**BAB IV**

**METODOLOGI PENELITIAN**

1. **Diagram Metodologi Penelitian**

Metodologi penelitian adalah sekumpulan peraturan, kegiatan, dan prosedur yang digunakan oleh pelaku suatu disiplin ilmu. Metodologi juga merupakan analisis teoritis mengenai suatu cara atau metode. Penelitian merupakan suatu penyelidikan yang sistematis untuk meningkatkan sejumlah pengetahuan, juga merupakan suatu usaha yang sistematis dan terorganisasi untuk menyelidiki masalah tertentu yang memerlukan jawaban. Hakekat penelitian dapat dipahami dengan mempelajari berbagai aspek yang mendorong penelitian untuk melakukan penelitian. Setiap orang mempunyai motivasi yang berbeda, di antaranya dipengaruhi oleh tujuan dan profesi masing-masing. Motivasi dan tujuan penelitian secara umum pada dasarnya adalah sama, yaitu bahwa penelitian merupakan refleksi dari keinginan manusia yang selalu berusaha untuk mengetahui sesuatu. Keinginan untuk memperoleh dan mengembangkan pengetahuan merupakan kebutuhan dasar manusia yang umumnya menjadi motivasi untuk melakukan penelitian. Adapun tujuan Penelitian adalah penemuan, pembuktian dan pengembangan ilmu pengetahuan.

1. **Penemuan**. Data yang diperoleh dari penelitian merupakan data-data yang baru yang belum pernah diketahui.
2. **Pembuktian**. Data yang diperoleh dari penelitian digunakan untuk membuktikan adanya keraguan terhadap informasi atau pengetahuan tertentu.
3. **Pengembangan**. Data yang diperoleh dari penelitian digunakan untuk memperdalam dan memperluas pengetahuan yang telah ada.

Kegunaan penelitian dapat dipergunakan untuk memahami masalah, memecahkan masalah, dan mengantisipasi masalah.

1. **Memahami masalah**. Data yang diperoleh dari penelitian digunakan untuk memperjelas suatu masalah atau informasi yang tidak diketahui dan selanjutnya diketahui.
2. **Memecahkan masalah**. Data yang diperoleh dari penelitian digunakan untuk meminimalkan atau menghilangkan masalah.
3. **Mengantisipasi masalah**. Data yang diperoleh dari penelitian digunakan untuk mengupayakan agar masalah tersebut tidak terjadi.

Metodologi penelitian yang digunakan oleh penulis mengarah pada *Design* *Science* *Research* *Methodology* (DSRM) yang dikemukakan oleh Ken Peffers, Tuure Tuunanen, Marcus A.Rothenberger, dan Samir Chatterjee pada jurnal berjudul “*A Design Science Research Methodology for Information Systems* *Research*” yang terdiri dari enam tahap yaitu identifikasi masalah, penetapan tujuan, perancangan, studi kasus, evaluasi dan komunikasi. Berikut gambar alur tahapan penelitian yang dilakukan :[16]



*Gambar 5.1 Diagram Alur Design Science Research Methodology.[16]*

1. **Tahapan – Tahapan Diagram Alur Metodologi Penelitian**
2. **Identifikasi Masalah dan Motivasi**

Tahap Identifikasi Masalah dan Motivasi merupakan tahap pertama dari penelitian yang dilakukan, pada tahap ini penulis melakukan identifikasi terhadap masalah yang terjadi pada penelitian berdasarkan jurnal internasional adalah sebagai berikut:

1. Bagaimana cara tanaman dapat tumbuh dengan baik?
2. Bagaimana penerapan metode untuk dapat mempertahankan PH?

1. **Penentuan Tujuan dari Solusi Penelitian**

Setelah penulis melakukan tahap mengidentifikasi masalah yang akan diteliti, selanjutnya penulis menentukan tujuan penelitian. Tujuan penelitian ini menjelaskan tentang hasil yang ingin dicapai dalam penelitian ini yaitu :

1. Mempertahankan PH pada nilai 5.6-5.7 untuk pertumbuhan tanaman.
2. Dengan menggunakan metode PID untuk tunning nilai PH dan *fuzzy* *logic* untuk memberikan delay pompa larutan asam dan basa pada tanaman.

Untuk dapat mencapai tujuan dari penelitian yang dilakukan oleh penulis akhirnya dibuatlah penelitian dengan judul Implementasi Fuzzy Logic dan PID untuk Mempertahankan Nilai PH pada Hidroponics Assistant Berbasis IoT.

1. **Perancangan dan Pengembangan Solusi**

Metode ini adalah desain sistem yang meliputi : alur kerja sistem, cara pengoprasian sistem, hasil keluaran *(output)* dengan menggunakan metode-metode seperti UML *(Unified Modeling Language)* tampilan sistem dan lain-lain yang telah disesuaikan dengan analisis kebutuhan pada tahap awal untuk menyelesaikan permasalahan tersebut. Sehingga programmer atau pihak yang terlibat dalam pembuatan kode program akan dipermudah karena sudah terarah. Perancangan dan pengembangan solusi akan menggunakan metode *prototype*

1. **Metode PID dan *Fuzzy Logic***

Dalam perancangan paper ini penulis menggunakan metode PIDdan *Fuzzy Logic*. *Fuzzy Logic* merupakan salah satu metode pengembangan perangkat lunak yang menerapkan nilai 0 dan 1 (ya dan tidak). Dengan Metode PIDdan *Fuzzy Logic* ini PH pada tamanan dapat bertahan pada angka 5.6-5.7.

* + - 1. **Log Hasil Komunikasi Dengan Pelanggan**

Penulis menulis aktivitas komunikasi dengan pelanggan dengan merangkum beberapa kebutuhan pelanggan dan membuat sebuah log aktivitas yang didefinisikan sebagai berikut :

Tabel 5.1 Log Aktivitas

|  |  |
| --- | --- |
| DAY | ACTIVITY |
| 1 | Melakukan Pengecekan keaktifan sensor PH |
| 2 | Melakukan Penyusunan pompa dan relay pada mikrokontroller |
| 3 | Memasukkan metode di koding arduino belum dapat melakukan kontrol nilai PH |

* + - 1. **Flowmap Yang Sedang Berjalan**

Setelah tahap komunikasi dilakukan, maka didefinisikan sebuah flowmap aplikasi yang sedang berjalan

* + 1. ***Flowmap* Yang Sedang Berjalan**



*Gambar 5.3 Flowmap Pembacaan Sensor PH*

Keterangan :

Pada tahap ini mikrokontroller hanya membaca nilai sensor PH pada air tanpa melakukan penjagaan nilai PH untuk dapat meningkatkan kualitas tanaman.

1. **Membangun Memperbaiki *Market***

Pada tahap ini dilakukan perancangan dan pembuatan *prototype* sistem. *Prototype* yang dibuat disesuaikan dengan kebutuhan sistem yang telah didefinisikan sebelumnya dari keluhan pelanggan atau pengguna. Perancangan sebagai berikut :

* + - 1. **Perancangan *Flowmap yang Akan Dibangun***

Pada tahap ini adalah perancangan alur secara umum dari microkontroler arduino mengirimkan data pembacaan melalui *service* *laravel* ke *firebase*.

1. ***Flowmap* Prosedur KontrolNilai PH**

Pada flowmap ini menerangkan bagaimana alur Mikrokontroler saat melakukan Kontrol Nilai PH.



*Gambar 5.5 Flowmap* Kontrol Nilai PH

Keterangan :

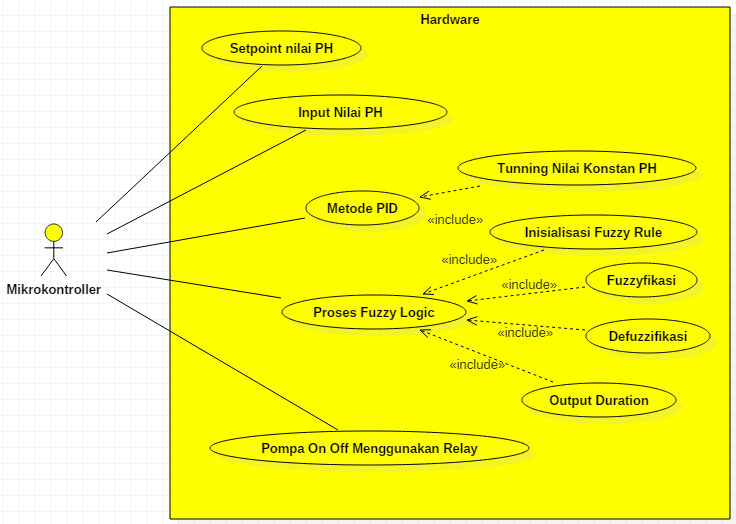
Pertama mikrokontroller membaca sensor PH kemudian nilai PH di tunning atau di kontrol menggunakan compute PID agar nilai PH konstan atau mengurangi nilai error PH. Kemudian memasukkan rule *fuzzy* PH berdasarkan volume air di dalam bak. Kemudian Melakukan memasukkan parameter nilai PH pada nilai 5.6 setelah itu melakukan fuzzyfikasi dan fuzzifikasi untuk mengetahui waktu atau durasi larutan asam dan basa yang keluar melalui pompa 1 dan pompa 2.

1. **Perancangan UML**

Pada tahap ini adalah perancangan diagram yang menjelaskan tentang aktivitas yang dapat dilakukan oleh *mikrocontroller* dan *service laravel* yang menunjukkan komunikasi perangkat keras.

* 1. ***Usecase Diagram***

*Usecase* diagram adalah diagram yang menunjukkan suatu kelompok *usecase* dan aktor serta *relationships*-nya

******

*Gambar 5.8 Usecase Diagram*

**Defini Aktor**

Pada bagian ini akan dijelaskan aktor-aktor yang terlibat dalam *hidroponics assistant*

Tabel 5.2 Definisi Aktor

|  |  |  |
| --- | --- | --- |
| **No** | **Aktor** | **Deskripsi** |
| 1. | Mikrokontroller | * Setpoint Nilai PH * Input Nilai PH * MetodePID * Proses *Fuzzy* *Logic* * Pompa On Off menggunakan Relay |

**Definisi *Usecase***

Tabel 5.3 Definisi *Usecase*

|  |  |  |
| --- | --- | --- |
| **No** | **Use Case** | **Deskripsi** |
| 1. | * Setpoint Nilai PH | Melakukan input setpont nilai PH 5.6 |
| 2. | * Input Nilai PH | Mikrokontroller membaca nilai PH pada air |
| 3 | * MetodePID | Melakukan tunning atau kontrol nilai PH untuk tidak memunculkan error terlalu jauh |
| 4 | * Proses *Fuzzy* *Logic* | Memberikan durasi pompa 1 dan pompa 2 melakukan pengisian larutan asam dan basa sehingga PH dapat bertahan pada 5.6 |
| 5 | * Pompa On Off menggunakan Relay | Relay mengaktifkan kerja pompa 1 dan pompa 2 sesuai perintah mikrokontroller |

**Skenario *Usecase***

Skenario use case diharapkan setelah berjalannya fungsional use case. Selain itu juga diberikan ulasan yang berkaitan dengan tanggapan dari sistem atas suatu aksi yang diberikan oleh aktor. Setiap use case akan diberikan sebuah skenario yang akan menjelaskan secara detail interaksi yang ada di dalamnya.

Tabel 5.4 Skenario *Usecase* Kelola *Push* data pembacaan sensor

|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nomor** | 1 |
| **Nama** | * Setpoint Nilai PH |
| **Tujuan** | Untuk melakukan setting nilai PH yang ingin di kontrol |
| **Deskripsi** | |
| **Aktor** | Mikrokontroler |
| **Skenario Utama** | |
| **Kondisi Awal** |  |
| **Aksi Aktor** | **Reaksi Sistem** |
| * Input nilai PH 5.6 | * Pada saat PH memiliki nilai 5.6 maka perintah untuk pompa 1 dan pompa 2 akan berhenti |
| **Kondisi Akhir** | Memasukkan Nilai pada setpoint sebagai acuan kerja mempertahankan PH |

* Tabel 5.5 Skenario *Usecase* Kelola *Update* Status On/Off

|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nomor** | 2 |
| **Nama** | * Input nila PH |
| **Tujuan** | Membaca nilai PH pada air |
| **Deskripsi** | |
| **Aktor** | Mikrokontroler |
| **Skenario Utama** | |
| **Kondisi Awal** |  |
| **Aksi Aktor** | **Reaksi Sistem** |
| * Membaca sensor PH | * Melakukan perhitungan untuk mendapatkan nilai PH |
| **Kondisi Akhir** | Nilai PH akan muncul pada monitor |

* Tabel 5.6 Skenario *Usecase* Metode PID

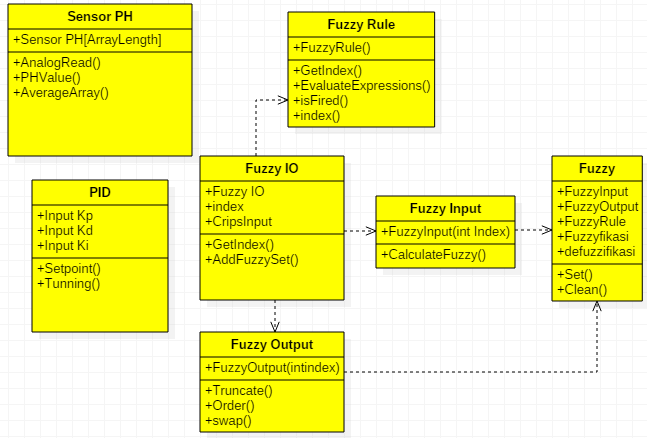
|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nomor** | 3 |
| **Nama** | * MetodePID |
| **Tujuan** | Untuk melakukan tunning atau kontrol nilai PH tidak memunculkan nilai error yang tinggi |
| **Deskripsi** | |
| **Aktor** | *Mikrokontroller* |
| **Skenario Utama** | |
| **Kondisi Awal** |  |
| **Aksi Aktor** | **Reaksi Sistem** |
| * Mengirimkan perintah servo bergerak 90 derajat | * Menjatuhkan 2 pupuk ke dalam wadah |
| **Kondisi Akhir** | 2 pupuk tercampur di bak utama |

* Tabel 5.7 Skenario *Usecase* Pembacaan Sensor

|  |  |
| --- | --- |
| **Identifikasi** | |
| **Nomor** | 4 |
| **Nama** | * Pembacaan Sensor |
| **Tujuan** | Membaca Sensor |
| **Deskripsi** | |
| **Aktor** | Mikrokontroler |
| **Skenario Utama** | |
| **Kondisi Awal** |  |
| **Aksi Aktor** | **Reaksi Sistem** |
| * Membaca sensor | * Menerima data dari sensor |
| * Mengubah data menjadi nilai *value* | Data berubah menjadi nilai |
| **Kondisi Akhir** | Data terkirim ke *laravel* |

* 1. ***Class Diagram***

*Class Diagram* adalah diagram yang menunjukan *class-class* yang ada dari sebuah sistem dan hubungannya secara logika. *Class diagram* menggambarkan struktur statis dari sebuah sistem. Karena itu *class* *diagram* merupakan tulang punggung atau kekuatan dasar dari hampir setiap metode berorientasi objek termasuk *UML.*



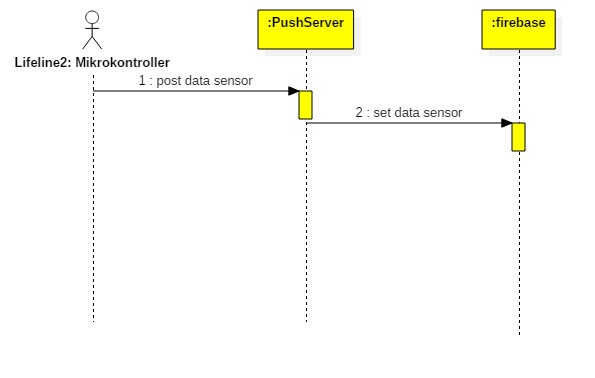
*Gambar 5.9 Class Diagram*

* 1. ***Sequence Diagram***

*Sequence diagram* disini adalah untuk menggambarkan kolaborasi dinamis antara sejumlah *object*, yang termasuk ke dalam sistem yang akan dibangun

1. ***Sequence Diagram* Kelola *Push* data pembacaan sensor**

Berikut ini merupakan *sequence diagram* Kelola *Push* data pembacaan sensor menjelaskan hubungan antara *microcontroller, service, dan firebase.*



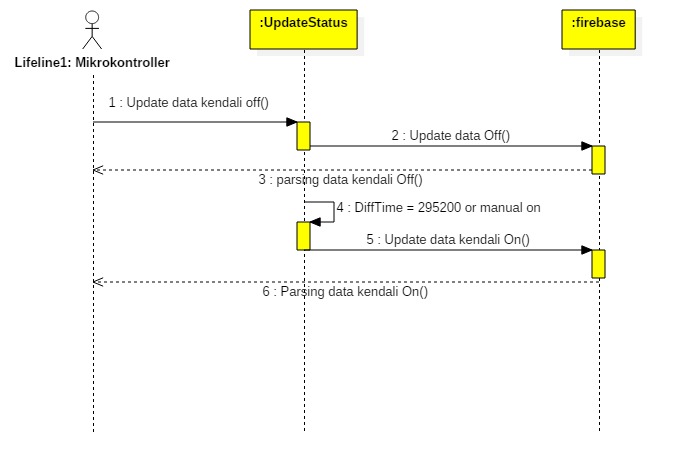
*Gambar 5.10 Sequence Diagram Kelola Push data pembacaan sensor*

Keterangan:

* 1. Microkontroller arduino mengirimkan data pembacaan sensor ke *service* *laravel*
  2. *Service* *Laravel* mengubah data pembacaan sensor dalam bentuk json
  3. Kemudian *Laravel* mengirim data ke *firebase*

1. ***Sequence Diagram* Kelola *Update* status On/Off**

Berikut ini merupakan *sequence diagram* Kelola *Update* status on/off menjelaskan hubungan antara *microcontroller, service,* dan *firebase.*

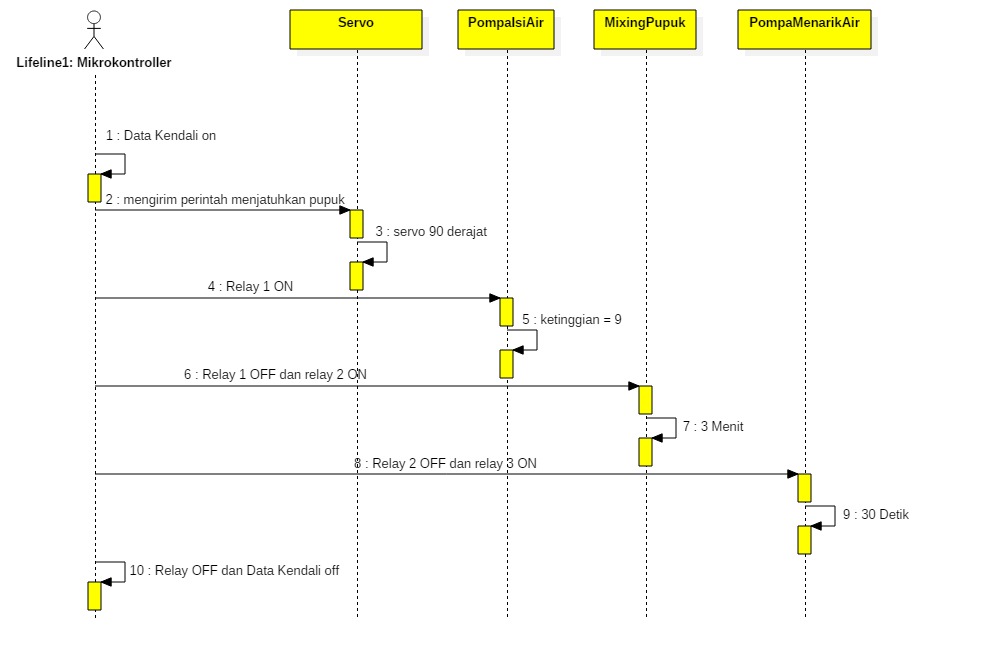


*Gambar 5.11 Sequence Diagram Kelola Update data status On/Off*

Keterangan:

1. Microkontroller arduino mengirimkan data pembacaan sensor ke *service* *laravel*
2. *Service* *Laravel* mengubah data pembacaan sensor dalam bentuk json
3. Kemudian *Laravel* mengirim data ke *firebase*
4. ***Sequence Diagram Mixing* Pupuk**

Berikut ini merupakan *sequence diagram* *Mixing* pupuk menjelaskan hubungan antara *microcontroller,* dan *hardware.*

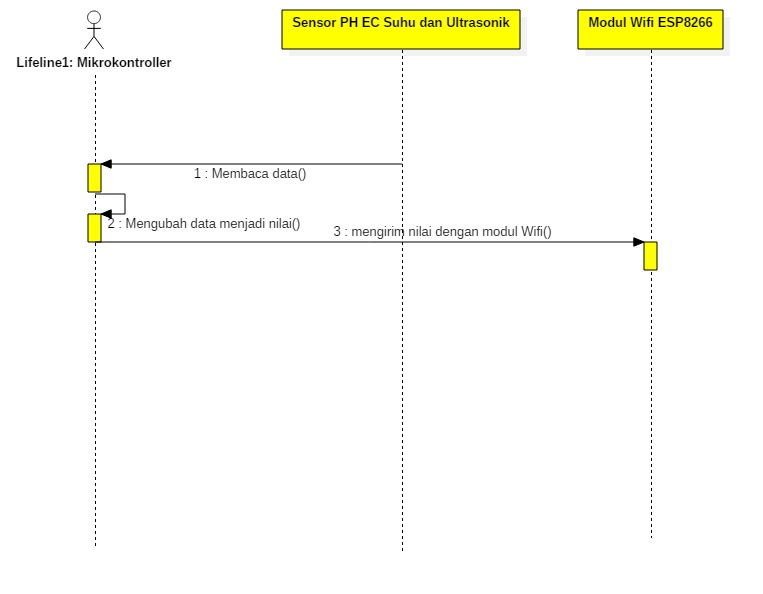
**

*Gambar 5.12 Sequence Diagram Mixing Pupuk*

Keterangan :

1. Mikrokontroler data kendali menjadi on.
2. Mikrokontroler mengirimkan perintah untuk menjatuhkan pupuk sehingga servo bergerak 90 derajat.
3. Mikrokontroler mengirim perintah Relay 1 Menjadi on hingga air mencapai ketinggan 9.
4. Kemudian Mikrokontroler mengirim Relay 1 off dan Relay 2 menjadi on selama 3 menit.
5. Mikrokontroler mengirim Relay 2 off dan Relay 3 on selama 30 detik.
6. Relay 3 Off dan data kendali ter-*Update* menjadi off kembali.
7. ***Sequence Diagram* Pembacaan Sensor**

Berikut ini merupakan *sequence diagram* pembacaan sensor menjelaskan hubungan antara microkontroler*,* sensor dan modul WIFI ESP8266*.*

******

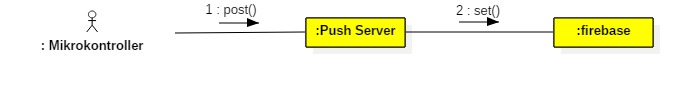
*Gambar 5.13 Sequence Diagram Pembacaan Sensor*

Keterangan :

1. Mikrokontroler membaca data yang terdapat pada Sensor PH, EC, Suhu, dan Ultrasonik
2. Mikrokontroler Mengubah data menjadi nilai
3. Selanjutnya Mikrokontroler mengirim nilai dengan modul Wifi ke Modul WIFI ESP8266
   1. ***Collaboration Diagram***

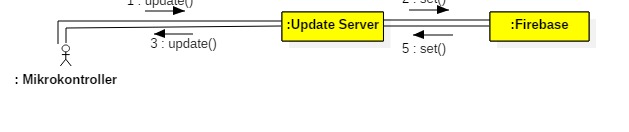
*Collaboration* *diagram* disini berfungsi untuk menggambarkan kolaborasi dinamis, dalam menunjukkan pertukaran pesan, menggambarkan *object* dan hubungannya berkaitan dengan sistem yang akan dibangun.

* + 1. ***Collaboration Diagram* Kelola *Push* data pembacaan sensor**



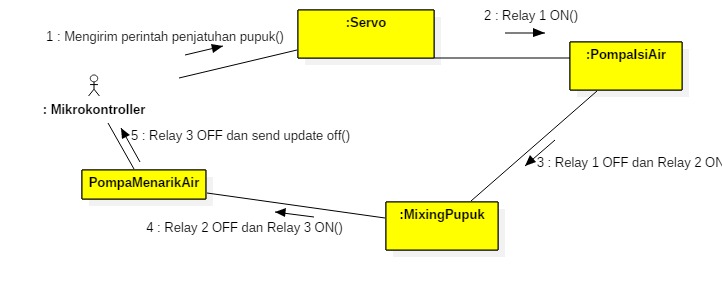
*Gambar 5.14 Collaboration Diagram Kelola Push Pembacaan Sensor*

* + 1. ***Collaboration Diagram* Kelola *Update* Status On/Off**



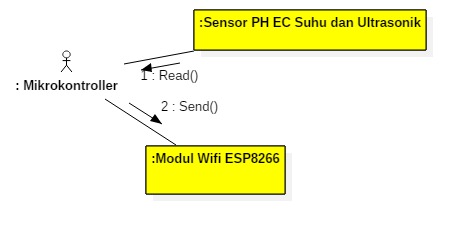
*Gambar 5.15 Collaboration Diagram Kelola Update Status On/Off*

* + 1. ***Collaboration Diagram* *Mixing* Pupuk**

**

*Gambar 5.16 Collaboration Diagram Mixing Pupuk*

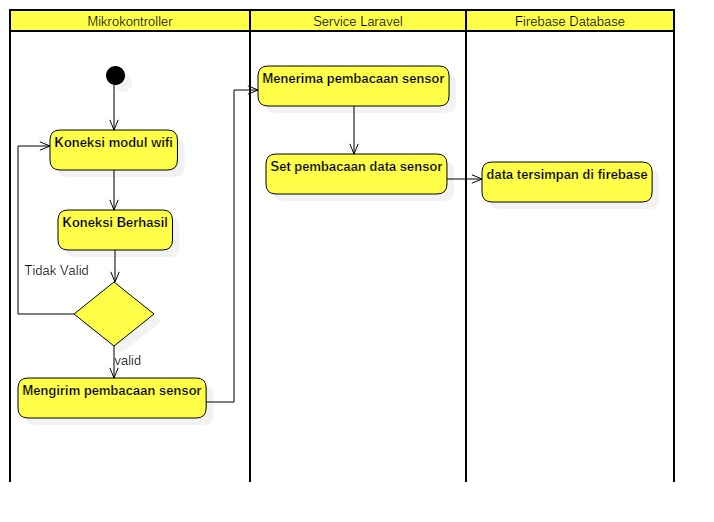
* + 1. ***Collaboration Diagram* Pembacaan Sensor**

**

*Gambar 5.17 Collaboration Diagram Pembacaan Sensor*

* 1. ***Activity Diagram***

1. ***Activity Diagram* Kelola Push Pembacaan Sensor**

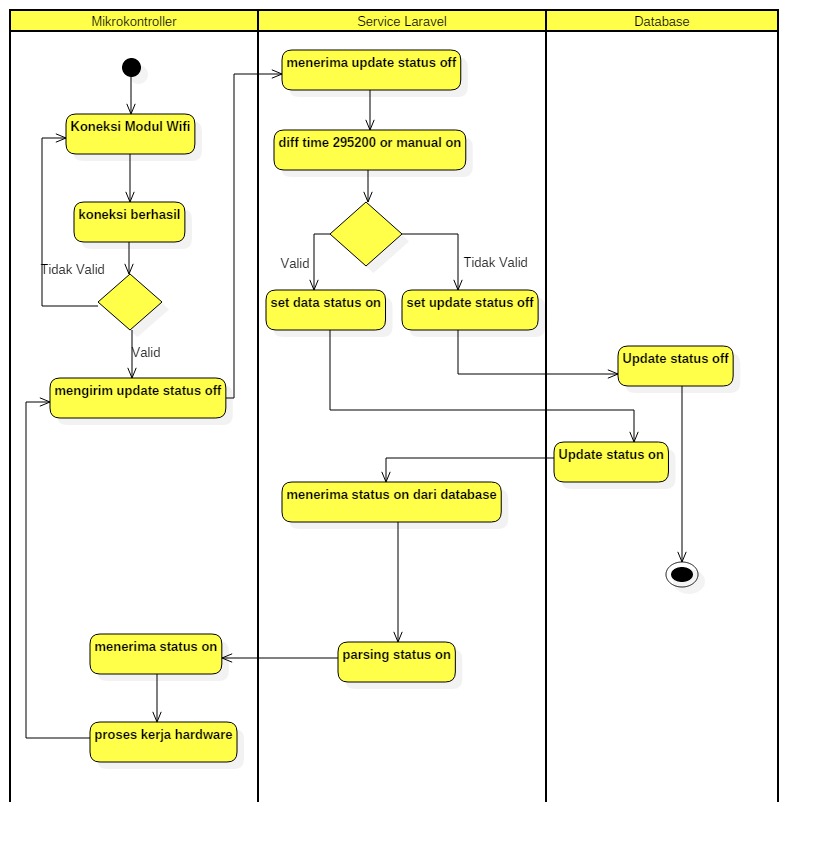
******

Valid?

*Gambar 5.18 Activity Diagram Kelola Push Pembacaan Sensor*

Keterangan :

1. Koneksi Modul Wifi dengan memasukkan SSID dan Password
2. Koneksi Berhasil dilakukan jika tidak maka kembali ke koneksi, jika berhasil maka mengirim data pembacaan sensor
3. *Service* *laravel* menerima data pembacaan sensor
4. Data tersimpan di *database*
5. ***Activity Diagram* Kelola *Update* Status On/Off**

******

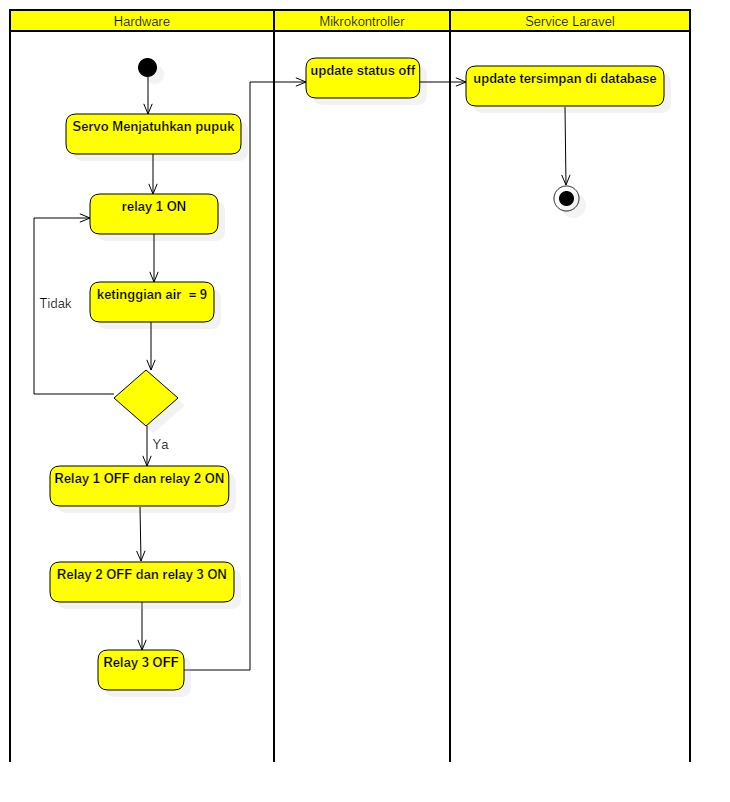
Valid?

Valid?

*Gambar 5.19 Activity Diagram Kelola Update Status On/Off*

Keterangan :

1. Koneksi Modul Wifi dengan memasukkan SSID dan Password
2. Koneksi Berhasil dilakukan jika tidak maka kembali ke koneksi, jika berhasil makan mengirim data *Update* status off
3. Service *laravel* menerima *Update* data off
4. Jika difftime =259200 atau manual = on maka mengirim *Update* status on ke *database*
5. Kemudian proses kerja *hardware*
6. Dan mengirimkan kembali status off.
7. ***Activity Diagram Mixing* Pupuk**

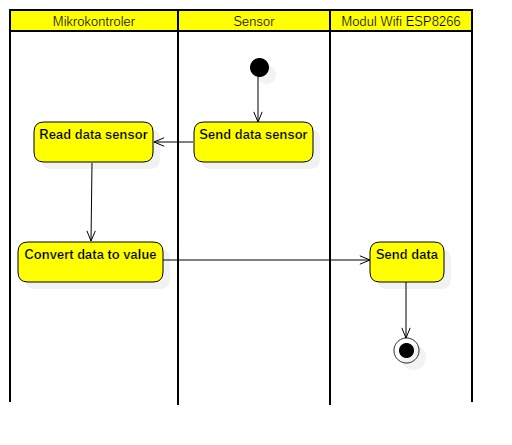
******

Valid?

*Gambar 5.20 Activity Diagram Mixing pupuk*

Keterangan :

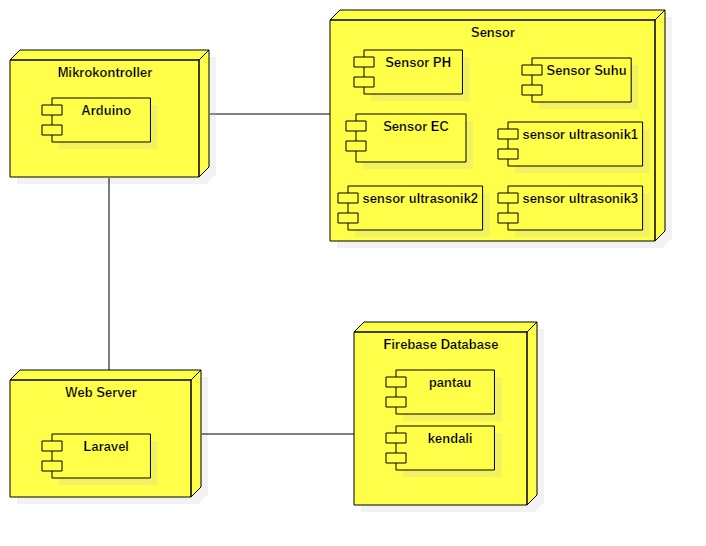
1. Tahap awal Servo menjatuhkan pupuk
2. Kemudian pompa mengisi air ke dalam wadah yang telah terdapat pupuk
3. Pompa air mengisi air dengan ketinggian 9
4. Jika tidak valid maka harus kembali mengulangin proses pengisian air hingga valid
5. Jika valid ketinggian telah mencapai 9 maka pompa air akan berhenti secara otomatis
6. Kemudian alat melakukan proses *Mixing* atau pencampuran pupuk dengan air selama 3 menit
7. Setelah tercampur pompa akan menarik air selamat 30 menit
8. Selanjutnya Mikrokontroler mengubah status menjadi Off
9. Dan perubahan data tersebut dikirimkan ke *database*.
10. ***Activity Diagram* Pembacaan Sensor**

******

*Gambar 5.21 Activity Diagram Pembacaan Sensor*

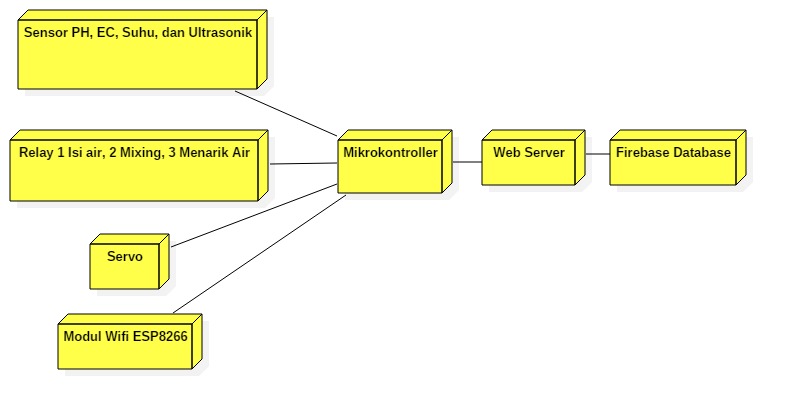
Keterangan :

1. Sensor mengirim data ke Mikrokontroler
2. Mikrokontroler membaca data sensor
3. Mikrokontroler mengconvert data ke value
4. Kemudian dari Mikrokontroler mengirim data ke Modul Wifi ESP8266
5. Selesai.
   1. ***Component Diagram***

******

*Gambar 5.22 Component Diagram*

* 1. ***Deployment Diagram***

******

*Gambar 5.23 Deployment Diagram*

1. **Perancangan *Interface***

Pada tahap ini perancangan interface yaitu sebagai desain awal untuk penggambaran sistem yang akan dibangun.

1. **Perancangan *Database***

Pada tahap ini perancangan *database* yaitu menggunakan *realtime firebase database* yang mana akan membantu proses analisa dan pembangunan *database* pada sistem. Dimana *realtime* *database* dapat menerima *Update* data terbaru secara otomatis.

* + - * 1. **Struktur *Firebase***



*Gambar 5.24 Struktur Firebase*

1. **Uji Pelanggan Mengendalikan *Market***

*prototype* dari sistem diujicoba oleh pelanggan atau pengguna. Kemudian dilakukan evaluasi kekurangan-kekurangan dari kebutuhan pelanggan. Pengembangan kemudian kembali mendengarkan keluhan dari pelanggan untuk memperbaiki *Prototype* yang ada.

* + - 1. **Pengujian Sensor EC**

Tabel 5.8 Pengujian Sensor EC

|  |  |  |
| --- | --- | --- |
| **No** | **Waktu Pengujian** | ***Value*** |
| 1 | 10.06 | 0 ms/cm |
| 2 | 10.07 | 0 ms/cm |
| 3 | 10.08 | 1 ms/cm |
| 4 | 10.09 | 3 ms/cm |
| 5 | 10.11 | 3 ms/cm |
| 6 | 10.17 | 1. ms/cm |

* + - 1. **Pengujian Sensor PH**

Tabel 5.9 Pengujian Sensor PH

|  |  |  |
| --- | --- | --- |
| **No** | **Waktu Pengujian** | ***Value*** |
| 1 | 10.06 | 5.21 |
| 2 | 10.07 | 6.35 |
| 3 | 10.08 | 7.49 |
| 4 | 10.09 | 8.63 |
| 5 | 10.11 | 8.64 |
| 6 | 10.17 | 8.66 |

* + - 1. **Pengujian Sensor Suhu**

Tabel 5.10 Pengujian Sensor Suhu

|  |  |  |
| --- | --- | --- |
| **No** | **Waktu Pengujian** | ***Value*** |
| 1 | 10.06 | 23.88 |
| 2 | 10.07 | 23.88 |
| 3 | 10.08 | 23.94 |
| 4 | 10.09 | 23.94 |
| 5 | 10.11 | 24 |
| 6 | 10.17 | 24.13 |

Kekurangan dari ujicoba yang dilakukan pelanggan yaitu pembacaan sensor PH melakukan kalibrasi pada saat alat pertama kali dihidupkan, tetapi pembacaan selanjutnya pembacaan normal.

1. **Demonstrasi**

Tahapan demonstrasi dilakukan dengan cara melakukan presentasi di depan dosen pembimbing utama dan Politeknik Pos Indonesia tujuannya untuk mendapatkan koreksi pada tahap instalasi awal aplikasi.

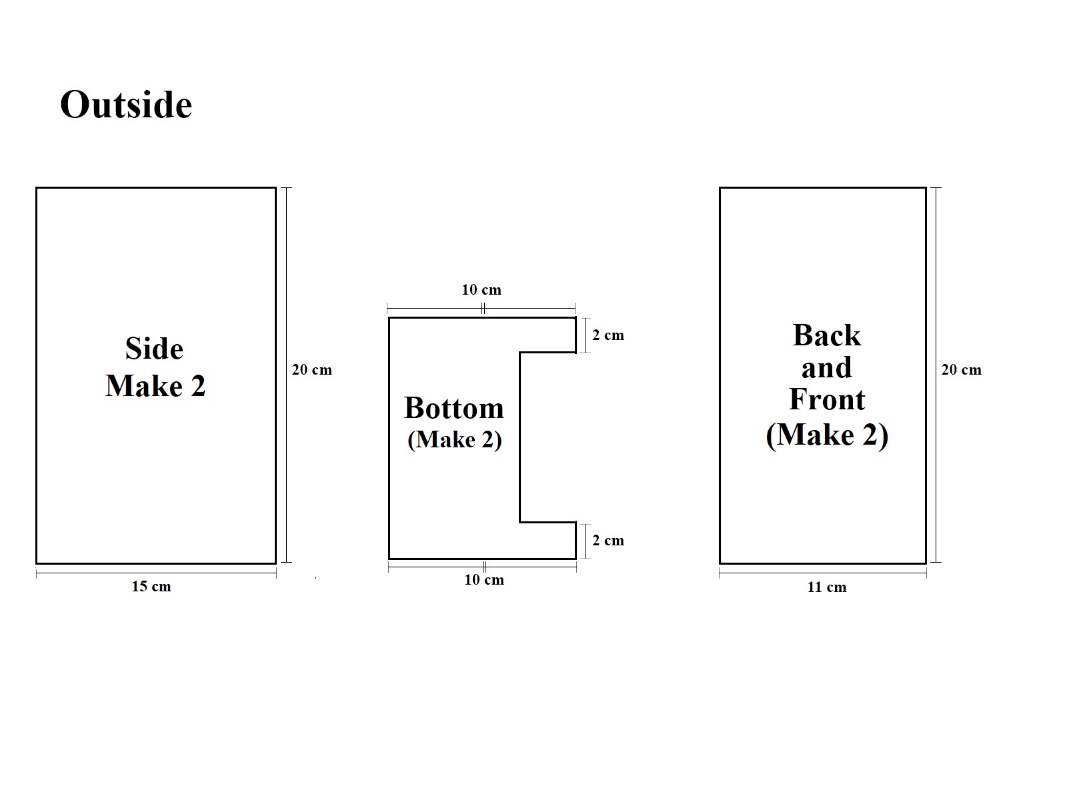
1. **Evaluasi dan Pengujian**

Evaluasi/pengujian dilakukan dengan tujuan untuk mengetahui sejauh mana aplikasi yang dikembangkan dapat berjalan dengan baik sesuai dengan fungsionalitasnya. Pengujian juga bermanfaat untuk mengukur sejauh mana aplikasi dapat menyelesaikan masalah dan apakah tujuan yang diinginkan tercapai. Pengujian dilakukan secara *blackbox* testing

1. **Komunikasi**

Tahap Komunikasi merupakan tahapan terakhir dari penelitian DSRM. Pada tahap ini proses yang dilakukan adalah men-dokumentasikan penelitian yang dikumpulkan menyebar dan dapat dibahas dalam komunitas akademik dengan cara mempublikasikan nya dalam bentuk laporan internship 1.

1. **Hasil Akhir Kerja**
2. **Desain Alat**
3. **Bagian Luar alat**

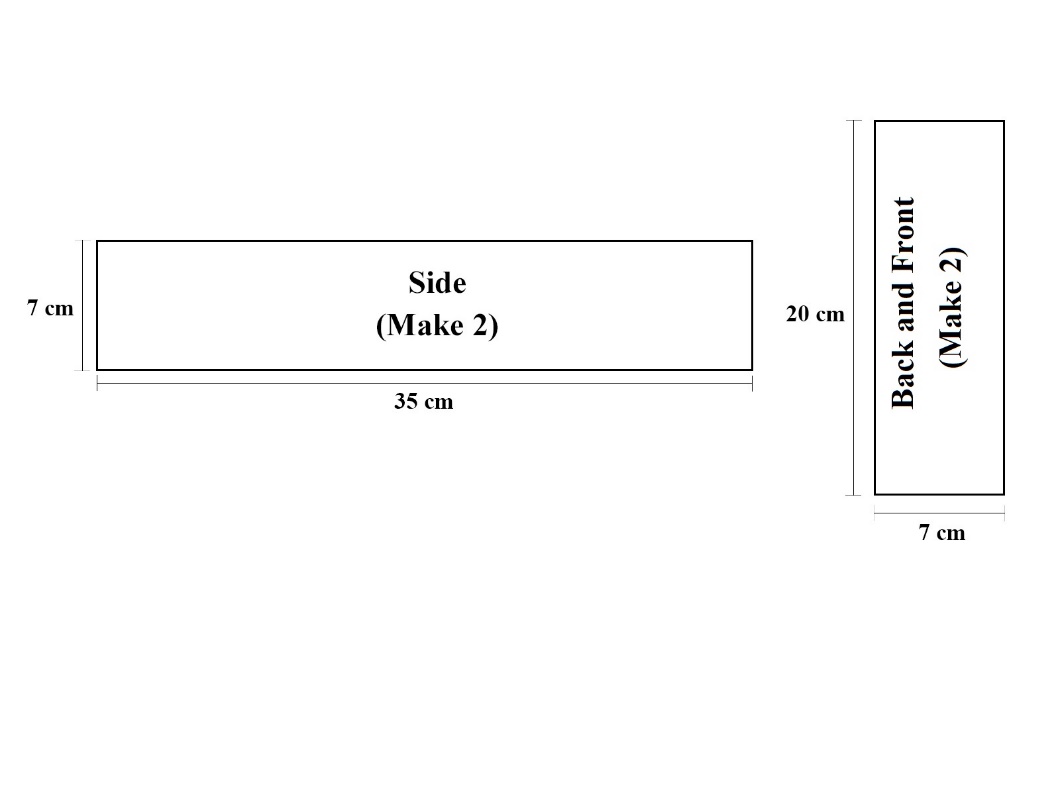
****

*Gambar 5.25 Bagian Luar Alat*

Keterangan :

* Sisi kanan dan sisi kiri untuk wadah penjatuhan pupuk 2 buah dengan tinggi 20 cm dan lebar 15 cm
* Sisi bawah untuk wadah penjatuhan pupuk 1 buah
* Sisi depan dan belakang untuk wadah penjatuhan pupuk 2 buah dengan tinggi 20 cm dan lebar 10 cm

1. **Bagian Kaki Penyangga**

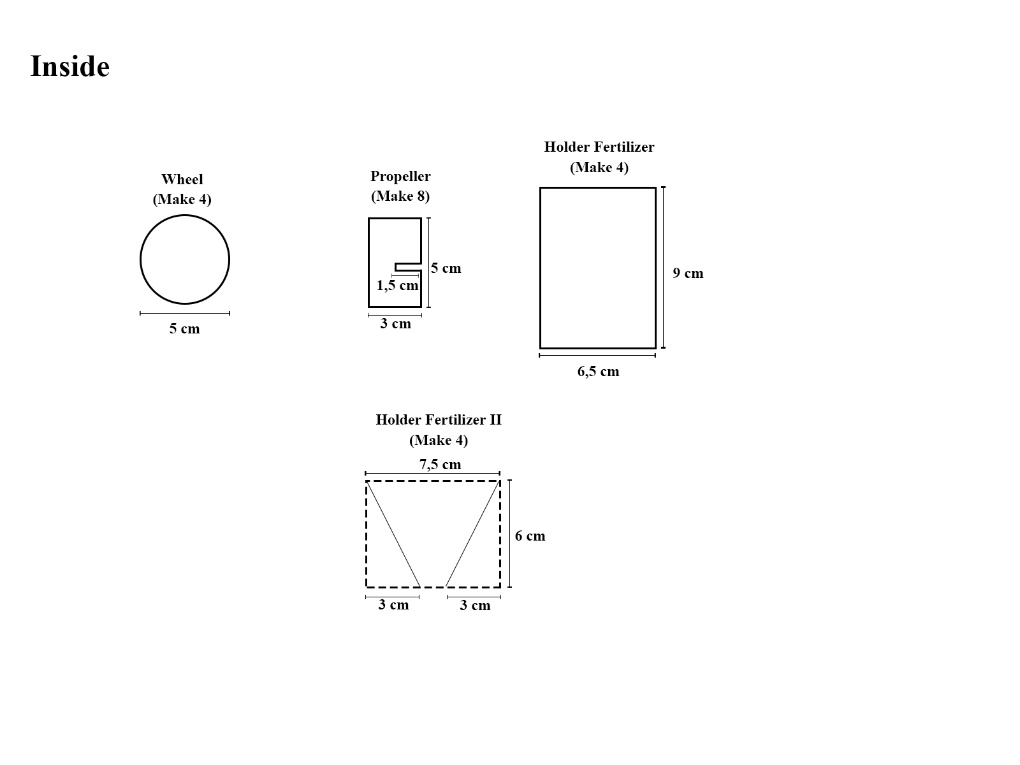
****

*Gambar 5.26 Bagian Kaki Penyangga*

Keterangan :

* Kaki penyangga digunakan untuk pengunci antara wadah penjatuhan pupuk dengan wadah *Mixing*.
* Bagian sisi kanan dan kiri dengan ukuran 7 cm dan 35 cm.
* Bagian depan dan belakang ukurang 7 cm dan 20 cm.

1. **Bagian Dalam**

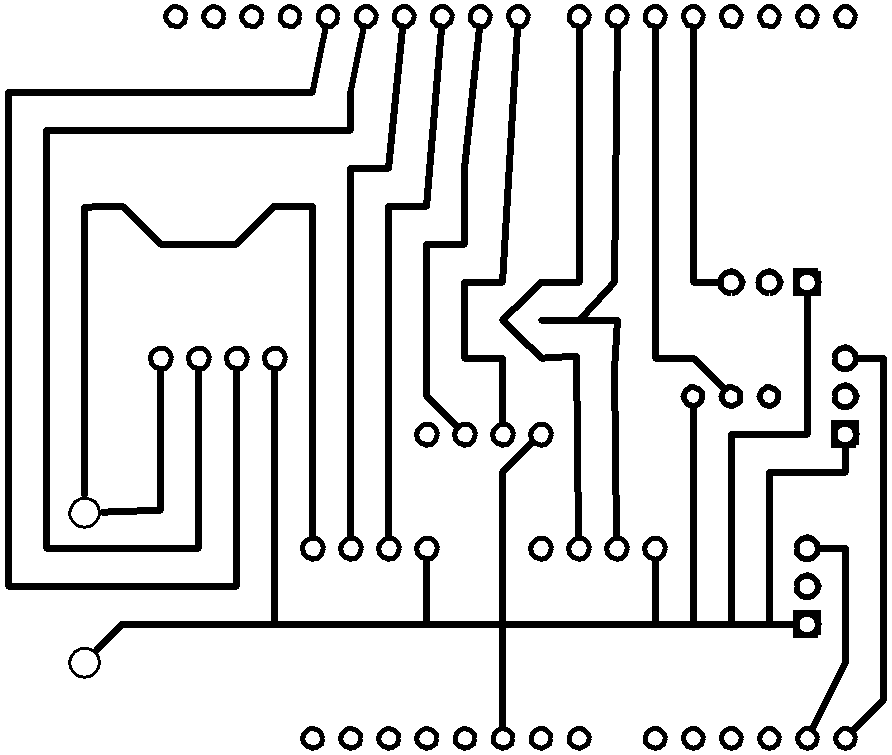


*Gambar 5.27 Bagian Dalam*

Keterangan :

* Bagian lingkarang membuat 2 bagian kemudian di antara 2 buah lingkaran ditambahkan bentuk persegi yang disatukan menjadi bentuk baling baling kemudian di tempel untuk dibuat menyerupai prototipe kincir air, dan tahap akhir disatukan dengan servo.
* Untuk segitiga digunakan untuk tempat meletakkan pupuk yang disusun terdiri dari 2 buah yang di bentuk menyerupai pemberi makanan anjing.

1. **Desain PCB Menggunakan *Fritzing***



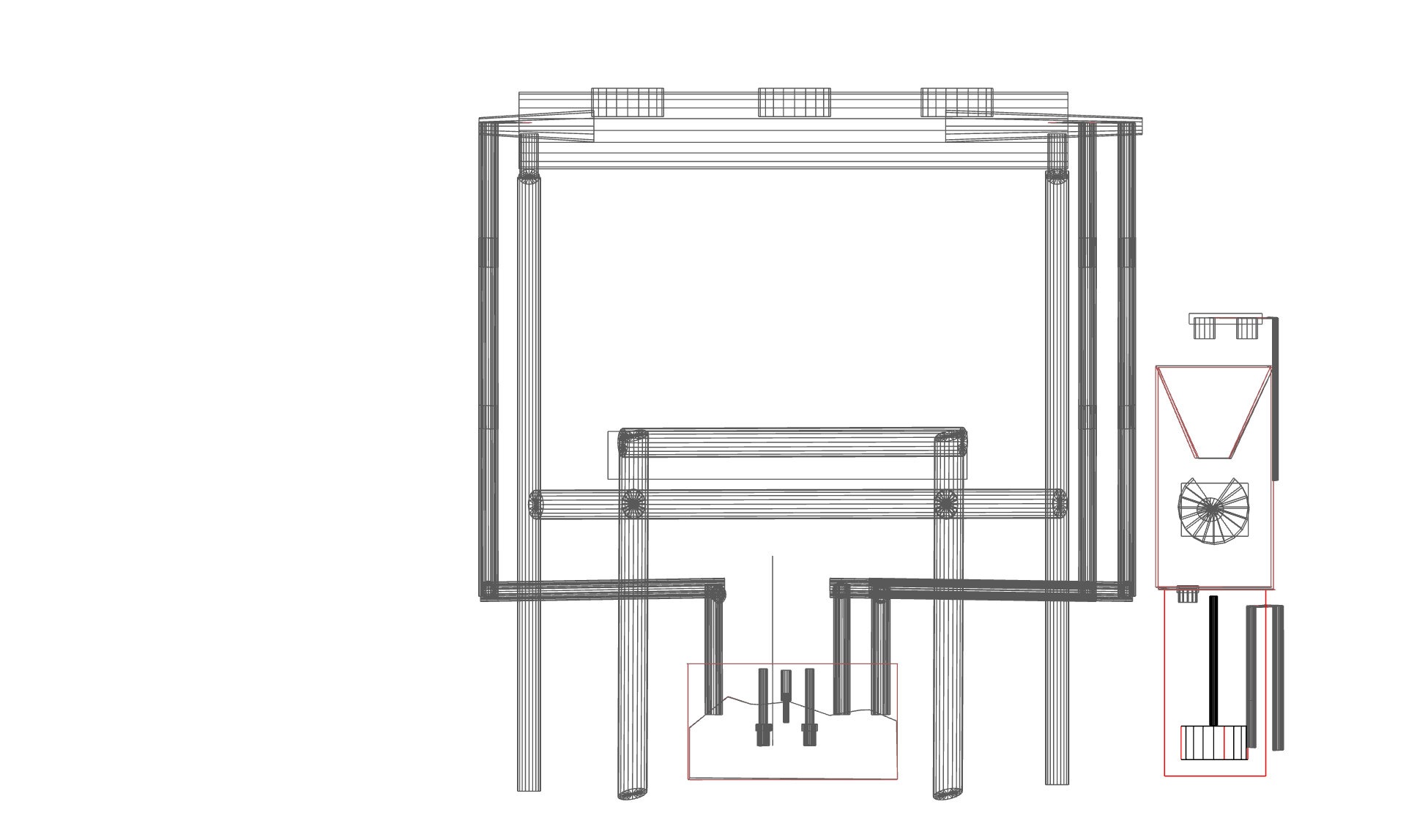
*Gambar 5.28 Desain PCB*

Keterangan :

Desain PCB ini didesain dengan menggunakan fritzing, desain ini berguna untuk merapikan rangkaian kabel yang disusun, dan dapat mengetahui posisi alat berdasarkan tempat PCB.

1. **Desain Sketsa Alat**

Adapun alur proses dari pengembangan sistem *hydroponics assistant* sebagai berikut :

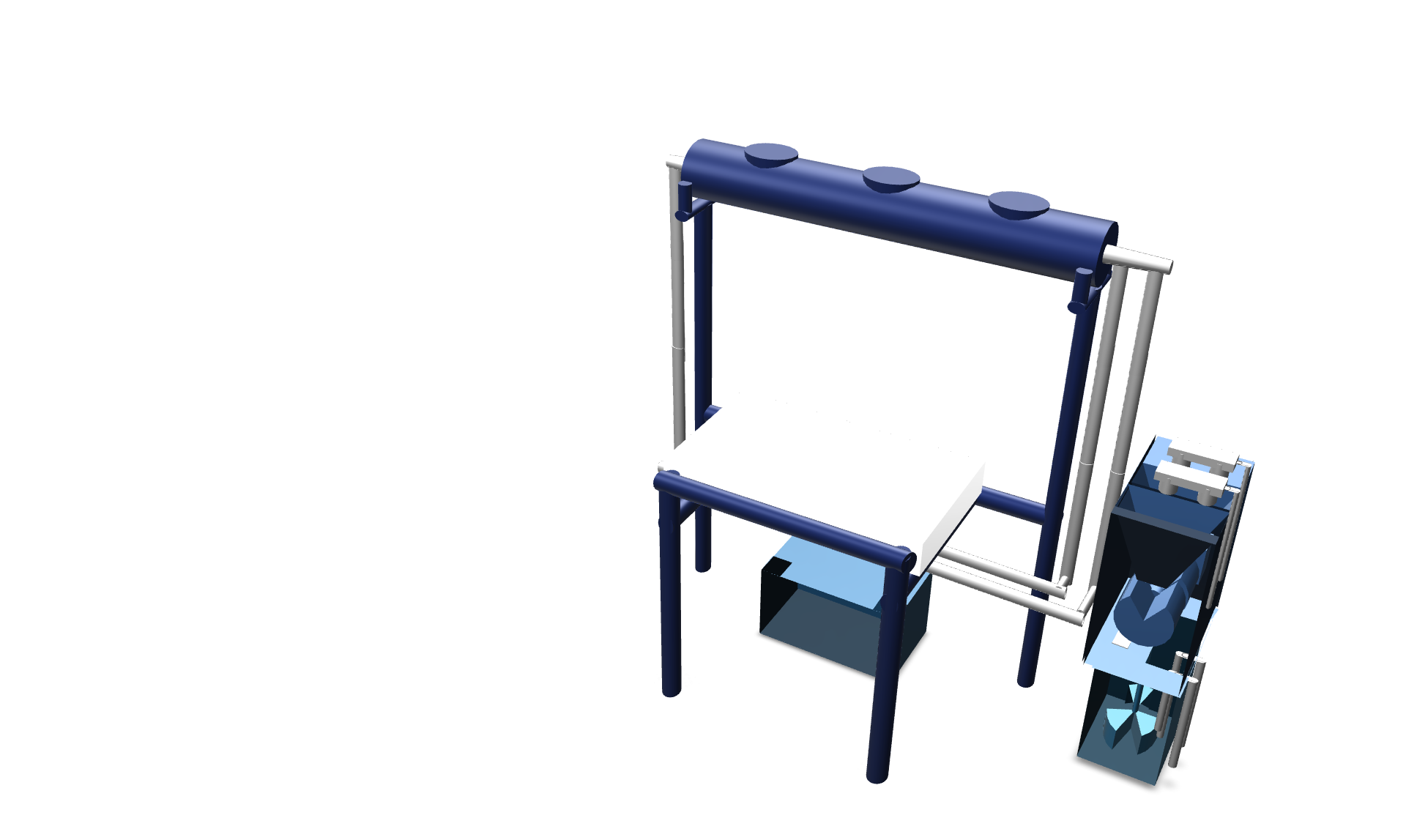


*Gambar 5.29 Desain Sketsa Alat*

Keterangan :

Desain alat ini merupakan sketsa dari keseluruhan alat yang sudah di susun dan di rapikan untuk di ujicoba

1. **Desain 3D Alat**

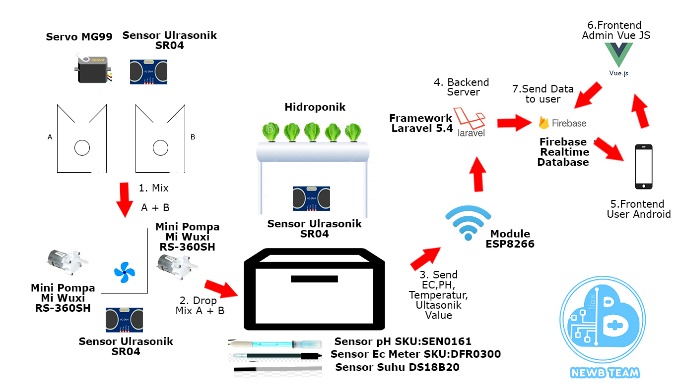


*Gambar 5.30 Desain 3D Alat*

Keterangan :

Desain alat ini merupakan bentuk 3Dkeseluruhan alat yang sudah di susun dan di rapikan untuk di ujicoba

1. **Alur Kerja *Hidroponics* *Assistant***



*Gambar 5.31 Desain Alur Kerja Hydroponics Assistant*

Keterangan :

Wadah A dan B merupakan tempat pupuk, dari kedua wadah pupuk akan jatuh berdasarkan waktu yang ditentukan oleh sistem atau pun pupuk dapat jatuh saat pengguna menekan tombol on pada aplikasi kemudian pupuk yang jatuh ditampung dalam wadah yang di dalamnya terdapat alat pencampur untuk mencampur pupuk A dan pupuk B yang sebelumnya telah terisi air oleh pompa A. Setelah itu alat *Mixing* akan melakukan *Mixing* secara otomatis sehingga kedua pupuk tercampur, setelah itu pompa B akan otomatis menarik air yang tedapat dalam wadah yang berisi campuran pupuk A dan B kemudian mencampurnya kedalam bak tempat sirkulasi tanaman hidroponik dan memberikan nutrisi kepada tanaman secara berkala. Dalam bak tersebut terdapat sensor yang digunakan untuk memantau yaitu sensor ph (keasaman) dalam air, sensor suhu yang digunakan untuk mengukur suhu di dalam air, dan sensor EC yang digunakan mengukur konduktifitas listrik di dalam air, alat-alat tersebut diintegrasikan dan berkomunikasi lewat modul wifi ESP8266, arduino yang kemudian akan mengirimkan hasil data yang diperoleh sensor yang terpasang dalam bak ke *web* server. Dalam *web* *server* ini teknologi yang digunakan adalah *web* dengan *framework laravel* kemudian *server* akan menyimpan data ke *database, database* yang digunakan adalah *firebase database realtime*, setelah itu pengguna aplikasi dapat memantau dan mengendalikan pemberian pupuk tanaman dari *smartphone android*