAL dan TINGGI dari variabel tinggi pada Fuzzy Tsukamoto

Fungsi derajat ketinggian air membagi menjadi 3 himpunan fuzzy dengan himpunan fuzzy dengan nilai keluaran rendah yaitu 0cm sampai dengan 20cm, keanggotaan normal dengan tinggi 20cm sampai dengan 30cm. Dan nilai keanggotaan tinggi dengan nilai diatas 30cm. Berikut adalah perhitungan himpunan keanggotaan ketinggian air yang bernilai 23 Cm.

µ Ketinggian NORMAL[x] =

=

= 0,6

Interference

Pada tahapan inteference melakukan proses penalaran pada fuzzy input dengan fuzzy rules . Pada Tabel 3.2 merupakan aturan – aturan fuzzy yang digunakan pada penelitian ini.

Table 3. 1 Tabel Aturan-aturan Fuzzy

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Kekeruhan** | **Suhu** | **pH** | **Ketinggian** | **Pompa air** |
| 1 | K1 | S1 | P1 | H1 | TB |
| 2 | K1 | S1 | P1 | H2 | TS |
| 3 | K1 | S1 | P1 | H3 | TR |
| 4 | K1 | S2 | P2 | H1 | TR |
| 5 | K1 | S2 | P2 | H2 | TR |
| 6 | K1 | S2 | P2 | H3 | TR |
| 7 | K1 | S3 | P3 | H1 | TB |
| 8 | K1 | S3 | P3 | H2 | TS |
| 9 | K1 | S3 | P3 | H3 | TR |
| 10 | K2 | S1 | P1 | H1 | TB |
| 11 | K2 | S1 | P1 | H2 | TS |
| 12 | K2 | S1 | P1 | H3 | TS |
| 13 | K2 | S2 | P2 | H1 | TB |
| 14 | K2 | S2 | P2 | H2 | TB |
| 15 | K2 | S2 | P2 | H3 | TS |
| 16 | K2 | S3 | P3 | H1 | TB |
| 17 | K2 | S3 | P3 | H2 | TS |
| 18 | K2 | S3 | P3 | H3 | TS |

Table 3. 2 Tabel Aturan-aturan Fuzzy

|  |  |  |
| --- | --- | --- |
| No | Keputusan | Action |
| 1 | K1 | Kekeruhan Jernih |
| 2 | K2 | Kekeruhan Keruh |
| 3 | P1 | pH Rendah |
| 4 | P2 | pH Normal |
| 5 | P3 | pH Tinggi |
| 6 | H1 | Ketinggian Rendah |
| 7 | H2 | Ketinggian Normal |
| 8 | H3 | Ketinggian Tinggi |
| 9 | TR | Tambah Rendah |
| 10 | TS | Tambah Sedang |
| 11 | TB | Tambah Banyak |

Agregasi

Jika nilai kekeruhan (jernih), nilai pH rendah , dan nilai ketinggian Normal , maka “Tambah Air”.

α-predikat R1 = µᴋJERNIH [x] ∩ µpHRENDAH [x] ∩ µᴋiNORMAL [x]

α-predikat R1 = min(µᴋJERNIH, µpHRENDAH, µᴋiNORMAL)

α-predikat R1 = min(0,4;0,98;0,6)

α-predikat R1 = 0,4

Defuzzification

Pada tahap ini merupakan proses pengubahan nilai fuzzy *output* menjadi sebuah crisp value yang diperoleh dengan cara mengambil nilai rata – rata yang terpusat sebagai keputusan yang diberikan oleh sistem. Contoh perhitungan nilai fuzzy *output* menjadi sebuah *crisp value* :

Nilai 1a = α-predikat R1 x z1

= (α2\*(24-20))+20

= 21,6

Nilai 1b = α-predikat R1 x z2

= 28-(α2\*(28-24))

= 26,4

Nilai\_1 = (Nilai\_1a + Nilai\_1b) / 2

= (21,6 + 26,4) / 2

= 24

Maka hasil yang diperoleh yaitu melakukan pengisian selama 24 detik.

### **4.1.2 Deskripsi Sistem**

Pada sistem monitoring ini terfokus pada pemantauan kualitas air terhadap perkembangan ikan nila dan tanaman hidroponik yang ada dengan menggunakan metode Fuzzy Tsukamoto. Pemantauan air tersebut akan menghasilkan sebuah keputusan untuk melakukan sebuah tindakan yang harus dilakukan oleh sistem. Pengguna dapat melakukan pemantauan terhadap kolam aquaponik secara *realtime* kapanpun dan dimanapun melalui *smartphone*.

### **4.1.3 Analisis Pengguna**

Analisis pengguna merupakan analisis terhadap *user* yang terlibat dalam sistem pemantauan dan pengendali sirkulasi air pada aquaponik. *User* dapat melakukan monitoring terhadap kolam aquaponik meliputi ketinggian air, kekeruhan air, dan pH air secara *realtime*, *user* dapat melakukan kontrol terhadap waktu update, dan *user* juga dapat melihat *history* monitoring pada aplikasi *mobile*.

### **4.1.4 Kebutuhan Fungsional**

Kebutuhan Fungsional merupakan fungsionalitas keseluruhan fitur yang tersedia dalam penelitian ini. Fitur yang tersedia pada sitem monitoring ini terdapat dalam Tabel 4.1.7, sebagai berikut :

Tabel 4.1.7 Kebutuhan Fungsional

|  |  |  |
| --- | --- | --- |
| **No.** | **Fitur** | **Keterangan** |
| 1. | Melihat Hasil *Monitoring* Ketinggian | Merupakan informasi mengenai hasil *monitoring* berupa nilai ketinggian air pada kolam aquaponik secara periodik. |
| 2. | Melihat Hasil *Monitoring* Kekeruhan | Merupakan informasi mengenai hasil *monitoring* berupa nilai kekeruhan air pada kolam aquaponik secara periodik. |
| 3. | Melihat Hasil *Monitoring* pH Air | Merupakan informasi mengenai hasil *monitoring* berupa nilai pH air pada kolam aquaponik secara periodik. |
| 4. | Melihat Hasil *Monitoring* Status Pompa | Merupakan informasi mengenai hasil *monitoring* berupa status pompa air pada kolam aquaponik secara periodik. |
| 5. | Melihat Waktu Update Data Terakhir | Merupakan informasi mengenai waktu update data terakhir pada sistem *Monitoring.* |
| 6. | Melihat *History* Monitoring | Melihat beberapa daftar terakhir dari hasil *Monitoring* kolam yang telah dilakukan oleh sistem. |
| 7. | *Refresh* Data Secara R*ealtime* | Melakukan update data *monitoring* secara *realtime* pada aplikasi *mobile*. |
| 8. | Manajemen Waktu Update | Mengatur berapa lama waktu untuk update data secara *realtime*. Nantinya waktu tersebut yang akan digunakan oleh sistem untuk update data secara periodik. |

### **4.1.5 Kebutuhan Non-Fungsional**

Kebutuhan Non-Fungsional merupakan spesifikasi yang dimiliki sistem sebagai kemampuan yang ditawarkan dalam penelitian Sistem Pemantauan dan Pengendali Sirkulasi Air pada Aquaponik terdapat dalam Tabel 4.1.3, sebagai berikut :

Tabel 4.1.3 Kebutuhan Non-Fungsional

|  |  |  |
| --- | --- | --- |
| **No.** | **Jenis** | **Keterangan** |
| 1. | *Usability* | Sistem ini dapat digunakan oleh petani aquaponik kapanpun dan dimanapun melalui platform *mobile*. |
| 2. | *Portability* | Sistem ini dapat digunakan di beberapa device sekaligus. |
| 3. | *Supportability* | Sistem ini membutuhkan koneksi internet dan gadget dalam pengoperasiannya. |
| 4. | *Reliability* | Sistem ini diharapkan dapat memiliki keandalan untuk memonitoring kolam aquaponik berdasarkan hasil yang didapat dari sensor. |

### **4.1.6 Kebutuhan Perangkat Keras (*Hardware*)**

Analisis kebutuhan kebutuhan perangkat keras *(Hardware)* dalam penelitian “Implementasi Sistem Pemantauan dan Pengendali Sirkulasi Air pada Aquaponik dengan Metode Fuzzy Tsukamoto” digunakan dalam pembangunan sistem terdapat dalam Tabel 4.1.5, sebagai berikut :

Tabel 4.1.5 Analisis Kebutuhan Software

|  |  |  |
| --- | --- | --- |
| **No.** | **Perangkat Keras *(Hardware)*** | **Keterangan** |
| 1. | *ESP8266* | Digunakan sebagai mikrokontroller untuk upload data digital dari sensor ke server. |
| 2. | *Arduino Uno* | Digunakan untuk mengendalikan dan memproses data dari perangkan input lalu meneruskannya ke alat *output*. |
| 3. | *Router* | Penghubung ke jaringan internet. |
| 4. | *Kabel Jumper* | Digunakan sebagai penghubung antara mikrokontroller dengan sensor. |
| 5. | *Kabel USB* | Digunakan sebagai kabel penghubung Mikrokontroller dengan daya listrik. |
| 6. | *Sensor Turbidity* | Digunakan sebagai sensor kekeruhan air. |
| 7. | *Sensor pHMeter* | Dignakan sebagai sensor pH air. |
| 8. | *Sensor HCSR04 Ultrasonic* | Digunakan sebagai sensor ketinggian air. |
| 10. | *PWM (Pulse Width Modulation)* | Digunakan untuk mengontrol daya yang berkaitan dengan power supply |
| 11. | *Travo 12volt* | Digunakan sebagai power supply untuk pompa air. |
| 12. | *Kabel Power* | Digunakan sebagai penghubung antara power supply dengan stop kontak. |
| 13. | *Aquarium* | Sebagai media untuk percobaan sistem *monitoring*. |
| 14. | *Pompa Air Celup* | Digunakan sebagai aktuator. |

### **4.1.7 Kebutuhan Perangkat Lunak (*Software*)**

Analisis kebutuhan kebutuhan perangkat lunak *(software)* dalam penelitian “Implementasi Sistem Pemantauan dan Pengendali Sirkulasi Air pada Aquaponik dengan Metode Fuzzy Tsukamoto” digunakan dalam pembangunan sistem terdapat dalam Tabel 4.1.5, sebagai berikut :

Tabel 4.1.5 Analisis Kebutuhan Software

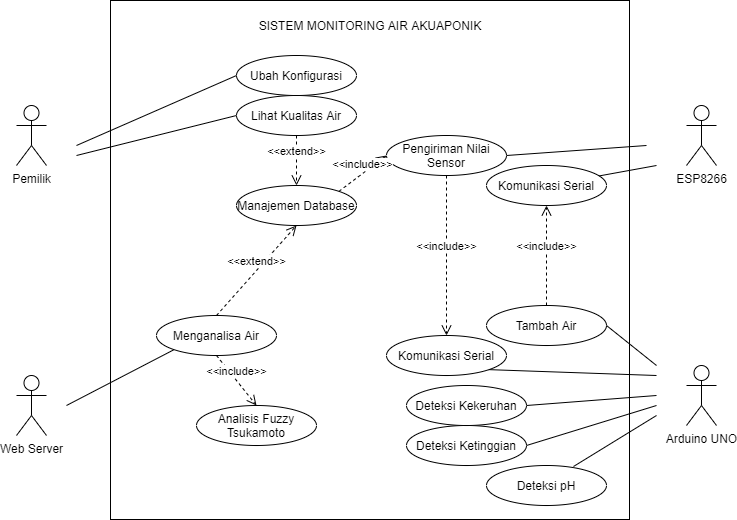
|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Perangkat Lunak *(Software)*** | **Fungsi** | **Keterangan** |
| 1. | *Operating System* | Sistem operasi yang digunakan untuk menjalankan program yang ada pada komputer | *Windows 10* |
| 2. | *Code Editor* | Fasilitas aplikasi yang digunakan untuk menuliskan kode program | *Android Studio, Arduino, Visual Studio Code* |
| 3. | *Database* | Fasilitas penyimpanan data yang digunakan untuk menampung data yang diperlukan | *Firebase Database* |
| 4. | *Internet Network* | Jaringan internet yang berfungsi sebagai akses terhadap sistem yang dibangun | *Indihome* |

## **Perancangan**

Perancangan pada “Implementasi Sistem Pemantauan dan Pengendali Sirkulasi Air pada Aquaponik dengan Metode Fuzzy Tsukamoto” berupa perancangan dalam bentuk diagram, sebagai berikut :

### **4.2.1 Perancangan Usecase Diagram**

Diagram usecase merupakan gambaran atau representasi dari interaksi yang terjadi antara sistem dan lingkungannya. Diagram usecase yang digunakan dalam pembangunan sistem terdapat dalam Tabel 12234234234, sebagai berikut :

****

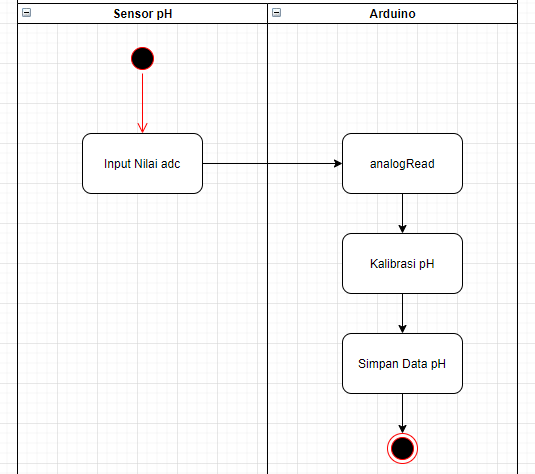
Gambar 4 1 Gambar Usecase

### **4.2.2 Perancangan *Activity Diagram***

Activity diagram menggambarkan aliran fungsionalitas dalam suatu sistem informasi. Secara lengkap, activity diagram mendefinisikan dimana workflow dimulai, dimana berhentinya, aktifitas apa yang terjadi selama workflow, dan bagaimana urutan kejadian aktifitas tersebut (Dewi et al., 2017). Ada beberapa *activity diagram* yang terdapat pada sistem diantaranya :

* + - 1. Deteksi pH

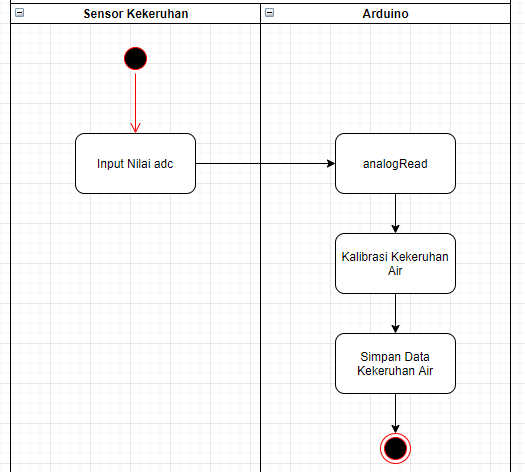
Kalibrasi pH merupakan pengubahan nilai adc (nilai input) yang didapatkan dari arduino menjadi nilai keasaman pH kemudian dikirimkan dan disimpan pada Sistem Monitoring Air.



Gambar 4 2 Activity Diagram Deteksi pH

* + - 1. Deteksi Kekeruhan

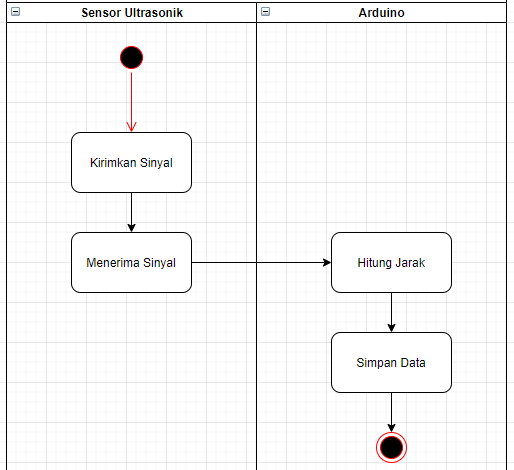
Kalibrasi kekeruhan merupakan pengubahan nilai adc (nilai input) yang didapatkan dari arduino menjadi nilai kekeruhan air kemudian dikirimkan dan disimpan pada Sistem Monitoring Air.



Gambar 4 3 Activity Diagram Deteksi Kekeruhan

* + - 1. Deteksi Ketinggian

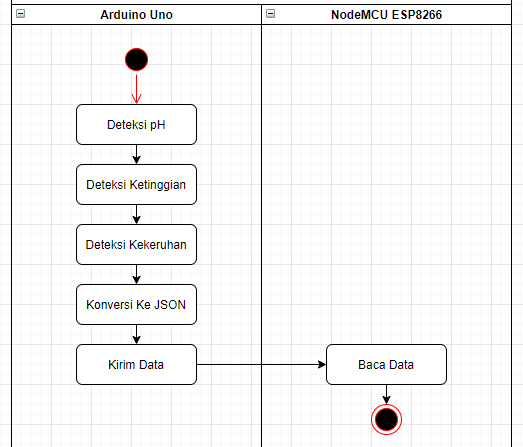
Kalibrasi ketinggian merupakan pengubahan nilai adc (nilai input) yang didapatkan dari arduino menjadi nilai ketinggian air kemudian dikirimkan dan disimpan pada Sistem Monitoring Air.



Gambar 4 4 Activity Diagram Deteksi Ketinggian

* + - 1. Komunikasi Serial Arduino

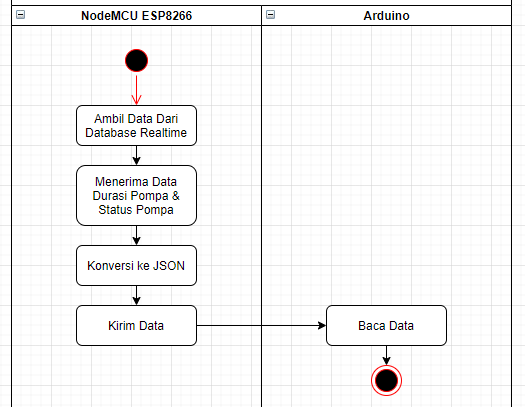
Komunikasi serial adalah komunikasi yang digunakan untuk menghubungkan antara dua mikrokontroller yaitu arduino uno dan NodeMCU ESP8266 dalam melakukan pertukaran data yang dikirimkan pada server dan data response yang didapatkan dari server.



Gambar 4 5 Activity Diagram Komunikasi Serial Arduino

* + - 1. Komunikasi Serial ESP8266

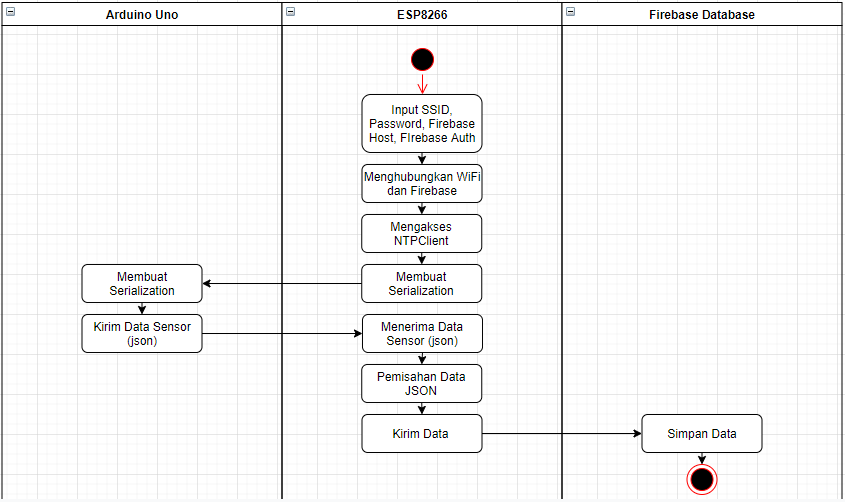
Komunikasi serial adalah komunikasi yang digunakan untuk menghubungkan antara dua mikrokontroller yaitu arduino uno dan NodeMCU ESP8266 dalam melakukan pertukaran data yang dikirimkan pada server dan data response yang didapatkan dari server.



Gambar 4 6 Activity Diagram Komunikasi Serial ESP8266

* + - 1. Pengiriman Nilai Sensor

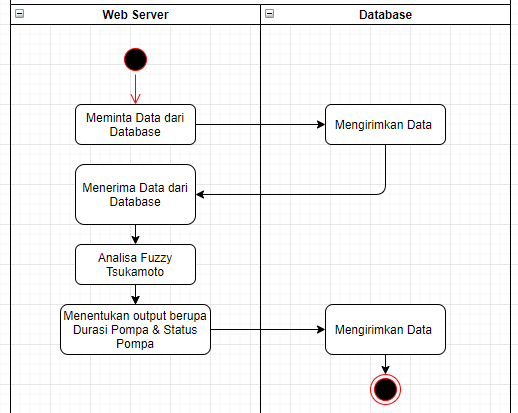
ESP8266 merupakan sebuah perangkat yang digunakan untuk mengirimkan nilai *output* pada arduino ke dalam sistem. NodeMCU ESP8266 mengirimkan semua nilai sensor menggunakan jaringan nirkabel atau Wifi.



Gambar 4 7 Activity Diagram Pengiriman Nilai Sensor

* + - 1. Menganalisa Air

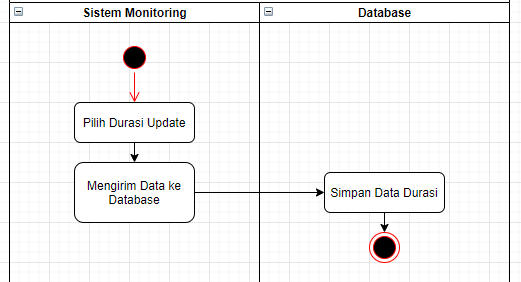
Analisa Fuzzy Tsukamoto merupakan sebuah metode yang digunakan sebagai pengambil keputusan dan memberikan sebuah perintah kepada arduino untuk melakukan penambahan air pada saat kondisi air kurang bersih.



Gambar 4 8 Activity Diagram Menganalisa Air

* + - 1. Manajemen Database

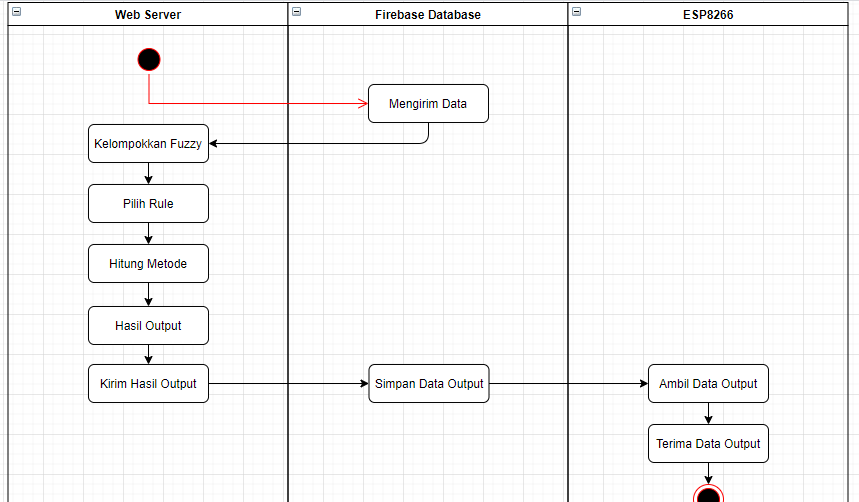
Proses manajemen database adalah proses yang melakukan penginputan dan pengambikan data yang tersimpan pada database.



Gambar 4 9 Activity Diagram Manajemen Database

* + - 1. Analisis Fuzzy Tsukamoto

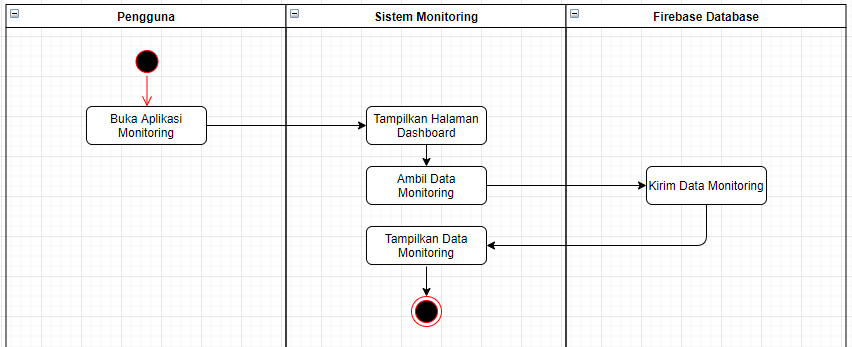
Tahapan – tahapan sistem dalam memberikan keputusan dalam sistem monitoring air.



Gambar 4 10 Activity Diagram Analisis Fuzzy Tsukamoto

* + - 1. Lihat Kualitas Air

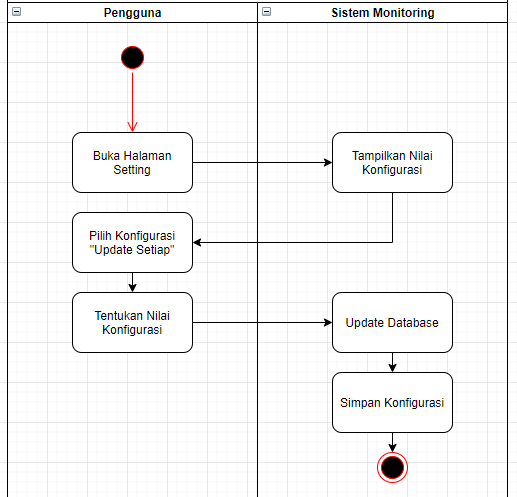
Pengguna dapat melihat kualitas air secara *realtime* meliputi ketinggian air, kekeruhan air, pH air, dan status pompa air.



Gambar 4 11 Activity Diagram Lihat Kualitas Air

* + - 1. Ubah Konfigurasi Sistem Monitoring

Konfigurasi sistem dapat dilakukan oleh pengguna secara dinamis . Sistem Monitoring menyediakan sebuah layanan berupa halaman yang digunakan untuk mengubah nilai konfigurasi yang harus dilakukan oleh sebuah sistem.



Gambar 4 12 Activity Diagram Ubah Konfigurasi Sistem Monitoring

### **4.2.3 Designer Database**

Perancangan desain database digunakan untuk merancang gambaran penyimpanan data. Sistem monitoring menggunakan Firebase Database yang merupakan penyimpanan basis data nonSQL yang memungkinan untuk menyimpan beberapa tipe data. Pada Database memuat beberapa node diantaranya : Sensor, Pompa, dan Settings.

* + - * 1. *Collection* Pompa

|  |
| --- |
| {  "pompa" {  "durasi": float,  "status": int  }  } |

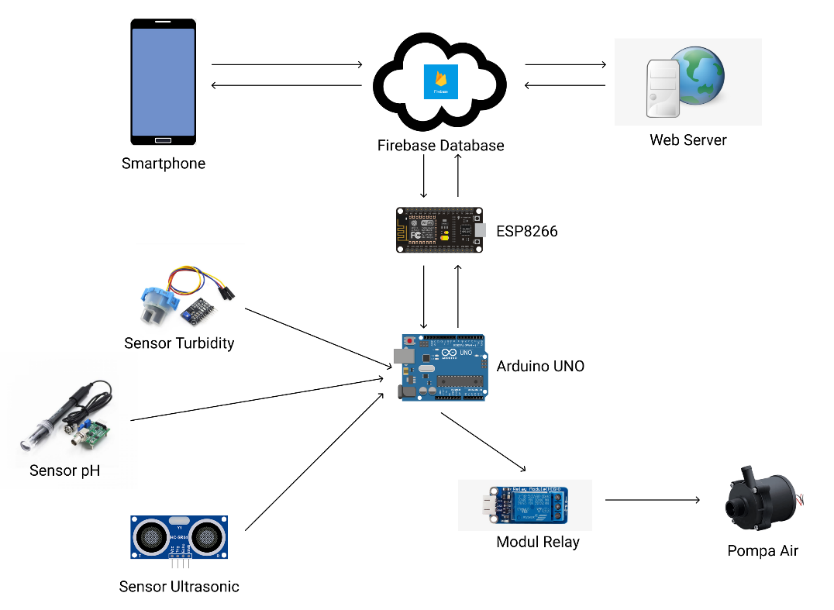
* + - * 1. *Collection* Sensor

|  |
| --- |
| {  "sensor" {  [  {  "kekeruhan": float,  "ketinggian": int,  "pH": float,  "update": String  }  ]  } |

* + - * 1. *Collection* Settings

|  |
| --- |
| {  {  "period": long,  "refresh": int  }  } |

### **4.2.4 Arsitektur Sistem**

****

Gambar 4 13 Arsitektur Sistem

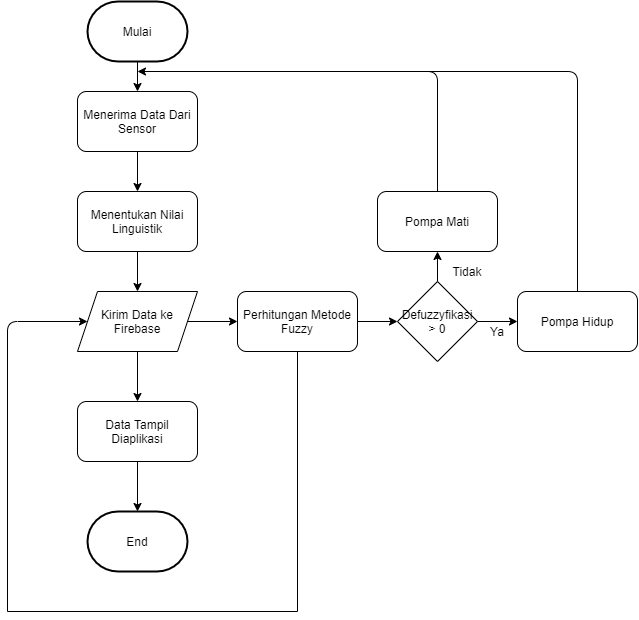
Pengukuran ketinggian air diambil berdasarkan pembacaan sensor ultrasonic, pengukuran pH air diambil berdasarkan sensor pH, dan pengukuran kekeruhan air diambil dari pembacaan sensor turbidity, yang kemudian data tersebut dibaca oleh Arduino Uno, yang kemudian data tersebut akan dikirimkan ke ESP8266 untuk dikirim ke *Firebase Database*.

Data pada *Firebase Database* akan di simpan ke dalam node yang telah ditentukan sebelumnya. Selanjutnya web server akan mengambil data tersebut dan akan dilakukan proses perhitungan menggunakan metode *fuzzy Tsukamoto* dan akan menghasilkan sebuah keputusan yang akan disimpan pada *Firebase Database.*

NodeMCU ESP8266 akan melakukan *get data* atau mengambil data hasil keputusan tersebut untuk dikirimkan ke Arduino UNO. Nantinya Arduino UNO akan menjalankan hasil keputusan tersebut.

### **4.2.5 Flowchart Sistem**

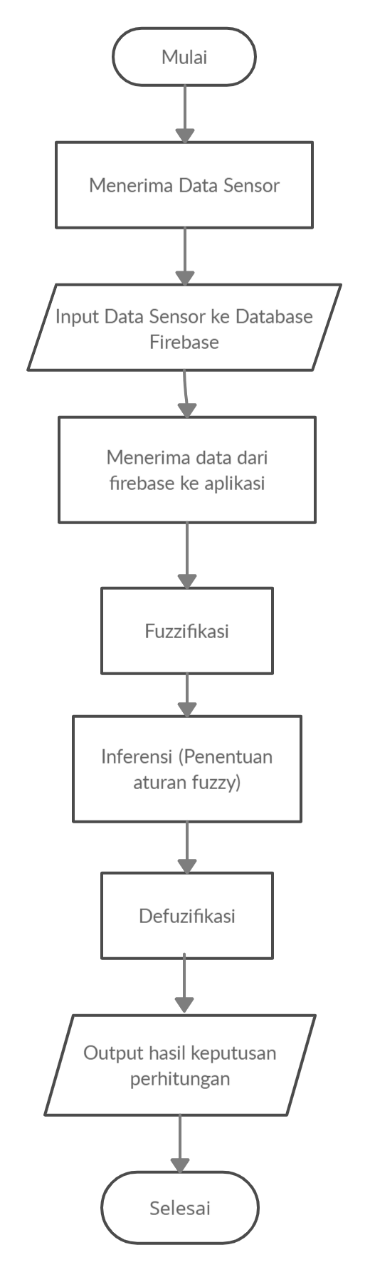
Flowchart merupakan alur urutan proses kerja sistem, pertama sistem akan menerima data dari sensor pH, sensor kekeruhan, dan sensor ketinggian, lalu inputan dari sensor tersebut akan dilakukan penentuan nilai linguistic dan selanjutnya data inputan tersebut akan dikirimkan ke *Firebase Database.* Data yang disimpan pada *Firebase Database* akan ditampilkan di aplikasi *mobile* dan juga akan diambil oleh web server untuk dilakukan perhitungan *fuzzy Tsukamoto.* Selanjutnya hasil perhitungan tersebut akan menghasilkan sebuah keputusan yang akan dijalankan sistem. Jika hasil defuzzyfikasi > 0 maka pompa akan hidup, jika hasil defuzzyfikasi = 0 maka pompa akan mati.



Gambar 4 14 Flowchart Sistem

### **4.2.6 Flowchart Perhitungan Metode**

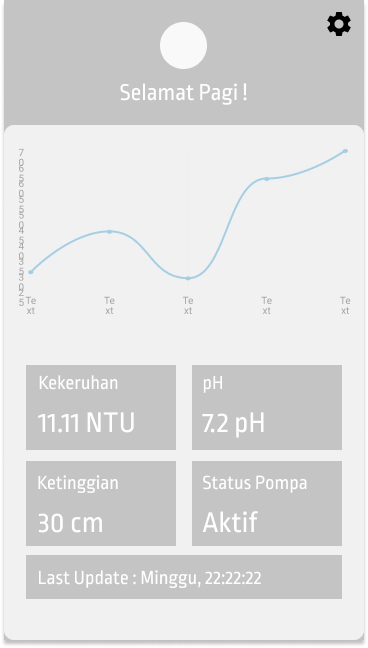
Flowchart Metode merupakan alur urutan proses kerja implementasi perhitungan metode, dimana dalam menentukan hasil *output* sistem menggunakan metode fuzzy tsukamoto, sebagai berikut :



Gambar 4 15 Flowchart Perhitungan Metode

### **4.2.7 Desain Tampilan**

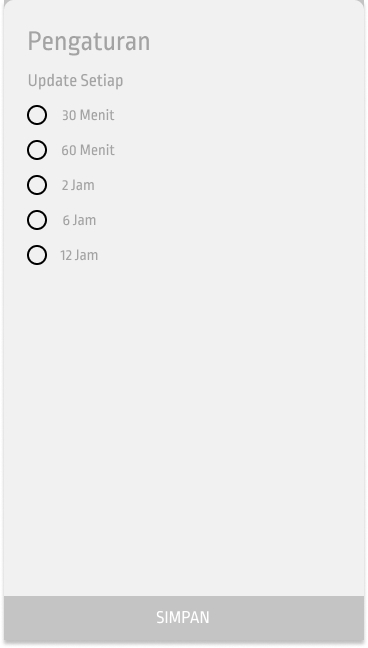
Tampilan grafis yang berhubungan langsung dengan pengguna yang disebut User Interface (UI) dalam sistem direpresentasikan dalam bentuk visual desain mock up mengenai konsep rancangan layout, sebagai berikut :



Gambar 4 16 Desain Tampilan Utama

****

Gambar 4 2 Desain Tampilan History

****

Gambar 4 3 Desain Tampilan Pengaturan

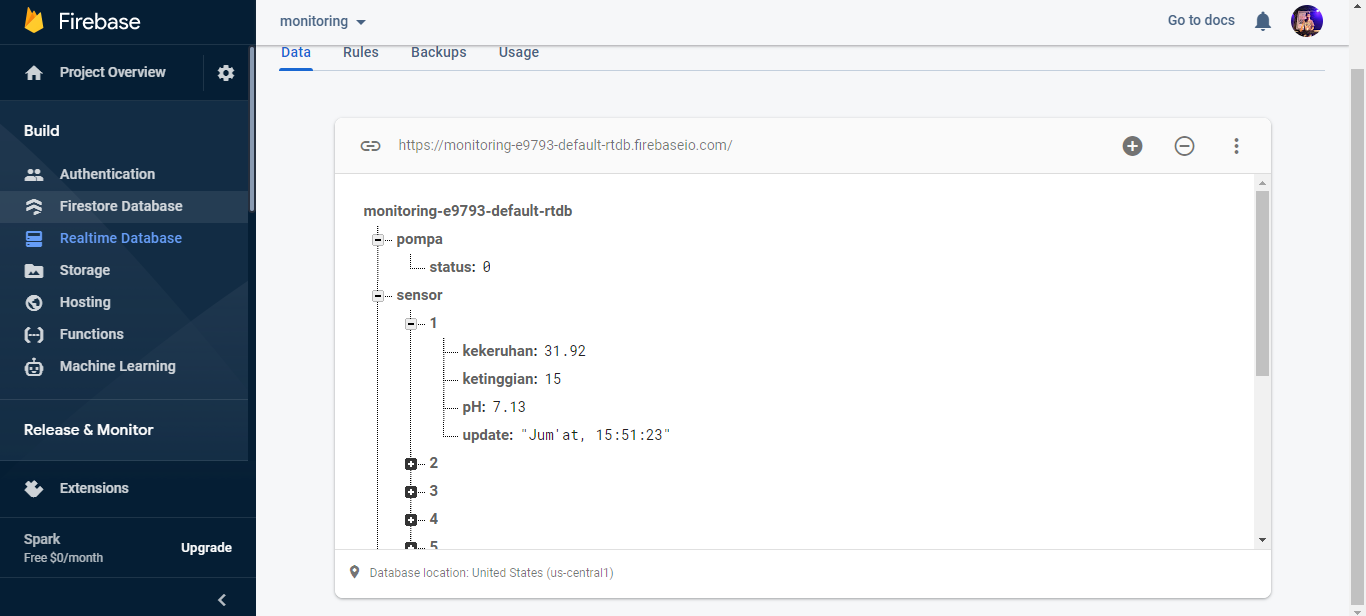
# **BAB V. IMPLEMENTASI DAN PENGUJIAN**

## **Implementasi**

Setelah dilakukan perancangan sistem, maka selanjutnya adalah implementasi sistem sesuai dengan perancangan yang dilakukan. Pada bagian ini menjelaskan tentang hasil dari sistem yang telah dibangun. Implementasi dijelaskan secara detail secara visual dengan tampilan gambar dan potongan kode program atau listing code, sebagai berikut :

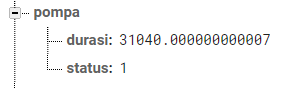
### **5.1.1 Implementasi *Database***

Implementasi database sesuai dengan perancangan menggunakan *Firebase Database* yang digunakan untuk menyimpan data dalam sistem, sebagai berikut :



Gambar 5 1 Tampilan Firebase

Pada Gambar 5.2 merupakan *collection* pompa, yang digunakan untuk menyimpan data durasi dan status yang ditambakan oleh web server.



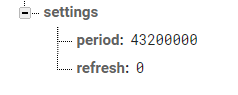
Gambar 5 2 Database Pompa

Pada Gambar 5.3 merupakan *collection* sensor, yang digunakan untuk menyimpan data kekeruhan, ketinggian, pH dan update yang ditambakan oleh NodeMCU.

****

Gambar 5 3 Databse Sensor

Pada Gambar 5.4 merupakan *collection* settings, yang digunakan untuk menyimpan data period dan refresh yang ditambakan oleh aplikasi *mobile*.

****

Gambar 5 4 Database Settings

### **5.1.2 Implementasi Kode Program**

Implementasi beberapa potongan kode program alur proses kerja yang ada dalam sistem sesuai dengan perancangan yang telah dilakukan sebelumnya. Sistem dibuat menggunakan Bahasa pemrograman Java, C, PHP, dan HTML. Implementasi kode program berdasarkan proses analisis dan perancangan dijelaskan secara detail, sebagai berikut :

1. **Aplikasi Android**
   * + 1. MainActivity.java

|  |
| --- |
| public class MainActivity extends AppCompatActivity {  private TextView greetings;  private TextView textKekeruhan;  private TextView textKetinggian;  private TextView textpH;  private TextView textUpdate;  private LineChart lineChart;  private DatabaseReference databaseReference;  private int ketinggian;  private float kekeruhan;  private float pH;  private String update;  private double minKetinggian, maxKetinggian, minKekeruhan, maxKekeruhan, minpH, maxpH;  private SwipeRefreshLayout swipeRefreshLayout;  private ScrollView scrollView;  private RecyclerView recyclerView;  private List<Sensor> sensorList = new ArrayList<>();  private SensorAdapter sensorAdapter;  int index = 0;  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_main);  // bind  greetings = findViewById(R.id.greetings);  textKetinggian = findViewById(R.id.textKetinggian);  textKekeruhan = findViewById(R.id.textKekeruhan);  textpH = findViewById(R.id.textpH);  textUpdate = findViewById(R.id.textUpdate);  swipeRefreshLayout = findViewById(R.id.refresh);  scrollView = findViewById(R.id.scrollView);  recyclerView = findViewById(R.id.recyclerView);  // ucapan selamat berdasarkan waktu.  int currentHour = Calendar.getInstance().get(Calendar.HOUR\_OF\_DAY);  if (currentHour < 10) {  greetings.setText("Selamat Pagi!");  } else if (currentHour < 15) {  greetings.setText("Selamat Siang!");  } else if (currentHour < 18) {  greetings.setText("Selamat Sore!");  } else {  greetings.setText("Selamat Malam!");  }  // Database  databaseReference = FirebaseDatabase.getInstance().getReference();  // last data  Query lastData = databaseReference.child("sensor");  lastData.addValueEventListener(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot dataSnapshot) {  long lengthData = dataSnapshot.getChildrenCount();  long lastChild = lengthData - 1;  try {  ketinggian = Integer.parseInt(dataSnapshot.child(lastChild + "/ketinggian").getValue().toString());  kekeruhan = Float.parseFloat(dataSnapshot.child(lastChild + "/kekeruhan").getValue().toString());  pH = Float.parseFloat(dataSnapshot.child(lastChild + "/pH").getValue().toString());  update = dataSnapshot.child(lastChild + "/update").getValue().toString();  textKetinggian.setText(ketinggian + " cm");  textKekeruhan.setText(kekeruhan + " NTU");  textpH.setText(pH + " pH");  textUpdate.setText("Last update : " +update);  } catch (NullPointerException nullPointer) {  Log.d("Error: ", nullPointer.getMessage());  }  while (index != 10) {  try {  // id = index;  ketinggian = Integer.parseInt(dataSnapshot.child(lastChild + "/ketinggian").getValue().toString());  kekeruhan = Float.parseFloat(dataSnapshot.child(lastChild + "/kekeruhan").getValue().toString());  pH = Float.parseFloat(dataSnapshot.child(lastChild + "/pH").getValue().toString());  update = dataSnapshot.child(lastChild + "/update").getValue().toString();  sensorList.add(new Sensor(ketinggian, kekeruhan, pH, update));  Log.d("aaas", String.valueOf(ketinggian));  Log.d("aaas", String.valueOf(kekeruhan));  Log.d("aaas", String.valueOf(pH));  Log.d("aaas", String.valueOf(update));  index++;  lastChild--;  } catch (NullPointerException nullPointer) {  Log.d("Error: ", nullPointer.getMessage());  }  }  DatabaseReference settings = FirebaseDatabase.getInstance().getReference().child("settings");  settings.addValueEventListener(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot dataSnapshot) {  }  @Override  public void onCancelled(@NonNull DatabaseError databaseError) {  }  });  }  @Override  public void onCancelled(@NonNull DatabaseError databaseError) {  }  });  // refresh  swipeRefreshLayout.setOnRefreshListener(new SwipeRefreshLayout.OnRefreshListener() {  @Override  public void onRefresh() {  swipeRefreshLayout.setRefreshing(true);  databaseReference.child("settings/refresh").setValue(1);  (new Handler()).postDelayed(new Runnable() {  @Override  public void run() {  swipeRefreshLayout.setRefreshing(false);  }  }, 1500);  }  });  //  // scrollview  scrollView.getViewTreeObserver().addOnScrollChangedListener(new ViewTreeObserver.OnScrollChangedListener() {  @Override  public void onScrollChanged() {  int scrollY = scrollView.getScrollY();  if (scrollY == 0) {  swipeRefreshLayout.setEnabled(true);  } else {  swipeRefreshLayout.setEnabled(false);  }  }  });  // set recycler  this.recyclerView = findViewById(R.id.recyclerView);  sensorAdapter = new SensorAdapter(sensorList, this);  Log.d("sensor", String.valueOf(sensorAdapter));  DividerItemDecoration itemDecoration = new DividerItemDecoration(getApplicationContext(), DividerItemDecoration.VERTICAL);  itemDecoration.setDrawable(ContextCompat.getDrawable(getApplicationContext(), R.drawable.line));  recyclerView.addItemDecoration(itemDecoration);  recyclerView.setLayoutManager(new LinearLayoutManager(this));  recyclerView.setAdapter(sensorAdapter);  //chart  lineChart = (LineChart)findViewById(R.id.chart);  LineDataSet lineDataSet = new LineDataSet(getData(), "Pompa");  lineDataSet.setColor(ContextCompat.getColor(this, R.color.colorPrimary));  lineDataSet.setValueTextColor(ContextCompat.getColor(this, R.color.colorPrimaryDark));  XAxis xAxis = lineChart.getXAxis();  xAxis.setPosition(XAxis.XAxisPosition.BOTTOM);  final String[] months = new String[]{"Apr", "May", "Jun", "Jul"};  ValueFormatter formatter = new ValueFormatter() {  @Override  public String getAxisLabel(float value, AxisBase axis) {  return months[(int) value];  }  };  xAxis.setGranularity(1f);  xAxis.setValueFormatter(formatter);  YAxis yAxisRight = lineChart.getAxisRight();  yAxisRight.setEnabled(false);  YAxis yAxisLeft = lineChart.getAxisLeft();  yAxisLeft.setGranularity(1f);  LineData data = new LineData(lineDataSet);  lineChart.setData(data);  lineChart.animateX(2500);  lineChart.invalidate();  }  private ArrayList getData(){  ArrayList<Entry> entries = new ArrayList<>();  entries.add(new Entry(0f, 4f));  entries.add(new Entry(1f, 1f));  entries.add(new Entry(2f, 2f));  entries.add(new Entry(3f, 4f));  return entries;  }  public void handleSettings(View view) {  Intent intent = new Intent(this, SettingsActivity.class);  startActivity(intent);  finish();  }  } |

* + - 1. SettingsActivity.java

|  |
| --- |
| public class SettingsActivity extends AppCompatActivity {  private DatabaseReference databaseReference;  private RadioButton tigaPuluhMenit, satuJam, duaJam, enamJam, duaBelasJam;  private EditText textMinSuhu, textMaxSuhu, textMinKelembaban, textMaxKelembaban;  private long period;  private double minSuhu, maxSuhu, minKelembaban, maxKelembaban;  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_settings);  tigaPuluhMenit = findViewById(R.id.tigaPuluhMenit);  satuJam = findViewById(R.id.satuJam);  duaJam = findViewById(R.id.duaJam);  enamJam = findViewById(R.id.enamJam);  duaBelasJam = findViewById(R.id.duaBelasJam);  databaseReference = FirebaseDatabase.getInstance().getReference();  Query setting = databaseReference.child("settings");  setting.addValueEventListener(new ValueEventListener() {  @Override  public void onDataChange(@NonNull DataSnapshot dataSnapshot) {  period = Long.parseLong(dataSnapshot.child("period").getValue().toString());  if (period == 1800000) {  tigaPuluhMenit.setChecked(true);  } else if (period == 3600000) {  satuJam.setChecked(true);  } else if (period == 7200000) {  duaJam.setChecked(true);  } else if (period == 21600000) {  enamJam.setChecked(true);  } else if (period == 43200000) {  duaBelasJam.setChecked(true);  }  }  @Override  public void onCancelled(@NonNull DatabaseError databaseError) {  }  });  }  @Override  public void onBackPressed() {  super.onBackPressed();  Intent intent = new Intent(this, MainActivity.class);  startActivity(intent);  finish();  }  public void handleSimpan(View view) {  if (tigaPuluhMenit.isChecked()) {  period = 1800000;  } else if (satuJam.isChecked()) {  period = 3600000;  } else if (duaJam.isChecked()) {  period = 7200000;  } else if (enamJam.isChecked()) {  period = 21600000;  } else if (duaBelasJam.isChecked()) {  period = 43200000;  }  DatabaseReference settings = databaseReference.child("settings").child("period");  settings.setValue(period);  new Handler().postDelayed(new Runnable() {  @Override  public void run() {  Intent intent = new Intent(SettingsActivity.this, MainActivity.class);  startActivity(intent);  finish();  }  }, 1000);  Toast.makeText(this, "Pengaturan tersimpan!", Toast.LENGTH\_SHORT).show();  }  } |

* + - 1. SplashActivity.java

|  |
| --- |
| public class SplashActivity extends AppCompatActivity {   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_splash);  new Handler().postDelayed(new Runnable() {  @Override  public void run() {  Intent i = new Intent(SplashActivity.this, MainActivity.class);  startActivity(i);  finish();  }  }, 1000);  } } |

* + - 1. Sensor.java

|  |
| --- |
| public class Sensor {  private int id;  private int ketinggian;  private float kekeruhan;  private float pH;  private String update;   public Sensor(int ketinggian, float kekeruhan, float pH, String update) {   this.ketinggian = ketinggian;  this.kekeruhan = kekeruhan;  this.pH = pH;  this.update = update;  }   public int getKetinggian() {  return ketinggian;  }   public float getKekeruhan() {  return kekeruhan;  }   public float getpH() {  return pH;  }   public String getUpdate() {  return update;  } } |

* + - 1. SensorAdapter.java

|  |
| --- |
| public class SensorAdapter extends RecyclerView.Adapter<SensorAdapter.MyViewHolder> {   List<Sensor> sensorList;  private Context context;   public SensorAdapter(List<Sensor> sensorList, Context context) {  this.sensorList = sensorList;  this.context = context;  }   @NonNull  @Override  public SensorAdapter.MyViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {  LayoutInflater layoutInflater = LayoutInflater.from(context);  View menuView = layoutInflater.inflate(R.layout.sensor\_item, parent, false);  MyViewHolder viewHolder = new MyViewHolder(menuView);  return viewHolder;  }   @Override  public void onBindViewHolder(@NonNull SensorAdapter.MyViewHolder holder, int position) {  Sensor sensor = sensorList.get(position);  holder.ketinggian.setText(String.valueOf(sensor.getKetinggian()));  holder.kekeruhan.setText(String.valueOf(sensor.getKekeruhan()));  holder.pH.setText(String.valueOf(sensor.getpH()));  holder.update.setText(String.valueOf(sensor.getUpdate()));  }   @Override  public int getItemCount() {  return sensorList.size();  }   public class MyViewHolder extends RecyclerView.ViewHolder {  public TextView ketinggian, kekeruhan, pH, update;   public MyViewHolder(@NonNull View itemView) {  super(itemView);  this.ketinggian = itemView.findViewById(R.id.ketinggian);  this.kekeruhan = itemView.findViewById(R.id.kekeruhan);  this.pH = itemView.findViewById(R.id.pH);  this.update = itemView.findViewById(R.id.update);  }  } } |

* + - 1. activity\_main.xml

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <androidx.swiperefreshlayout.widget.SwipeRefreshLayout xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:id="@+id/refresh"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  xmlns:android="http://schemas.android.com/apk/res/android">  <androidx.constraintlayout.widget.ConstraintLayout  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:background="@color/colorPrimaryDark"  tools:context=".MainActivity">   <ImageView  android:id="@+id/avatar"  android:layout\_width="50dp"  android:layout\_height="50dp"  android:layout\_marginTop="20dp"  android:src="@drawable/logo"  app:layout\_constraintTop\_toTopOf="parent"  app:layout\_constraintBottom\_toTopOf="@id/greetings"  app:layout\_constraintLeft\_toLeftOf="parent"  app:layout\_constraintRight\_toRightOf="parent"/>   <TextView  android:id="@+id/greetings"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginTop="8dp"  android:layout\_marginBottom="20dp"  android:text="Selamat Datang!"  android:textSize="20sp"  android:textColor="@color/colorWhite"  app:layout\_constraintTop\_toBottomOf="@id/avatar"  app:layout\_constraintBottom\_toTopOf="@id/scrollView"  app:layout\_constraintLeft\_toLeftOf="parent"  app:layout\_constraintRight\_toRightOf="parent"/>   <ImageButton  android:id="@+id/settings"  android:layout\_width="25dp"  android:layout\_height="25dp"  android:background="@drawable/settings"  android:layout\_marginTop="15dp"  android:layout\_marginRight="15dp"  app:layout\_constraintRight\_toRightOf="parent"  app:layout\_constraintTop\_toTopOf="parent"  android:onClick="handleSettings"/>   <ScrollView  android:id="@+id/scrollView"  android:layout\_width="match\_parent"  android:layout\_height="0dp"  android:background="@drawable/shape\_rounded\_white"  app:layout\_constraintTop\_toBottomOf="@id/greetings"  app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintLeft\_toLeftOf="parent"  app:layout\_constraintRight\_toRightOf="parent">   <LinearLayout  android:id="@+id/main"  android:layout\_width="match\_parent"  android:layout\_height="0dp"  android:orientation="vertical">   <com.github.mikephil.charting.charts.LineChart  android:id="@+id/chart"  android:layout\_width="match\_parent"  android:layout\_height="250dp"  android:layout\_margin="8dp"/>   <LinearLayout  android:id="@+id/kekeruhanpH"  android:baselineAligned="false"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:weightSum="2"  app:layout\_constraintTop\_toTopOf="parent">   <LinearLayout  android:id="@+id/forKekeruhan"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginBottom="8dp"  android:layout\_marginRight="4dp"  android:layout\_marginLeft="8dp"  android:padding="16dp"  android:background="@drawable/shape\_rounded\_blue"  android:orientation="vertical"  android:layout\_weight="1">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Kekeruhan"  android:textColor="@color/colorWhite"  android:textSize="16sp"/>   <TextView  android:id="@+id/textKekeruhan"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="0 NTU"  android:textColor="@color/colorWhite"  android:textSize="30sp"/>  </LinearLayout>   <LinearLayout  android:id="@+id/forpH"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginBottom="8dp"  android:layout\_marginRight="8dp"  android:layout\_marginLeft="4dp"  android:padding="16dp"  android:background="@drawable/shape\_rounded\_blue"  android:orientation="vertical"  android:layout\_weight="1">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="pH"  android:textColor="@color/colorWhite"  android:textSize="16sp"/>   <TextView  android:id="@+id/textpH"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="0 pH"  android:textColor="@color/colorWhite"  android:textSize="30sp"/>  </LinearLayout>  </LinearLayout>   <LinearLayout  android:id="@+id/ketinggianSuhu"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:weightSum="2"  app:layout\_constraintTop\_toTopOf="parent">   <LinearLayout  android:id="@+id/forKetinggian"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginBottom="8dp"  android:layout\_marginRight="4dp"  android:layout\_marginLeft="8dp"  android:padding="16dp"  android:background="@drawable/shape\_rounded\_blue"  android:orientation="vertical"  android:layout\_weight="1">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Ketinggian Air"  android:textColor="@color/colorWhite"  android:textSize="16sp"/>   <TextView  android:id="@+id/textKetinggian"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="0 cm"  android:textColor="@color/colorWhite"  android:textSize="30sp"/>  </LinearLayout>   <LinearLayout  android:id="@+id/forPompa"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginBottom="8dp"  android:layout\_marginRight="8dp"  android:layout\_marginLeft="4dp"  android:padding="16dp"  android:background="@drawable/shape\_rounded\_blue"  android:orientation="vertical"  android:layout\_weight="1">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Status Pompa"  android:textColor="@color/colorWhite"  android:textSize="16sp"/>   <TextView  android:id="@+id/textKelembaban"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Aktif"  android:textColor="@color/colorWhite"  android:textSize="30sp"/>  </LinearLayout>  </LinearLayout>   <LinearLayout  android:id="@+id/forUpdate"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginTop="0dp"  android:layout\_marginBottom="8dp"  android:layout\_marginRight="8dp"  android:layout\_marginLeft="8dp"  android:padding="16dp"  android:background="@drawable/shape\_rounded\_blue"  android:orientation="vertical"  android:layout\_weight="1">   <TextView  android:id="@+id/textUpdate"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Last Update : "  android:textColor="@color/colorWhite"  android:textSize="16sp"/>  </LinearLayout>   <TextView  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:textAlignment="center"  android:textColor="@color/colorBlack"  android:text="History"  android:textSize="18sp"  android:layout\_margin="8dp"  android:gravity="center\_horizontal" />   <androidx.recyclerview.widget.RecyclerView  android:id="@+id/recyclerView"  android:layout\_width="match\_parent"  android:layout\_height="600dp"  android:layout\_margin="8dp"/>  </LinearLayout>  </ScrollView>  </androidx.constraintlayout.widget.ConstraintLayout> </androidx.swiperefreshlayout.widget.SwipeRefreshLayout> |

* + - 1. activity\_settings.xml

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:background="@color/colorWhite"  tools:context=".SettingsActivity">   <androidx.constraintlayout.widget.Guideline  android:id="@+id/guideline"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:orientation="vertical"  app:layout\_constraintGuide\_percent="0.3"/>   <TextView  android:id="@+id/textPengaturan"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Pengaturan"  android:textColor="@color/colorBlack"  android:textSize="30sp"  android:textStyle="bold"  android:layout\_marginTop="15dp"  android:layout\_marginLeft="15dp"  app:layout\_constraintTop\_toTopOf="parent"  app:layout\_constraintLeft\_toLeftOf="parent"/>   <TextView  android:id="@+id/textUpdate"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginTop="15dp"  android:layout\_marginLeft="15dp"  android:text="Update setiap:"  android:textStyle="bold"  android:textColor="@color/colorBlack"  app:layout\_constraintTop\_toBottomOf="@id/textPengaturan"  app:layout\_constraintLeft\_toLeftOf="parent"/>   <RadioGroup  android:id="@+id/radioGroup"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_margin="15dp"  app:layout\_constraintTop\_toBottomOf="@id/textUpdate"  app:layout\_constraintLeft\_toLeftOf="parent">   <RadioButton  android:id="@+id/tigaPuluhMenit"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:text="30 Menit"  android:textColor="@color/colorBlack"/>   <RadioButton  android:id="@+id/satuJam"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:text="1 Jam"  android:textColor="@color/colorBlack"/>   <RadioButton  android:id="@+id/duaJam"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:text="2 Jam"  android:textColor="@color/colorBlack"/>   <RadioButton  android:id="@+id/enamJam"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:text="6 Jam"  android:textColor="@color/colorBlack"/>   <RadioButton  android:id="@+id/duaBelasJam"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:text="12 Jam"  android:textColor="@color/colorBlack"/>   </RadioGroup>   <Button  android:id="@+id/buttonSimpan"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:background="@color/colorPrimary"  android:text="Simpan"  android:textColor="@color/colorWhite"  app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintLeft\_toLeftOf="parent"  app:layout\_constraintRight\_toRightOf="parent"  android:onClick="handleSimpan"/> </androidx.constraintlayout.widget.ConstraintLayout> |

* + - 1. activity\_splash.xml

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:orientation="vertical"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:background="@color/colorWhite">   <ImageView  android:id="@+id/logo"  android:layout\_width="100dp"  android:layout\_height="100dp"  android:layout\_marginTop="250dp"  android:src="@drawable/logo"  app:layout\_constraintTop\_toTopOf="parent"  app:layout\_constraintBottom\_toTopOf="@id/textbawang"  app:layout\_constraintLeft\_toLeftOf="parent"  app:layout\_constraintRight\_toRightOf="parent"/>   <TextView  android:id="@+id/textbawang"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="MONITORING KOLAM AQUAPONIK"  android:textColor="@color/colorGrey"  android:textSize="20sp"  android:layout\_marginBottom="300dp"  app:layout\_constraintTop\_toBottomOf="@id/logo"  app:layout\_constraintLeft\_toLeftOf="parent"  app:layout\_constraintRight\_toRightOf="parent"  app:layout\_constraintBottom\_toBottomOf="parent" /> </androidx.constraintlayout.widget.ConstraintLayout> |

* + - 1. sensor\_item.xml

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_margin="8dp"  android:orientation="vertical">   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:baselineAligned="false"  android:orientation="horizontal"  android:weightSum="2">   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_weight="1"  android:orientation="horizontal">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Ketinggian: "  android:textSize="15sp" />   <TextView  android:id="@+id/ketinggian"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="100"  android:textColor="@color/colorBlack"  android:textSize="15sp" />   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text=" cm"  android:textColor="@color/colorBlack"  android:textSize="15sp" />  </LinearLayout>   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_weight="1">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Kekeruhan: "  android:textSize="15sp" />   <TextView  android:id="@+id/kekeruhan"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="100"  android:textColor="@color/colorBlack"  android:textSize="15sp" />   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text=" NTU"  android:textColor="@color/colorBlack"  android:textSize="15sp" />  </LinearLayout>  </LinearLayout>   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:orientation="horizontal"  android:weightSum="2">   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:baselineAligned="false"  android:orientation="horizontal"  android:weightSum="2">   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_weight="1"  android:orientation="horizontal">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="pH: "  android:textSize="15sp" />   <TextView  android:id="@+id/pH"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="100"  android:textColor="@color/colorBlack"  android:textSize="15sp" />   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text=" pH"  android:textColor="@color/colorBlack"  android:textSize="15sp" />  </LinearLayout>   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_weight="1">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Update: "  android:textSize="15sp" />   <TextView  android:id="@+id/update"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Senin, 00:00:00"  android:textColor="@color/colorBlack"  android:textSize="15sp" />  </LinearLayout>  </LinearLayout>  </LinearLayout> </LinearLayout> |

1. **Arduino Uno**

|  |
| --- |
| #include <SoftwareSerial.h>  #include <ArduinoJson.h>  SoftwareSerial linkSerial(2, 3); //RX,TX  const int trig = 11; // membuat varibel trig yang di set ke-pin 3  const int echo = A0; // membuat variabel echo yang di set ke-pin 2  long durasi\_ultrasonic, jarak; // membuat variabel durasi dan jarak  //pompa  const int ena = 9;  const int in1 = 5;  //milliskirim  unsigned long interval; // the time we need to wait  unsigned long previousMillis = 0; // millis() returns an unsigned long.  //kekeruhan  static float kekeruhan;  static float tegangan;  //ph  const int ph\_Pin = A2;  float Po = 0;  float PH\_step;  int nilai\_analog\_PH;  double TeganganPh;  //untuk kalibrasi  float PH4 = 3.226;  float PH7 = 2.691;  //*output*pompa  int status\_pompa;  int durasi;  void setup() {  Serial.begin(9600);  linkSerial.begin(9600);  pinMode(trig, OUTPUT); // set pin trig menjadi OUTPUT  pinMode(echo, INPUT); // set pin echo menjadi INPUT  pinMode (ph\_Pin, INPUT);  pinMode(ena, OUTPUT);  pinMode(in1, OUTPUT);  }  void loop() {  digitalWrite(trig, LOW);  delayMicroseconds(8);  digitalWrite(trig, HIGH);  delayMicroseconds(8);  digitalWrite(trig, LOW);  delayMicroseconds(8);  durasi\_ultrasonic = pulseIn(echo, HIGH); // menerima suara ultrasonic  jarak = (durasi\_ultrasonic / 2) / 29.1; // mengubah durasi menjadi jarak (cm)  int val = analogRead(A1);  tegangan = val \* (5.0 / 1024);  kekeruhan = 100.00 - (tegangan / 4.16) \* 100.00;  //ph  nilai\_analog\_PH = analogRead(ph\_Pin);  TeganganPh = 3.3 / 1024.0 \* nilai\_analog\_PH;  PH\_step = (PH4 - PH7) / 3;  Po = 7.00 + ((PH7 - TeganganPh) / PH\_step); //Po = 7.00 + ((teganganPh7 - TeganganPh) / PhStep);  if (jarak != 0 && kekeruhan != 0 && Po != 0)  {  //Kirim data  StaticJsonBuffer<1000> jsonBuffer;  JsonObject& data = jsonBuffer.createObject();  data["jarak"] = jarak;  data["kekeruhan"] = kekeruhan;  data["ph"] = Po;  //Send data to NodeMCU  data.printTo(linkSerial);  jsonBuffer.clear();  }  else  {  Serial.println("Input Sensor Invalid!");  }  StaticJsonBuffer<1000> jsonBuffer;  JsonObject& hasil = jsonBuffer.parseObject(linkSerial);  if (hasil == JsonObject::invalid()) {  Serial.println("Invalid Json Object");  jsonBuffer.clear();  return;  }  Serial.println("JSON Object Recieved");  Serial.print("Pompa : ");  status\_pompa = hasil["status\_pompa"];  Serial.println(status\_pompa);  Serial.print("Durasi : ");  durasi = hasil["durasi"];  Serial.println(durasi);  Serial.println("-----------------------------------------");  if (status\_pompa == 1) {  digitalWrite(ena, HIGH);  digitalWrite(in1, HIGH);  }  else if (status\_pompa == 0) {  digitalWrite(ena, LOW);  digitalWrite(in1, LOW);  }  else {  Serial.println("Pompa Bermasalah");  }  delay(1000);  } |

1. **ESP8266**

|  |
| --- |
| //Wifi  #include <FirebaseArduino.h>  #include <ArduinoJson.h>  #include <ESP8266WiFi.h>  #include <NTPClient.h>  #include <WiFiUdp.h>  #include <SoftwareSerial.h>  SoftwareSerial linkSerial(D2, D1); //RX,TX  #define FIREBASE\_HOST "monitoring-e9793-default-rtdb.firebaseio.com"  #define FIREBASE\_AUTH "6JoSO8ti008rSFc94Lin17CoabXxMmgAmVOQDbaw"  #define WIFI\_SSID "AFIF HOME"  #define WIFI\_PASSWORD "afif5758"  int ketinggian;  float kekeruhan;  float ph;  String waktu;  int period;  int refresh = 0;  String durasi;  int status\_pompa;  unsigned long interval; // the time we need to wait  unsigned long previousMillis = 0; // millis() returns an unsigned long.  //waktu  const long utcOffsetInSeconds = 25200;  char daysOfTheWeek[7][12] = {"Minggu", "Senin", "Selasa", "Rabu", "Kamis", "Jum'at", "Sabtu"};  // Define NTP Client to get time  WiFiUDP ntpUDP;  NTPClient timeClient(ntpUDP, "id.pool.ntp.org", utcOffsetInSeconds);  void setup() {  Serial.begin(9600);  linkSerial.begin(9600);  timeClient.begin();  WiFi.begin(WIFI\_SSID, WIFI\_PASSWORD);  Serial.print("connecting");  while (WiFi.status() != WL\_CONNECTED) {  Serial.print(".");  delay(500);  }  Serial.println();  Serial.print("connected: ");  Serial.println(WiFi.localIP());  Firebase.begin(FIREBASE\_HOST, FIREBASE\_AUTH);  }  void loop() {  StaticJsonBuffer<1000> jsonBuffer;  JsonObject& data = jsonBuffer.parseObject(linkSerial);  if (data == JsonObject::invalid()) {  Serial.println("Invalid Json Object");  jsonBuffer.clear();  return;  }  if (data["jarak"] != 0 && data["kekeruhan"] != 0 && data["ph"] != 0) {  Serial.println("JSON Object Recieved");  Serial.print("Jarak : ");  ketinggian = data["jarak"];  Serial.println(ketinggian);  Serial.print("Kekeruhan : ");  float k = data["kekeruhan"];  String strK(k, 2);  kekeruhan = strK.toFloat();  Serial.println(kekeruhan);  Serial.print("PH : ");  float p = data["ph"];  String strP(p, 2);  ph = strP.toFloat();  Serial.println(ph);  Serial.println("-----------------------------------------");  }  //waktu  timeClient.update();  String hari = (daysOfTheWeek[timeClient.getDay()]);  int jam = (timeClient.getHours());  int menit = (timeClient.getMinutes());  int detik = (timeClient.getSeconds());  String waktu = (hari + ", " + jam + ":" + menit + ":" + detik);  refresh = Firebase.getInt("settings/refresh");  if (refresh == 1) {  Firebase.setInt(String("sensor/" + String(id) + "/ketinggian"), ketinggian);  Firebase.setFloat(String("sensor/" + String(id) + "/kekeruhan"), kekeruhan);  Firebase.setFloat(String("sensor/" + String(id) + "/pH"), ph);  Firebase.setString(String("sensor/" + String(id) + "/update"), waktu);  id++;  if (Firebase.failed()) {  return;  }  refresh = 0;  Firebase.setInt("settings/refresh", refresh);  // Serial.println("refresh");  }  period = Firebase.getInt("settings/period");  Serial.println(period);  if (period == 1800000) {  interval = period;  unsigned long currentMillis = millis(); // grab current time  if ((unsigned long)(currentMillis - previousMillis) >= interval) {  Firebase.setInt(String("sensor/" + String(id) + "/ketinggian"), ketinggian);  Firebase.setFloat(String("sensor/" + String(id) + "/kekeruhan"), kekeruhan);  Firebase.setFloat(String("sensor/" + String(id) + "/pH"), ph);  Firebase.setString(String("sensor/" + String(id) + "/update"), waktu);  id++;  previousMillis = millis();  }  }  else if (period == 3600000) {  interval = period;  unsigned long currentMillis = millis(); // grab current time  if ((unsigned long)(currentMillis - previousMillis) >= interval) {  Firebase.setInt(String("sensor/" + String(id) + "/ketinggian"), ketinggian);  Firebase.setFloat(String("sensor/" + String(id) + "/kekeruhan"), kekeruhan);  Firebase.setFloat(String("sensor/" + String(id) + "/pH"), ph);  Firebase.setString(String("sensor/" + String(id) + "/update"), waktu);  id++;  previousMillis = millis();  }  }  else if (period == 7200000) {  interval = period;  unsigned long currentMillis = millis();  if ((unsigned long)(currentMillis - previousMillis) >= interval) {  Firebase.setInt(String("sensor/" + String(id) + "/ketinggian"), ketinggian);  Firebase.setFloat(String("sensor/" + String(id) + "/kekeruhan"), kekeruhan);  Firebase.setFloat(String("sensor/" + String(id) + "/pH"), ph);  Firebase.setString(String("sensor/" + String(id) + "/update"), waktu);  id++;  previousMillis = millis();  }  }  else if (period == 21600000) {  interval = period;  unsigned long currentMillis = millis(); // grab current time  if ((unsigned long)(currentMillis - previousMillis) >= interval) {  Firebase.setInt(String("sensor/" + String(id) + "/ketinggian"), ketinggian);  Firebase.setFloat(String("sensor/" + String(id) + "/kekeruhan"), kekeruhan);  Firebase.setFloat(String("sensor/" + String(id) + "/pH"), ph);  Firebase.setString(String("sensor/" + String(id) + "/update"), waktu);  id++;  previousMillis = millis();  }  }  else if (period == 43200000) {  interval = period;  unsigned long currentMillis = millis(); // grab current time  if ((unsigned long)(currentMillis - previousMillis) >= interval) {  Firebase.setInt(String("sensor/" + String(id) + "/ketinggian"), ketinggian);  Firebase.setFloat(String("sensor/" + String(id) + "/kekeruhan"), kekeruhan);  Firebase.setFloat(String("sensor/" + String(id) + "/pH"), ph);  Firebase.setString(String("sensor/" + String(id) + "/update"), waktu);  id++;  previousMillis = millis();  }  }  status\_pompa = Firebase.getInt("pompa/status");  durasi = Firebase.getInt("pompa/durasi");  Serial.println(status\_pompa);  Serial.println(durasi);  JsonObject& hasil = jsonBuffer.createObject();  hasil["status\_pompa"] = status\_pompa;  hasil["durasi"] = durasi;  hasil.printTo(linkSerial);  jsonBuffer.clear();  } |

1. **Web**

|  |
| --- |
| <!DOCTYPE html>  <html lang="en">  <head>      <meta charset="UTF-8">      <meta http-equiv="X-UA-Compatible" content="IE=edge">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>Document</title>      <script src="https://www.gstatic.com/firebasejs/8.6.8/firebase-app.js"></script>      <script src="https://www.gstatic.com/firebasejs/8.6.8/firebase-database.js"></script>      <script src="//ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.min.js"></script>    </head>  <body onload = "JavaScript:AutoRefresh(60000);">      Ketinggian : <ketinggian></ketinggian><br>      Kekeruhan : <kekeruhan></kekeruhan><br>      pH : <pH></pH><br>      Update : <update></update><br><br>      Hasil : <hasil></hasil><br>      <br>      <script type="text/javascript">      //Refresh      function AutoRefresh( t ) {          setTimeout("location.reload(true);", t);      }      // Your web app's Firebase configuration      var firebaseConfig = {          apiKey: "AIzaSyA5O30h-KsmkFPg0qaYw7eQAJi\_Q\_baKQM",          authDomain: "monitoring-e9793.firebaseapp.com",          databaseURL: "https://monitoring-e9793-default-rtdb.firebaseio.com",          projectId: "monitoring-e9793",          storageBucket: "monitoring-e9793.appspot.com",          messagingSenderId: "131986672780",          appId: "1:131986672780:web:7e65548439176a5be43fa9"      };      // Initialize Firebase      firebase.initializeApp(firebaseConfig);      var firebaseRef = firebase.database().ref("sensor");      firebaseRef.limitToLast(1).on("value", function(snapshot){          var data = snapshot.val();            var \_ketinggian = document.querySelector('ketinggian');          var \_kekeruhan = document.querySelector('kekeruhan');          var \_pH = document.querySelector('pH');          var \_update = document.querySelector('update');          var \_hasil = document.querySelector('hasil');          for(let i in data) {              console.log(data [i]);              \_ketinggian.innerHTML = data [i].ketinggian;              \_kekeruhan.innerHTML = data [i].kekeruhan;              \_pH.innerHTML = data [i].pH;              \_update.innerHTML = data [i].update;               dataKetinggian = parseInt(data [i].ketinggian);               $(document).ready(function () {                  createCookie("ket", dataKetinggian, "10");              });              function createCookie(name, value, days) {                  var expires;                    if (days) {                      var date = new Date();                      date.setTime(date.getTime() + (days \* 24 \* 60 \* 60 \* 1000));                      expires = "; expires=" + date.toGMTString();                  }                  else {                      expires = "";                  }                    document.cookie = escape(name) + "=" +                      escape(value) + expires + "; path=/";              }              console.log("Ketinggian: " +dataKetinggian);              // Creating a cookie after the document is ready              dataKekeruhan = parseFloat(data [i].kekeruhan);              $(document).ready(function () {                  createCookie("kek", dataKekeruhan, "10");              });              function createCookie(name, value, days) {                  var expires;                  if (days) {                      var date = new Date();                      date.setTime(date.getTime() + (days \* 24 \* 60 \* 60 \* 1000));                      expires = "; expires=" + date.toGMTString();                  }                  else {                      expires = "";                  }                  document.cookie = escape(name) + "=" +                      escape(value) + expires + "; path=/";              }              console.log("Kekeruhan: " +dataKekeruhan);              // Creating a cookie after the document is ready              dataPH = parseFloat(data [i].pH);              $(document).ready(function () {                  createCookie("ph", dataPH, "10");              });              function createCookie(name, value, days) {                  var expires;                    if (days) {                      var date = new Date();                      date.setTime(date.getTime() + (days \* 24 \* 60 \* 60 \* 1000));                      expires = "; expires=" + date.toGMTString();                  }                  else {                      expires = "";                  }                    document.cookie = escape(name) + "=" +                      escape(value) + expires + "; path=/";              }              console.log("pH: " +dataPH);          }          //cob      });      </script>  <?php  $ketinggian = $\_COOKIE["ket"];  $kekeruhan = $\_COOKIE["kek"];  $ph = $\_COOKIE["ph"];  $alfa = array();  $zn = 0;  $zt = 0;  $total = 0;  $z = array();  $hasil = 0;  echo $ketinggian;  ?> <br> <?php  echo $kekeruhan;  ?> <br> <?php  echo $ph;    if($ketinggian != 0 && $kekeruhan !=0 && $ph !=0){      $hasil = perhitungan($ketinggian, $kekeruhan, $ph);  }  function findMin($x, $y, $z){      if($x <= $y && $x <= $z){          return $x;      } elseif($y <= $x && $y <= $z){          return $y;      } else{          return $z;      }  }  function ketRendah($ketinggian){      if($ketinggian <= 15){          return 1;      } elseif ($ketinggian > 15 && $ketinggian < 20){          return (20 - $ketinggian) / (20-15);      } else{          return 0;      }  }  function ketNormal($ketinggian){      if($ketinggian > 20 && $ketinggian <= 25){          return ($ketinggian - 20) / (25-20);      } elseif($ketinggian > 25 && $ketinggian < 30){          return (25 - $ketinggian) / (30-25);      } else {          return 0;      }  }  function ketTinggi($ketinggian){      if($ketinggian <= 30){          return 0;      } elseif ($ketinggian > 30 && $ketinggian < 35){          return ($ketinggian - 30) / (35-30);      } else{          return 1;      }  }  function kekJernih($kekeruhan){      if($kekeruhan <= 0){          return 1;      } elseif ($kekeruhan > 0 && $kekeruhan < 50){          return (50 - $kekeruhan) / (50-0);      } else{          return 0;      }  }  function kekKeruh($kekeruhan){      if($kekeruhan <= 41){          return 0;      } elseif ($kekeruhan > 41 && $kekeruhan < 100){          return ($kekeruhan - 41) / (100-41);      } else{          return 1;      }  }  function phRendah($ph){      if($ph <= 7){          return 1;      } elseif ($ph > 7 && $ph <= 7.5){          return (7.5 - $ph) / (7.5-7);      } else{          return 0;      }  }  function phNormal($ph){      if($ph > 7.5 && $ph <= 8){          return ($ph - 7.5) / (8-7.5);      } elseif($ph > 8 && $kekeruhan < 8.5){          return (8.5 - $ph) / (8.5-8);      } else{          return 0;      }  }  function phTinggi($ph){      if($ph <= 8.5){          return 0;      } elseif ($ph > 8.5 && $ph < 9){          return ($ph - 8.5) / (9-8.5);      } else{          return 1;      }  }  function TambahRendah($alfa){      if($alfa <= 0){          return 1;      } else if($alfa > 0 && $alfa < 1){          return (20 - ($alfa \* (20 - 12)));      } else{          return 0;      }  }  function TambahSedang($alfa){      if($alfa > 0 && $alfa < 1){          $zn = (($alfa \* (24 - 20)) + 20);          $zt = (28 - ($alfa \* (28 - 24)));          return $total = ($zn + $zt) / 2;      } else{          return 0;      }  }  function TambahBanyak($alfa){      if($alfa <= 0){          return 0;      } elseif($alfa > 0 && $alfa < 1){          return (($alfa \* (36 - 28)) + 28);      } else{          return 1;      }  }  function perhitungan($ketinggian, $kekeruhan, $ph){      $alfa[0] = findMin(ketRendah($ketinggian), kekJernih($kekeruhan), phRendah($ph));      $z[0] = TambahBanyak($alfa[0]);      $alfa[1] = findMin(ketNormal($ketinggian), kekJernih($kekeruhan), phRendah($ph));      $z[1] = TambahSedang($alfa[1]);      $alfa[2] = findMin(ketTinggi($ketinggian), kekJernih($kekeruhan), phRendah($ph));      $z[2] = TambahRendah($alfa[2]);      $alfa[3] = findMin(ketRendah($ketinggian), kekJernih($kekeruhan), phNormal($ph));      $z[3] = TambahRendah($alfa[3]);      $alfa[4] = findMin(ketNormal($ketinggian), kekJernih($kekeruhan), phNormal($ph));      $z[4] = TambahRendah($alfa[4]);      $alfa[5] = findMin(ketTinggi($ketinggian), kekJernih($kekeruhan), phNormal($ph));      $z[5] = TambahRendah($alfa[5]);      $alfa[6] = findMin(ketRendah($ketinggian), kekJernih($kekeruhan), phTinggi($ph));      $z[6] = TambahBanyak($alfa[6]);      $alfa[7] = findMin(ketNormal($ketinggian), kekJernih($kekeruhan), phTinggi($ph));      $z[7] = TambahSedang($alfa[7]);      $alfa[8] = findMin(ketTinggi($ketinggian), kekJernih($kekeruhan), phTinggi($ph));      $z[8] = TambahRendah($alfa[8]);      $alfa[9] = findMin(ketRendah($ketinggian), kekKeruh($kekeruhan), phRendah($ph));      $z[9] = TambahBanyak($alfa[9]);      $alfa[10] = findMin(ketNormal($ketinggian), kekKeruh($kekeruhan), phRendah($ph));      $z[10] = TambahSedang($alfa[10]);      $alfa[11] = findMin(ketTinggi($ketinggian), kekKeruh($kekeruhan), phRendah($ph));      $z[11] = TambahSedang($alfa[11]);      $alfa[12] = findMin(ketRendah($ketinggian), kekKeruh($kekeruhan), phNormal($ph));      $z[12] = TambahBanyak($alfa[12]);      $alfa[13] = findMin(ketNormal($ketinggian), kekKeruh($kekeruhan), phNormal($ph));      $z[13] = TambahBanyak($alfa[13]);      $alfa[14] = findMin(ketTinggi($ketinggian), kekKeruh($kekeruhan), phNormal($ph));      $z[14] = TambahSedang($alfa[14]);      $alfa[15] = findMin(ketRendah($ketinggian), kekKeruh($kekeruhan), phTinggi($ph));      $z[15] = TambahBanyak($alfa[15]);      $alfa[16] = findMin(ketNormal($ketinggian), kekKeruh($kekeruhan), phTinggi($ph));      $z[16] = TambahSedang($alfa[16]);      $alfa[17] = findMin(ketTinggi($ketinggian), kekKeruh($kekeruhan), phTinggi($ph));      $z[17] = TambahSedang($alfa[17]);      $temp\_1 = 0;      $temp\_2 = 0;        for($i = 0; $i < 18; $i++){          $temp\_1 = $temp\_1 + $alfa[$i] \* $z[$i];          $temp\_2 = $temp\_2 + $alfa[$i];      }      $hasil = $temp\_1 / $temp\_2;      return $hasil\* 1000;  }  echo "<br>";  echo "Hasil = " .$hasil . "<br>";  echo $ketinggian ."<br>";  echo $kekeruhan ."<br>";  echo $ph ."<br>";  ?>  <script>      var status\_pompa = <?php echo json\_encode($hasil);?>;      firebase.database().ref('pompa/').set({          status: 1,          durasi: status\_pompa,          }, (error) => {              if (error) {                  console.log("Upload Gagal");              } else {                  console.log("Upload Berhasil");              }          });          setInterval(function() {          firebase.database().ref('pompa/').set({          status: 0,          }, (error) => {              if (error) {                  console.log("Upload Gagal");              } else {                  console.log("Upload Berhasil");              }          });              }, status\_pompa);  </script>  </body>  </html> |

### **5.1.3 Implementasi Tampilan Sistem**

Implementasi tampilan user interface dari sistem sesuai dengan perancangan desain tampilan yang dilakukan sebelumnya, sebagai berikut :

Tampilan *splash screen* merupakan halaman awal saat kita membuka aplikasi terdapat pada Gambar 5.5, tampilan ini menampilkan logo dari aplikasi.



Gambar 5.5 Implementasi Tampilan Splash Screen

Tampilan dashboard pada halaman utama setelah tampilan *splash screen* muncul terdapat pada Gambar 5.6, menampilkan grafik data pompa dan beberapa variabel yang didapat secara *realtime*. Data variabel tersebut meliputi data kekeruhan, data pH, data ketinggian air, data status pompa, dan yang terakhir adalah waktu update dari beberapa data tersebut.

****

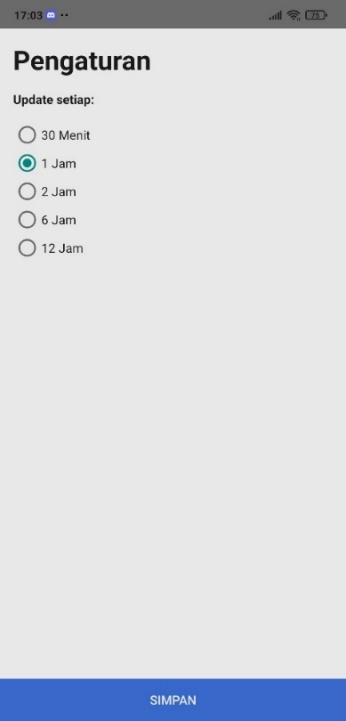
Gambar 5.6 Implementasi Tampilan Dashboard

Tampilan *history* data *realtime* terdapat pada Gambar 5.7 menampilkan 10 daftar riwayat data *realtime* yang dilakukan oleh sistem. Setiap data memuat kekeruhan air, pH air, ketinggian air, dan waktu update.

****

Gambar 5.7 Implementasi Tampilan History

Tampilan untuk melakukan pengaturan terdapat pada Gambar 5.8 ditampilkan dengan beberapa opsi durasi update otomatis pada sistem. *user* dapat memilih waktu untuk update otomatis sistem, kemudian menekan *button* simpan.

****

Gambar 5.8 Implementasi Tampilan Settings

## **Pengujian**

Pengujian merupakan proses untuk menentukan apakah hasil dari penelitian sudah sesuai dengan kebutuhan sistem dan berjalan sesuai kebutuhan yang diharapkan. Pengujian yang dilakukan dalam penelitian yaitu, pengujian fungsional dan pengujian non fungsional. Pada proses pengujian dipaparkan secara detail mengenai metode pengujian, tujuan pengujian, proses pengujian serta analisis hasil pengujian.

### **5.2.1 Pengujian Fungsional Sistem**

Pengujian fungsionalitasi sistem dilakukan dengan metode pengujian black box. Merupakan pengujian dimana menguji keseluruhan fungsional proses kerja dari sistem apakah berjalan sesuai dengan kebutuhan yang diinginkan. Ada beberapa case yang diuji diantaranya :

Deteksi pH

|  |  |
| --- | --- |
| Nama Kolom | Keterangan |
| ID | #1 |
| Name | Deteksi pH |
| Deskripsi | Usecase ini mendeskripsikan tentang pengujian deteksi pH air |
| Goals | Alat dapat memberikan nilai pH dengan akurat |
| Pre Kondisi | Arduino terhubung dengan sumber listrik |
| Aktor | Arduino dan sensor pH probe |
| Normal Flow | 1. Arduino melakukan deteksi nilai analog pH.  2. Arduino melakukan proses kalibrasi pH  3. Arduino menyimpan data pH. |
| Alternative Flow | - |
| Frekuensi Penggunaan | 5 Menit |

Tabel 5. 1 Usecase Test Deteksi pH

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Test Case | Hasil yang diharapkan | Hasil yang didapatkan | Status | Gambar |
| 1. | Arduino melakukan deteksi analog pH | Arduino dapat memberikan nilai analog pH. | Sesuai yang diharapkan | Valid | - |
| 2. | Arduino melakukan proses kalibrasi pH | Arduino mengubah nilai analog menjadi nilai pH. | Sesuai yang diharapkan | Valid | - |
| 3. | Arduino menyimpan data pH. | Arduino menyimpan nilai pH pada suatu variabel. | Sesuai yang diharapkan | Valid | - |

Deteksi Kekeruhan

|  |  |
| --- | --- |
| Nama Kolom | Keterangan |
| ID | #2 |
| Name | Deteksi Kekeruhan |
| Deskripsi | Usecase ini mendeskripsikan tentang pengujian deteksi kekeruhan air |
| Goals | Alat dapat memberikan nilai kekeruhan dengan akurat |
| Pre Kondisi | Arduino terhubung dengan sumber listrik |
| Aktor | Arduino dan sensor turbidity |
| Normal Flow | 1. Arduino melakukan deteksi nilai analog kekeruhan air.  2. Arduino melakukan proses kalibrasi nilai kekeruhan air.  3. Arduino menyimpan data kekeruhan air. |
| Alternative Flow | - |
| Frekuensi Penggunaan | 5 Menit |

Tabel 5. 2 Usecase Test Deteksi Kekeruhan

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Test Case | Hasil yang diharapkan | Hasil yang didapatkan | Status | Gambar |
| 1. | Arduino melakukan deteksi nilai analog kekeruhan air. | Arduino dapat memberikan nilai analog kekeruhan air. | Sesuai yang diharapkan | Valid | - |
| 2. | Arduino melakukan proses kalibrasi nilai kekeruhan air. | Arduino mengubah nilai analog menjadi nilai kekeruhan air. | Sesuai yang diharapkan | Valid | - |
| 3. | Arduino menyimpan data kekeruhan air. | Arduino menyimpan nilai kekeruhan air pada suatu variabel. | Sesuai yang diharapkan | Valid | - |

Deteksi Ketinggian

|  |  |
| --- | --- |
| Nama Kolom | Keterangan |
| ID | #3 |
| Name | Deteksi Kekeruhan Air |
| Deskripsi | Usecase ini mendeskripsikan tentang pengujian deteksi ketinggian air |
| Goals | Alat dapat memberikan nilai ketinggian air dengan akurat |
| Pre Kondisi | Arduino terhubung dengan sumber listrik |
| Aktor | Arduino dan sensor HCR04 |
| Normal Flow | 1. Arduino mengirimkan sinyal ultrasonic.  2. Arduino melakukan proses perhitungan jarak permukaan air. 3. Arduino menyimpan data ketinggian air. |
| Alternative Flow | - |
| Frekuensi Penggunaan | 5 Menit |

Tabel 5. 3 Usecase Test Deteksi Ketinggian

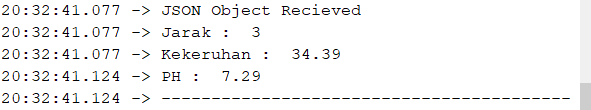
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Test Case | Hasil yang diharapkan | Hasil yang didapatkan | Status | Gambar |
| 1. | Arduino melakukan deteksi nilai analog ketinggian air. | Arduino memberikan nilai analog ketinggian air. | Sesuai yang diharapkan | Valid | - |
| 2. | Arduino melakukan proses perhitungan ketinggian air | Arduino menghitung jarak sensor dan permukaan air. | Sesuai yang diharapkan | Valid | - |
| 3. | Arduino menyimpan data ketinggian air. | Arduino menyimpan nilai ketinggian air pada suatu variabel. | Sesuai yang diharapkan | Valid | - |

Komunikasi Serial Arduino

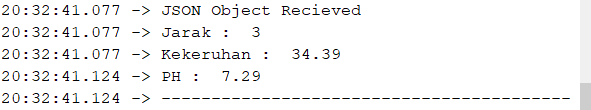
|  |  |
| --- | --- |
| Nama Kolom | Keterangan |
| ID | #4 |
| Name | Komunikasi Serial Arduino |
| Deskripsi | Usecase ini mendeskripsikan tentang komunikasi Arduino Uno dan NodeMCU ESP8266. |
| Goals | Arduino Uno dapat melakukan pengiriman data sensor ke NodeMCU. |
| Pre Kondisi | Arduino dan ESP8266 terhubung dengan sumber listrik |
| Aktor | Arduino dan NodeMCU ESP8266 |
| Normal Flow | 1. Arduino mendeteksi seluruh nilai sensor.  2. Arduino melakukan konversi data ke dalam json.  3. Arduino dan NodeMCU ESP8266 membuat komunikasi serial.  4. Arduino Uno mengirimkan data sensor ke NodeMCU ESP8266. |
| Alternative Flow | 3.1 Arduino dan NodeMCU gagal membuat komunikasi serial. 3.1.1 Melakukan pengecekan kabel jumper secara manual. 3.1.2 Menghubungkan kabel pada pin 2,3 pada Arduino. |
| Frekuensi Penggunaan | 5 Menit |

Tabel 5. 4 Usecase Test Komunikasi Serial Arduino

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Test Case | Hasil yang diharapkan | Hasil yang didapatkan | Status | Gambar |
| 1. | Arduino mendeteksi seluruh nilai sensor | Arduino memberikan nilai sensor pH, kekeruhan, dan ketinggian air | Sesuai yang diharapkan | Valid | 5. 9 |
| 2. | Arduino melakukan konversi data ke dalam json. | Arduino mengubah nilai sensor menjadi json. | Sesuai yang diharapkan | Valid | - |
| 3. | Arduino dan ESP8266 membuat komunikasi serial. | Arduino menghubungkan komunikasi serial untuk transfer data. | Sesuai yang diharapkan | Valid | - |
| 4. | Arduino mengirimkan data sensor ke ESP8266. | Arduino dapat melakukan pengiriman data pada NodeMCU. | Sesuai yang diharapkan | Valid | 5. 10 |



Gambar 5. 9 Pembacaan Sensor Arduino



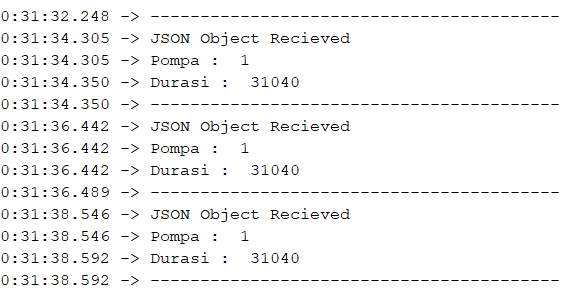
Gambar 5. 10 Nilai Sensor Diterima ESP8266

Komunikasi Serial ESP8266

|  |  |
| --- | --- |
| Nama Kolom | Keterangan |
| ID | #5 |
| Name | Komunikasi Serial ESP8266 |
| Deskripsi | Usecase ini mendeskripsikan tentang komunikasi NodeMCU ESP8266 dan Arduino Uno. |
| Goals | Alat dapat memberikan nilai pH dengan akurat |
| Pre Kondisi | Arduino dan ESP8266 terhubung dengan sumber listrik |
| Aktor | Arduino dan NodeMCU ESP8266 |
| Normal Flow | 1. ESP8266 mengambil data dari database.  2. ESP8266 menerima data status dan durasi pompa.  3. ESP8266 melakkan konversi data ke dalam json.  4. ESP8266 dan Arduino membuat komunikasi serial.  5. ESP8266 mengirimkan data ke Arduino Uno. |
| Alternative Flow | 4.1 Arduino dan NodeMCU gagal membuat komunikasi serial. 4.1.1 Melakukan pengecekan kabel jumper secara manual. 4.1.2 Menghubungkan kabel pada pin D2, D1 pada ESP8266. |
| Frekuensi Penggunaan | 5 Menit |

Tabel 5.5 Usecase Test Komunikasi Serial ESP8266

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Test Case | Hasil yang diharapkan | Hasil yang didapatkan | Status | Gambar |
| 1. | ESP8266 mengambil data dari database. | Data berhasil diambil oleh ESP8266. | Sesuai yang diharapkan | Valid | - |
| 2. | ESP8266 menerima data status dan durasi pompa. | Data diterima oleh ESP8266. | Sesuai yang diharapkan | Valid | 5.11 |
| 3. | ESP8266 melakkan konversi data ke dalam json. | Arduino mengubah nilai sensor menjadi json. | Sesuai yang diharapkan | Valid | - |
| 4. | ESP8266 dan Arduino membuat komunikasi serial. | ESP8266 menghubungkan komunikasi serial untuk transfer data | Sesuai yang diharapkan | Valid | 5.11 |
| 5. | ESP8266 mengirimkan data ke Arduino Uno. | ESP8266 dapat melakukan pengiriman data pada Arduino. | Sesuai yang diharapkan | Valid | 5.11 |



Gambar 5. 11 Komunikasi Serial Yang Diterima Arduino

Pengiriman Nilai Sensor

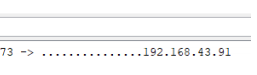
|  |  |
| --- | --- |
| Nama Kolom | Keterangan |
| ID | #6 |
| Name | Pengiriman Nilai Sensor |
| Deskripsi | Usecase ini mendeskripsikan tentang pengiriman data sensor kepada sistem monitoring. |
| Goals | NodeMCU ESP8266 dapat mengirimkan data sensor dan mendapatkan response dari sistem monitoring. |
| Pre Kondisi | 1. Arduino terhubung dengan sumber listrik  2. NodeMCU terhubung dengan koneksi wifi. |
| Aktor | Arduino, NodeMCU ESP8266, dan Database |
| Normal Flow | 1. ESP8266 melakukan penginputan ssid, password, firebase host, firebase auth.  2. ESP8266 melakukan koneksi ke wifi dan firebase  3. ESP8266 mengakses NTPClient  4. ESP8266 melakukan komunikasi serial  5. ESP8266 menerima data sensor dari Arduino Uno  6. ESP8266 melakukan pemisahan data json.  7. ESP8266 melakukan pengiriman data sensor ke database. |
| Alternative Flow | 2.1 NodeMCU gagal mengkoneksikan dengan jaringan wifi. 2.1.1 Menampilkan pesan berupa string titik “.” .  2.1.2 Menghubungkan ulang pada jaringan wifi.  2.1.3 NodeMCU gagal terkoneksi dengan wifi selama 1 menit. 2.1.3.1 Melakukan konfigurasi manual pada NodeMCU ESP8266.  4.1 NodeMCU dan Arduino gagal membuat komunikasi serial. 4.1.1 Melakukan pengecekan kabel jumper secara manual. 4.1.2 Menghubungkan kabel pada pin D2,D1 pada NodeMCU.  7.1 Gagal melakukan upload ke database  7.1.1 Menampilkan pesan gagal pada NodeMCU. |
| Frekuensi Penggunaan | 5 Menit |

Tabel 5.6 Usecase Test Pengiriman Nilai Sensor

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Test Case | Hasil yang diharapkan | Hasil yang didapatkan | Status | Gambar |
| 1. | ESP8266 melakukan penginputan ssid, password, firebase host, firebase auth. | ESP8266 menginputkan ssid dan password dengan benar | Sesuai yang diharapkan | Valid | 5.12 |
| 2. | ESP8266 melakukan koneksi ke wifi dan firebase. | NodeMCU dapat terhubung dengan jaringan Wifi dan firebase | Sesuai yang diharapkan | Valid | 5.13 |
| 3. | ESP8266 mengakses NTPClient. | NodeMCU berhasil mengakses server NTPCLient. | Sesuai yang diharapkan | Valid | - |
| 4. | ESP8266 melakukan komunikasi serial | ESP8266 menghubungkan komunikasi serial untuk transfer data | Sesuai yang diharapkan | Valid | 5.14 |
| 5. | ESP8266 menerima data sensor dari Arduino Uno. | ESP8266 dapat melakukan pengiriman data pada Arduino. | Sesuai yang diharapkan | Valid | 5.10 |
| 6. | ESP8266 melakukan pemisahan data json. | ESP8266 berhasil melakukan pemisahan data sensor. | Sesuai yang diharapkan | Valid | - |
| 7. | ESP8266 melakukan pengiriman data sensor ke database. | ESP8266 berhasil melakukan pengiriman data ke *Firebase Database* | Sesuai yang diharapkan | Valid | 5.14 |



Gambar 5. 12 Define SSID Password Firebase



Gambar 5. 13 Koneksi SSID Berhasil



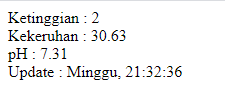
Gambar 5. 14 Data Berhasil Terupload ke Firebase

Menganalisa Air

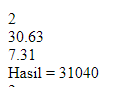
|  |  |
| --- | --- |
| Nama Kolom | Keterangan |
| ID | #7 |
| Name | Menganalisa Air |
| Deskripsi | Usecase ini mendeskripsikan tentang proses analisa kualitas air berdasarkan perhitungan metode fuzzy tsukamoto. |
| Goals | Web server dapat memberikan keputusan dan mengirimkannya ke database. |
| Pre Kondisi | Web terhubung ke database. |
| Aktor | Web dan database. |
| Normal Flow | 1. Web menerima data dari database.  2. Web melakukan analisa menggunakan fuzzy Tsukamoto.  3. Web menentukan *output* berupa durasi pompa  4. Web mengirim hasil *output* ke database. |
| Alternative Flow | 1.1 Web gagal menerima data sensor.  1.1.1 Web memberikan pesan gagal pada halaman.  4.1 Web gagal mengirim hasil *output*.  4.1.1 Web memberikan pesan gagal pada halaman. |
| Frekuensi Penggunaan | - |

Tabel 5.7 Usecase Test Menganalisa Air

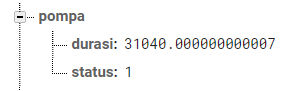
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Test Case | Hasil yang diharapkan | Hasil yang didapatkan | Status | Gambar |
| 1. | Web menerima data dari database. | Data berhasil diterima oleh web | Sesuai yang diharapkan | Valid | 5.15 |
| 2. | Web melakukan analisa menggunakan fuzzy Tsukamoto. | Web berhasil menganalisa dan menghasilkan keputusan. | Sesuai yang diharapkan | Valid | - |
| 3. | Web menentukan *output* berupa durasi pompa. | Web berhasil memberikan keputusan. | Sesuai yang diharapkan | Valid | 5.16 |
| 4. | Web mengirim hasil *output* ke database. | Web berhasil mengirim hasil keputusan ke database. | Sesuai yang diharapkan | Valid | 5.17 |



Gambar 5. 15 Get Data dari Firebase



Gambar 5. 16 Hasil Perhitungan Web



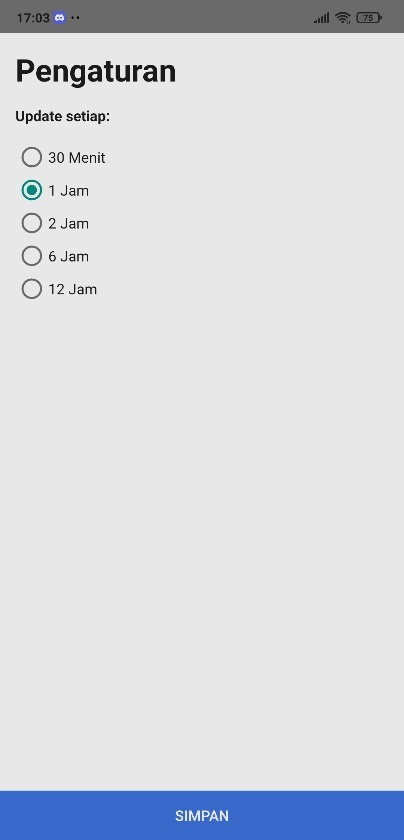
Gambar 5. 17 Web Berhasil Upload Data ke Firebase

Manajemen Database

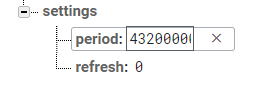
|  |  |
| --- | --- |
| Nama Kolom | Keterangan |
| ID | #8 |
| Name | Manajemen Database |
| Deskripsi | Usecase ini mendeskripsikan tentang proses insert pada database. |
| Goals | Sistem monitoring dapat memberikan data yang tersimpan pada database. |
| Pre Kondisi | Sistem monitoring membutuhkan data dari database. |
| Aktor | Sistem monitoring dan database |
| Normal Flow | 1. Sistem monitoring meminta data dari database  2. Sistem monitoring menyimpan data ke dalam database |
| Alternative Flow | - |
| Frekuensi Penggunaan | - |

Tabel 5.8 Usecase Test Manajemen Database

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Test Case | Hasil yang diharapkan | Hasil yang didapatkan | Status | Gambar |
| 1. | Sistem monitoring meminta data dari database. | Data berhasil diterima oleh sistem monitoring. | Sesuai yang diharapkan | Valid | 5.18 |
| 2. | Sistem monitoring menyimpan data ke dalam database. | Sistem monitoring berhasil menyimpan data pada database. | Sesuai yang diharapkan | Valid | 5.19 |



Gambar 5 18 Tampilan Settings



Gambar 5 19 Data Collection Settings

Analisis Fuzzy Tsukamoto

|  |  |
| --- | --- |
| Nama Kolom | Keterangan |
| ID | #9 |
| Name | Analisis Fuzzy Tsukamoto |
| Deskripsi | Usecase ini mendeskripsikan tentang proses perhitungan kualitas air menggunakan kriteria ph,kekeruhan, dan ketinggian air |
| Goals | Sistem monitoring dapat menghitung kualitas air dengan baik. |
| Pre Kondisi | Web server membutuhkan data dari database. |
| Aktor | Web server, database dan ESP8266. |
| Normal Flow | 1. Web melakukan pengelompokkan data menggunakan himpunan fuzzifikasi.  2. Web melihat data inteference.  3. Web melakukan perhitungan defuzzifikasi.  4. Web memberikan response kepada database.  5. ESP8266 menerima data dari database. |
| Alternative Flow | 4.1 Web server gagal memberikan hasil.  4.1.1 Webserver memberikan pesan gagal pada halaman.  5.1 ESP8266 gagal menerima hasil *output*.  5.1.1 ESP8266 memberikan pesan gagal pada serial. |
| Frekuensi Penggunaan | - |

Tabel 5.9 Usecase Test Analisa Fuzzy Tsukamoto

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Test Case | Hasil yang diharapkan | Hasil yang didapatkan | Status | Gambar |
| 1. | Web melakukan pengelompokkan data menggunakan himpunan fuzzifikasi. | Web dapat mengelompokkan himpunan fuzzy dengan tepat | Sesuai yang diharapkan | Valid | - |
| 2. | Web melihat data inteference. | Web dapat memberikan rule aksi yang harus dilakukan sistem | Sesuai yang diharapkan | Valid | - |
| 3. | Web melakukan perhitungan defuzzifikasi. | Web dapat melakukan perhitungan defuzzikasi. | Sesuai yang diharapkan | Valid | - |
| 4. | Web memberikan response kepada database. | Web dapat memberikan respon ke database. | Sesuai yang diharapkan | Valid | 5.17 |
| 5. | ESP8266 menerima data dari database. | ESP8266 menerima data respon dari database. | Sesuai yang diharapkan | Valid | 5.11 |

Lihat Kualitas Air

|  |  |
| --- | --- |
| Nama Kolom | Keterangan |
| ID | #10 |
| Name | Lihat Kualitas Air |
| Deskripsi | Usecase ini mendeskripsikan proses user memasuki sistem. |
| Goals | Sistem monitoring dapat memberikan data sensor secara *realtime*. |
| Pre Kondisi | User telah berhasil memasuki sistem monitoring. |
| Aktor | Sistem Monitoring, Database, dan User |
| Normal Flow | 1. Sistem monitoring menampilkan halaman utama.  2. Sistem menampilkan data sensor. |
| Alternative Flow | - |
| Frekuensi Penggunaan | - |

Tabel 5.10 Usecase Test Lihat Kualitas Air

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Test Case | Hasil yang diharapkan | Hasil yang didapatkan | Status | Gambar |
| 1. | Sistem monitoring menampilkan halaman utama. | Sistem monitoring berhasil menampilkan halaman utama. | Sesuai yang diharapkan | Valid | 5.20 |
| 2. | Sistem menampilkan data sensor. | Sistem monitoring berhasil menampilkan data sensor. | Sesuai yang diharapkan | Valid | 5.20 |



Gambar 5. 20 Tampilan Sistem Berhasil Menampilkan Data

Ubah Konfigurasi Sistem Monitoring

|  |  |
| --- | --- |
| Nama Kolom | Keterangan |
| ID | #11 |
| Name | Ubah Konfigurasi Sistem Monitoring |
| Deskripsi | Usecase ini mendeskripsikan tentang proses pengguna melakukan perubahan pengaturan pada sistem monitoring. |
| Goals | Sistem monitoring dapat mengubah pengaturan yang tersimpan pada database. |
| Pre Kondisi | User memilih konfigurasi.  User mengubah nilai konfigurasi. |
| Aktor | Sistem Monitoring dan User |
| Normal Flow | 1. Pengguna memilih halaman setting.  2. Sistem monitoring menampilkan data konfigurasi  3. Pengguna mengganti nilai konfigurasi |
| Alternative Flow | 1.1 Sistem gagal menampikkan halaman setting  1.1.1 Sistem meredirect pengguna ke halaman dashboard  2.1 sistem tidak menampilkan nilai sensor  2.1.2 Menyambungkan koneksi internet. |
| Frekuensi Penggunaan | - |

Tabel 5.11 Usecase Test Ubah Konfigurasi Sistem Monitoring

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Test Case | Hasil yang diharapkan | Hasil yang didapatkan | Status | Gambar |
| 1. | Pengguna memilih halaman setting. | Sistem dapat menampilkan halaman setting | Sesuai yang diharapkan | Valid | 5.18 |
| 2. | Sistem monitoring menampilkan data konfigurasi. | Sistem dapat menampilkan konfigurasi dari database. | Sesuai yang diharapkan | Valid | 5.18 |
| 3. | Pengguna mengganti nilai konfigurasi. | Nilai database dapat terupdate dan tersimpan | Sesuai yang diharapkan | Valid | 5.19 |

Akurasi uji coba didapatkan dengan pengujian keseluruhan fungsionalitas sistem menggunakan metode black box sebanyak 41 sampel, sebagai berikut :

Jumlah akurasi dihitung berdasarkan jumlah kesesuaian dan keberhasilan dibanding dengan total sampel ∑ 𝑇𝑜𝑡𝑎𝑙 𝑆𝑎𝑚𝑝𝑒𝑙 mengacu pada rumus yang direpresentasikan pada persamaan 3.1. terdapat dalam Bab III Uji Coba Sistem. Hasil pengujian fungsionalitas secara black box didapatkan kesesuaian dengan yang diinginkan terhadap segala aktifitas pengujian memiliki tingkat akurasi uji fungsionalitas sistem sebesar 100%

### **5.2.2 Pengujian Validitas Perhitungan**

Pengujian validitas perhitungan merupakan pengujian untuk mengetahui apakah sistem dapat menghasilkan perhitungan yang sesuai. Dimana proses melakukan pengujian dengan membandingkan hasil keputusan sistem dengan hasil excel, berdasarkan Tabel 5.12 menampilkan perbandingan terhadap hasil perhitungan sistem dan perhitungan excel, sebagai berikut :

Tabel 5.12 Pengujian Validitas Perhitungan

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Perhitungan** | **Pengujian Rumus Excel** | **Pengujian Sistem** |
| 1. | Fuzzy Tsukamoto | 30.8928 s | 30892.8 ms |
| 2. | Fuzzy Tsukamoto | 31.0992 s | 31099.2 ms |
| 3. | Fuzzy Tsukamoto | 24 s | 24000 ms |
| 4. | Fuzzy Tsukamoto | 18.4 s | 18400 ms |
| 5. | Fuzzy Tsukamoto | 29.6 s | 29600 ms |
| 6. | Fuzzy Tsukamoto | 15.84 s | 15840 ms |
| 7. | Fuzzy Tsukamoto | 31.04 s | 31040 ms |
| 8. | Fuzzy Tsukamoto | 29.28 s | 29280 ms |
| 9. | Fuzzy Tsukamoto | 30.08 s | 30080 ms |
| 10. | Fuzzy Tsukamoto | 24 s | 24000 ms |

Akurasi uji coba didapatkan dengan pengujian validitas perhitungan metode pada sistem dengan 10 sampel, sebagai berikut :

Jumlah akurasi dihitung berdasarkan jumlah kesesuaian dan keberhasilan dibanding dengan total sampel ∑ 𝑇𝑜𝑡𝑎𝑙 𝑆𝑎𝑚𝑝𝑒𝑙 mengacu pada rumus yang direpresentasikan pada persamaan 3.1. terdapat dalam Bab III Uji Coba Sistem. Hasil pengujian fungsionalitas secara black box didapatkan kesesuaian dengan yang diinginkan terhadap segala aktifitas pengujian memiliki tingkat akurasi uji validitas perhitungan sistem sebesar 100%

# **BAB VI. HASIL DAN PEMBAHASAN**

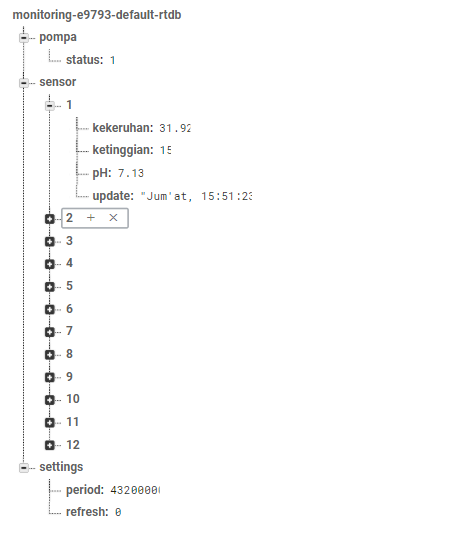
## **6.2 Hasil Penelitian**

Pada penelitian ini implementasi monitoring akuaponik dapat melakukan evaluasi kualitas air. Berdasarkan usecase testing yang telah dilakukan pada pembahasan sebelumnya, sistem monitoring akuaponik dapat mengevaluasi kualitas air pada aquarium, seperti yang ditunjukkan pada tabel 5.1 test case 3 yaitu sistem melakukan deteksi pH, pada saat sistem melakukan pengujian deteksi kekeruhan air ditunjukkan pada tabel 5.2 test case 3, dan proses deteksi ketinggian air ditunjukkan pada tabel 5.3 test case 3.

Setelah mendapatkan nilai deteksi, arduino melakukan pengiriman data pada NodeMCU ESP8266 menggunakan komunikasi serial yang ditunjukkan pada tabel 5.4 test case 3 dan 4. Di samping itu NodeMCU berhasil menghubungkan dan mengirimkan data dari arduino ke dalam database dan menerima hasil keputusan dari web yang ditunjukkan pada tabel 5.5 test case 2. Ketika keputusan yang diterima adalah melakukan menghidupkan pompa air, maka pompa akan diaktifkan oleh sistem.

Data yang tersimpan pada *database*, ditampilkan pada sistem monitoring berbasis aplikasi *mobile*. Sehingga pengguna dapat melakukan pemantauan secara *realtime* kapanpun dan dimanapun.

Dari hasil pengiriman data dari Arduino uno melalui NodeMCU ESP8266 menghasilkan sebuah data json :

****

Gambar 6 1 Data json pada Firebase

ESP8266 NodeMCU yang difokuskan untuk mengirimkan seluruh data dari Arduino ke dalam sistem monitoring. Hal ini bertujuan agar mempermudah untuk mencari solusi ketika melakukan troubleshooting dan jika mengganti komponen, maka tidak mengganti script program pada semua komponen. Perancangan hardware dapat digambarkan pada gambar 6.2

## **6.2 Pembahasan**

Berdasarkan pengujian dan hasil penelitian yang telah didapatkan, maka terdapat pembahasan sebagai berikut :

Sistem monitoring kualitas air akuaponik ini bekerja dengan baik, ketinggian air, kekeruhan air, dan pH air dapat dipantau langsung pada aplikasi *mobile* dimanapun.

Implementasi metode dapat berjalan dengan baik dengan hasil akurasi cukup baik yaitu sebesar 100% berdasarkan perbandingan dengan perhitungan manual.

Sistem berjalan sesuai dengan lingkungan yang diinginkan dengan hasil uji fungsionalitas black box 41 data uji memilik tingkat akurasi uji fungsionalitas sistem sebesar 100%.

# **BAB VII. KESIMPULAN DAN SARAN**

## **Kesimpulan**

Kesimpulan berisi uraian singkat dan jelas tentang hasil penelitian yang diperoleh sesuai dengan tujuan penelitian. Berdasarkan hasil penelitian dan hasil pengujian sistem, maka dapat ditarik kesimpulan sebagai berikut :

1. Dalam penelitian ini, sistem monitoring kualitas air dapat mendeteksi nilai keasaman air, kekeruhan air, dan ketinggian air berdasarkan waktu yang diinginkan pengguna.
2. NodeMCU ESP8266 dapat mengirim data sensor yang diperoleh dari Arduino Uno ke database sehingga pengguna dapat memantau kualitas air dengan waktu yang diinginkan pengguna.
3. Sistem monitoring kualitas air akuaponik dapat meningkatkan nilai kualitas berdasarkan kriteria nilai ph, kekeruhan, dan ketinggian melalui sebuah keputusan yang dihasilkan dari perhitungan *fuzzy tsukamoto*.
4. Sistem dapat melakukan semua kebutuhan fungsional, berdasarkan pengujian – pengujian yang telah dilakukan.

## **Saran**

Saran dapat digunakan untuk menyampaikan hal-hal yang dapat diperbaiki, dikembangkan atau dijadikan penelitian lebih lanjut. Berdasarkan kesimpulan yang telah didapatkan, maka diberikan saran sebagai berikut :

Penambahan fitur notifikasi agar pengguna dapat pemberitahuan terhadap kualitas air.

Menambahkan fitur manajemen pengguna, agar sistem dapat dikembangkan lebih dari satu pengguna.

Penambahan perbandingan hasil menggunakan sensor lain.

Dilakukan penambahan perangkat keras LCD sebagai pemantau langsung sistem.

# **DAFTAR PUSTAKA**

Adriantantri, E., & Irawan, J. (2018). *Rangkaian Kontroller*. *1*(1), 56–60.

Dewi, L. P., Indahyanti, U., & S, Y. H. (2017). Pemodelan Proses Bisnis Menggunakan Activity Diagram Uml Dan Bpmn ( Studi Kasus Frs Online ). *Informatika*, 1–9.

Fatkhudin. (2016). *Toko Elektronik Lubada Jaya Kajen Dengan*. *6*(1), 23–36.

Indartono, K., Kusuma, B. A., Putra, A. P., & Kunci, K. (2020). *PERANCANGAN SISTEM PEMANTAU KUALITAS AIR PADA BUDIDAYA IKAN AIR TAWAR Program Studi Informatika , Fakultas Ilmu Komputer , Universitas Amikom Purwokerto Abstraksi Keywords : Pendahuluan Tinjauan Pustaka*. *1*(2), 11–17.

Kurniatuty, S. A., & Widodo, K. A. (2015). Rancang Bangun Sistem Kontrol Pakan Ikan dan Kekeruhan Air yang Dilengkapi Dengan Monitoring Kualitas Air Berbasis Internet of Things ( IoT ). *Informatika*, *02*(01), 1–5.

Kuswinta, A. J., Wedashwara W, I. G. P. W., & Arimbawa, I. W. A. (2019). Implementasi IoT Cerdas Berbasis Inference Fuzzy Tsukamoto pada Pemantauan Kadar pH dan Ketinggian Air dalam Akuaponik. *Journal of Computer Science and Informatics Engineering (J-Cosine)*, *3*(1), 65–74. https://doi.org/10.29303/jcosine.v3i1.245

Mubarok, F., Harliana, H., & Hadijah, I. (2015). Perbandingan Antara Metode RUP dan Prototype Dalam Aplikasi Penerimaan Siswa Baru Berbasis Web. *Creative Information Technology Journal*, *2*(2), 114. https://doi.org/10.24076/citec.2015v2i2.42

Prasetyo, A., Nurhasan, U., & Lazuardi, G. (2018). Implementasi Iot Pada Sistem Monitoring Dan Pengendali Sirkulasi Air Tanaman Hidroponik. *Jurnal Informatika Polinema*, *5*(1), 31. https://doi.org/10.33795/jip.v5i1.241

Rahayu, N., Utami, W. S., & Razabi, M. M. (2018). RANCANG BANGUN SISTEM KONTROL DAN PEMANTAUAN AQUAPONIC BERBASIS IoT PADA KELURAHAN KUTAJAYA. *ICIT Journal*, *4*(2), 192–201. https://doi.org/10.33050/icit.v4i2.93

Sandy, L. A., Akbar, R. J., & Hariadi, R. R. (2017). Rancang Bangun Aplikasi Chat pada Platform Android dengan Media Input Berupa Canvas dan Shareable Canvas untuk Bekerja dalam Satu Canvas Secara Online. *Jurnal Teknik ITS*, *6*(2). https://doi.org/10.12962/j23373539.v6i2.23782

Syaddad, H. N. (2017). Rancang Bangun Digital Archiving Di Perguruan Tinggi Menggunakan Metode Prototype Model Studi Kasus: Universitas Suryakancana. *Media Jurnal Informatika*, *9*(1), 49–57.