

## **HALAMAN SAMPUL LAPORAN PROYEK AKHIR LAPORAN PROYEK AKHIR**



### **LAPORAN PENGEMBANGAN PRODUK PROYEK AKHIR**

#### **MONITORING AQUARIUM BERBASIS IOT**

**Disusun Oleh:**

**13323020 : Naomi Aprilia Butar butar  
13323025 : Basanta Alfonso Hutasoit  
13323031 : Handika Pratama Nainggolan**

**PROGRAM STUDI D3 TEKNOLOGI KOMPUTER  
FAKULTAS VOKASI  
INSTITUT TEKNOLOGI DEL  
2025**

## **HALAMAN JUDUL LAPORAN PROYEK AKHIR**

## **LAPORAN PROYEK AKHIR**



### **LAPORAN PENGEMBANGAN PRODUK PROYEK AKHIR**

#### **MONITORING AQUARIUM BERBASIS IOT**

**Disusun Oleh:**

<b>13323020</b>	<b>:</b>	<b>Naomi Aprilia Butar butar</b>
<b>13323025</b>	<b>:</b>	<b>Basanta Alfonso Hutasoit</b>
<b>13323031</b>	<b>:</b>	<b>Handika Pratama Nainggolan</b>

**PROGRAM STUDI D3 TEKNOLOGI KOMPUTER**  
**FAKULTAS VOKASI**  
**INSTITUT TEKNOLOGI DEL**  
**2025**

## DAFTAR ISI

<b>HALAMAN SAMPUL LAPORAN PROYEK AKHIR .....</b>	<b>1</b>
<b>HALAMAN JUDUL LAPORAN PROYEK AKHIR.....</b>	<b>2</b>
<b>DAFTAR ISI.....</b>	<b>3</b>
<b>DAFTAR TABEL .....</b>	<b>6</b>
<b>DAFTAR GAMBAR.....</b>	<b>7</b>
<b>BAB I PRODUCT REQUIREMENT SPECIFICATION (SPESIFIKASI KEBUTUHAN PRODUK) .....</b>	<b>8</b>
<b>1.1. PENDAHULUAN.....</b>	<b>8</b>
<b>1.1.1. Tujuan Penulisan Dokumen .....</b>	<b>8</b>
<b>1.1.2. Latar Belakang Produk.....</b>	<b>9</b>
<b>1.1.3. Tujuan Produk.....</b>	<b>9</b>
<b>1.1.4. Ruang Lingkup Produk .....</b>	<b>10</b>
<b>1.1.5. Batasan Masalah.....</b>	<b>10</b>
<b>1.1.6. Manfaat Produk.....</b>	<b>11</b>
<b>1.1.7. Definisi dan Singkatan .....</b>	<b>11</b>
<b>1.1.8. Referensi .....</b>	<b>12</b>
<b>1.2. DESKRIPSI UMUM PRODUK.....</b>	<b>12</b>
<b>1.2.1. Permasalahan dan Persoalan .....</b>	<b>12</b>
<b>1.2.2. Produk yang menjadi Inspirasi .....</b>	<b>13</b>
<b>1.2.3. Produk yang akan dibangun .....</b>	<b>14</b>
<b>1.2.4. Konteks .....</b>	<b>15</b>
<b>1.2.5. Deskripsi Kebutuhan Produk.....</b>	<b>16</b>
<b>1.2.6. Environment Hardware dan Software.....</b>	<b>17</b>
<b>1.2.7. Metodologi dan Tools Pengembangan .....</b>	<b>18</b>
<b>BAB II PROJECT PLANNING (PP) (PERENCANAAN PENGEMBANGAN PRODUK).....</b>	<b>20</b>
<b>2.1. PENDAHULUAN.....</b>	<b>20</b>
<b>2.2. DESKRIPSI PROYEK .....</b>	<b>20</b>
<b>2.2.1. Project Organization .....</b>	<b>21</b>
<b>2.2.2. Work Breakdown Structure (WBS).....</b>	<b>22</b>
<b>2.2.3. Budget .....</b>	<b>23</b>

2.2.4. Tools .....	24
2.2.5. Resiko dan Hambatan .....	25
<b>BAB III PRODUCT DESIGN (PD) (DESAIN PENGEMBANGAN PRODUK).....</b>	<b>26</b>
3.1. PENDAHULUAN.....	26
3.2. DESKRIPSI PRODUK DESIGN.....	27
3.2.1. Current System .....	27
3.2.2. Proses Bisnis Target System .....	28
3.2.3. User Interface Layout [Software].....	29
3.2.4. Arsitektur Sistem [Hardware] .....	31
3.2.5. Desain Rangkaian Elektronik/Skematik Desain dan Desain PCB [Hardware] .....	32
3.2.6. Mekanisme Komunikasi Data [Hardware] .....	34
3.2.7. Desain Fisik/Case Hardware.....	36
<b>BAB IV PRODUCT IMPLEMENTATION (PI) (IMPLEMENTASI PENGEMBANGAN PRODUK) .....</b>	<b>38</b>
4.1. PENDAHULUAN.....	38
4.2. DESKRIPSI .....	39
4.2.1. Prinsip Implementasi.....	40
4.2.2. Lingkungan Pengembangan Terintegrasi (IDE).....	41
4.2.3. Implementasi Desain PCB.....	43
4.2.4. Implementasi Desain Fisik/Desain Case Hardware.....	43
4.2.5. Integrasi Hardware dan Software .....	44
<b>BAB V PRODUCT TESTING (PT) (PENGUJIAN PRODUK).....</b>	<b>55</b>
5.1 PENDAHULUAN.....	55
5.2 DESKRIPSI PENGUJIAN .....	55
5.2.1. BUTIR UJI .....	57
5.2.2. TOOLS PENGUJIAN .....	60
5.3 METODE PENGUJIAN.....	61
5.3.1. Pengujian Fungsional .....	61
5.3.2. Pengujian Non Fungsional .....	62
5.3.3. Pengujian Hardware.....	63
5.3.4. Pengujian Integrasi Software dan Hardware.....	64
5.3.5. Pengujian Prototipe (Prototype Testing) .....	66
<b>BAB VI PRODUCT RELEASE (PR) (PELUNCURAN PRODUK) .....</b>	<b>70</b>

<b>6.1.</b>	<b>PENDAHULUAN</b>	<b>70</b>
<b>6.2.</b>	<b>DESKRIPSI</b>	<b>70</b>
<b>6.3.</b>	<b>DAYA GUNA PRODUK</b>	<b>71</b>
<b>6.4.</b>	<b>POSTER PRODUK</b>	<b>72</b>
	<b>DAFTAR PUSTAKA</b>	<b>73</b>
	<b>LAMPIRAN</b>	<b>75</b>

## DAFTAR TABEL

<b>Table 1 Definisi dan Singkatan.....</b>	<b>11</b>
<b>Table 2 Environment Hrdware dan Software.....</b>	<b>17</b>
<b>Table 3 Tools Pengembangan .....</b>	<b>19</b>
<b>Table 4 Budget.....</b>	<b>23</b>
<b>Table 5 Tools .....</b>	<b>24</b>
<b>Table 6 Pengujian dan Hasil yang Diharapkan .....</b>	<b>62</b>

## DAFTAR GAMBAR

Gambar 1. 1 Siklus Pengembangan Sistem Model Agile-Scrum.....	8
Gambar 1. 2 Visualisasi Konteks Sistem .....	16
Gambar 1. 3 Tahapan Metodologi Penelitian .....	18
Gambar 2. 1 Project Organization .....	21
Gambar 2. 2 Work Breakdown Structure .....	22
Gambar 2. 3 Work Breakdown Structure .....	22
Gambar 3. 1 Current System.....	27
Gambar 3. 2 Proses bisnis target system dari Smart Aquarium Ikan Hias .....	28
Gambar 3. 3 Proses bisnis target system dari Smart Aquarium Ikan Hias .....	28
Gambar 3. 4 Desain Website Halaman Utama.....	29
Gambar 3. 5 Desain Website setelah memilih salah satu menu Ikan.....	29
Gambar 3. 6 Desain Website untuk Menambah Data Ikan Baru .....	30
Gambar 3. 7 Desain Website untuk Mengelola Data Ikan.....	30
Gambar 3. 8 Desain Website untuk Mengedit Data Ikan .....	30
Gambar 3. 9 Arsitektur Sistem.....	31
Gambar 3. 10 Skematik Desain Hardware.....	32
Gambar 3. 11 Desain Skematik Hardware.....	33
Gambar 3. 12 Desain PCB Bawah.....	33
Gambar 3. 13 Desain PCB Atas .....	34
Gambar 3. 14 Desain PCB tampilan atas bawah .....	34
Gambar 3. 15 Case Hardware .....	36
Gambar 3. 16 Desain Case PCB .....	36
Gambar 4. 1 Implementasi Desain PCB tampak atas dan bawah.....	43
Gambar 4. 2 Impelementasi Hardware.....	44
Gambar 5. 1 Pengujian ketika semua sensor masuk ke air .....	58
Gambar 5. 2 Pengujian ketika semua sensor masuk ke air .....	58
Gambar 5. 3 Pengujian ketika semua sensor masuk ke air .....	59
Gambar 5. 4 Tampilan saat pH masuk kedalam larutan pH 9 .....	59
Gambar 5. 5 Tampilan saat sensor kekeruhan masuk kedalam air yang bercampur tanah.....	59
Gambar 5. 6 Tampilan Dashboard Jenis Ikan Koki.....	66
Gambar 5. 7 Tampilan Dashboard Jenis Ikan Discus .....	67
Gambar 5. 8 Tampilan Dashboard Jenis Ikan Manfish.....	67
Gambar 5. 9 Tampilan Dashboard Jenis Ikan Arwana .....	67
Gambar 5. 10 Tampilan Dashboard Jenis Ikan Cupang.....	68
Gambar 5. 11 Tampilan Dashboard Jenis Ikan Koi.....	68
Gambar 5. 12 Tampilan Dashboard Jenis Ikan Glofish.....	68
Gambar 5. 13 Tampilan Dashboard Jenis Ikan Snow White .....	69
Gambar 5. 14 Tampilan Dashboard Jenis Ikan Molly .....	69
Gambar 5. 15 Tampilan Dashboard Jenis Ikan Platy .....	69
Gambar 6. 1 Poster Smart Aquarium Ikan Hias .....	72

# BAB I

## PRODUCT REQUIREMENT SPECIFICATION

### (SPESIFIKASI KEBUTUHAN PRODUK)

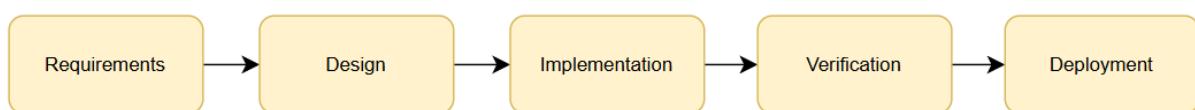
#### 1.1. PENDAHULUAN

##### 1.1.1. Tujuan Penulisan Dokumen

Dokumen ini disusun untuk mendefinisikan spesifikasi kebutuhan dalam pengembangan Smart Aquarium Ikan Hias, yang bertujuan untuk meningkatkan efisiensi dan efektivitas dalam pemeliharaan ikan hias melalui sistem otomatisasi pemantauan kualitas air serta pemberian pakan. Sistem ini dirancang agar dapat memonitor parameter utama seperti pH, *Total Dissolved Solids* (TDS), suhu, dan tingkat kekeruhan air secara real-time, serta mengotomatiskan pemberian pakan sesuai dengan jadwal dan kebutuhan ikan.

Tujuan utama dari dokumen ini adalah memberikan panduan yang terstruktur bagi tim pengembang dalam merancang, mengimplementasikan, dan mengevaluasi sistem, sehingga seluruh fitur yang dikembangkan dapat memenuhi kebutuhan pengguna secara optimal. Dokumen ini juga bertujuan untuk memastikan bahwa seluruh pihak yang terlibat dalam proyek, termasuk pengembang, pemilik akuarium, serta pemangku kepentingan lainnya, memiliki pemahaman yang jelas mengenai fungsi, spesifikasi teknis, dan batasan dari sistem yang akan dibangun.

Selain itu, dokumen ini menjadi dasar dalam penyusunan arsitektur sistem, pemilihan teknologi yang sesuai, serta pengujian dan evaluasi kinerja sistem agar dapat beroperasi secara andal. Dengan adanya dokumen ini, pengembangan sistem dapat dilakukan secara lebih terarah dan sistematis, sehingga produk yang dihasilkan benar-benar mampu memberikan solusi atas permasalahan pemeliharaan ikan hias, seperti ketidakstabilan kualitas air akibat perubahan pH, TDS, suhu, serta tingkat kekeruhan, serta pemberian pakan yang tidak teratur. Secara keseluruhan, dokumen ini berperan sebagai referensi utama dalam pengembangan Smart Aquarium Ikan Hias Berbasis IoT, guna memastikan bahwa sistem yang dibangun dapat membantu pengguna dalam merawat ikan hias secara lebih mudah, efisien, dan praktis, serta menciptakan lingkungan akuarium yang lebih sehat dan stabil bagi ikan.



Gambar 1. 1 Siklus Pengembangan Sistem Model Agile-Scrum

### **1.1.2. Latar Belakang Produk**

Pemeliharaan ikan hias dalam akuarium memiliki nilai estetika dan efek relaksasi, namun memerlukan perhatian khusus terhadap kualitas air dan pemberian pakan agar ikan tetap sehat. Jenis ikan hias yang umum dipelihara antara lain ikan koki, discus, manfish, arwana, cupang, koi, glofish, snow white, molly, dan platy. Masing-masing jenis ikan ini memiliki karakteristik dan sensitivitas yang berbeda terhadap kondisi lingkungan, sehingga membutuhkan pemantauan yang lebih teliti.

Parameter air seperti pH, Total Dissolved Solids (TDS), suhu, dan kekeruhan sangat berpengaruh terhadap kesehatan ikan, sementara ketidakseimbangan pakan dapat menyebabkan pencemaran air atau kekurangan nutrisi. Pemantauan manual sering kali tidak konsisten, menyebabkan stres, penyakit, hingga kematian ikan. Selain itu, zat beracun seperti amonia dan nitrit dari sisa pakan dan kotoran ikan dapat menurunkan kualitas air jika tidak dikelola dengan baik. Dalam jangka panjang, pemeliharaan yang tidak efisien meningkatkan biaya operasional dan menyulitkan pemilik akuarium dalam menjaga keseimbangan ekosistem. Sebagai solusi, Smart Aquarium Ikan Hias dikembangkan untuk memantau kualitas air secara real-time dan mengotomatisasi pemberian pakan. Sistem ini memungkinkan pemilik akuarium mengelola kondisi air dan pola makan ikan secara lebih praktis dan efisien melalui perangkat digital. Dengan teknologi ini, pemeliharaan ikan hias menjadi lebih mudah, sehat, dan berkelanjutan.

### **1.1.3. Tujuan Produk**

Tujuan dari pengembangan produk Monitoring Aquarium Berbasis IoT ini adalah untuk menciptakan sistem otomatis yang mampu memantau kualitas air secara real-time dan mengatur pemberian pakan ikan secara terjadwal otomatis dan manual melalui dashboard berbasis web, sehingga pemeliharaan ikan hias seperti koki, discus, manfish, arwana, cupang, koi, glofish, snow white, molly, dan platy dapat dilakukan dengan lebih mudah, efisien, dan berkelanjutan.

#### **1.1.4. Ruang Lingkup Produk**

Pengembangan Smart Aquarium Ikan Hias mencakup:

1. Pemantauan kualitas air akuarium secara *real-time*, termasuk pengukuran pH, *Total Dissolved Solids* (TDS), suhu, dan tingkat kekeruhan.
2. Sistem pemberian pakan otomatis yang dapat dijadwalkan dan disesuaikan dengan kebutuhan ikan.
3. Keterangan yang dapat terlihat pada bagian website apabila kondisi air pada aquarium tidak sesuai.
4. Integrasi melalui website berbasis IoT, memungkinkan pengguna untuk memantau dan mengontrol sistem akuarium dari jarak jauh.
5. Batasan pengembangan: Sistem hanya berfokus pada parameter air tertentu dan tidak mencakup deteksi penyakit ikan atau otomatisasi pembersihan akuarium.
6. Pengguna akhir meliputi pemilik akuarium ikan hias, pecinta ikan hias, dan komunitas pemelihara ikan skala kecil hingga menengah.

#### **1.1.5. Batasan Masalah**

Dalam pengembangan sistem Smart Aquarium Ikan Hias, terdapat beberapa batasan yang ditetapkan untuk menjaga fokus dan ketercapaian tujuan proyek, yaitu:

1. Sistem hanya dapat dikendalikan melalui dashboard web lokal yang dijalankan di laptop/server, sehingga tidak mendukung kontrol penuh secara cloud jika perangkat tidak terkoneksi internet atau laptop dimatikan.
2. Pemantauan kualitas air terbatas pada empat parameter, yaitu: suhu, pH, TDS (*Total Dissolved Solids*), dan kekeruhan. Parameter lain seperti kadar amonia, nitrit, dan oksigen terlarut belum terintegrasi.
3. Prototipe dirancang untuk akuarium skala kecil hingga menengah ( $\pm 10\text{--}40$  liter), sehingga tidak cocok langsung diterapkan untuk kolam besar atau akuarium komersial skala industri.
4. Sistem notifikasi belum menggunakan sistem push notification ke smartphone, melainkan hanya tampil di dashboard web.
5. Fungsi otomasi hanya mencakup penjadwalan pemberian pakan dan pembacaan sensor secara berkala, belum mendukung sistem otomatis pembersihan akuarium atau penggantian air.

### **1.1.6. Manfaat Produk**

Manfaat yang diharapkan dari produk ini meliputi:

1. Teknis: Memungkinkan pemantauan kualitas air akuarium secara otomatis dan *real-time* tanpa perlu pemeriksaan manual yang berulang.
2. Ekonomis: Mengoptimalkan pemberian pakan ikan agar tidak berlebih.
3. Lingkungan: Membantu menjaga keseimbangan ekosistem air dengan mengurangi limbah pakan berlebih yang dapat mencemari akuarium.
4. Pengguna: Memberikan kemudahan bagi pemilik akuarium dalam merawat ikan hias tanpa harus terus-menerus memantau kondisi air dan memberi makan secara manual.
5. Keamanan dan Kesehatan Ikan: Mengurangi risiko stres, penyakit, dan kematian ikan akibat perubahan kualitas air yang mendadak atau pola pemberian pakan yang tidak teratur.
6. Efisiensi Waktu: Memungkinkan pemilik ikan hias untuk tetap merawat akuarium dengan baik meskipun memiliki jadwal yang padat.

### **1.1.7. Definisi dan Singkatan**

Bagian ini berisi daftar istilah teknis dan singkatan yang digunakan dalam dokumen ini, beserta penjelasannya untuk memastikan pemahaman yang konsisten bagi pembaca.

*Table 1 Definisi dan Singkatan*

No	Definisi/Singkatan	Keterangan
1	pH	Potensial Hidrogen, ukuran keasaman atau kebasaan air.
2	Total Dissolved Solids (TDS)	Mendeteksi total padatan terlarut dalam air.
3	NTU	Nephelometric Turbidity Unit, satuan untuk mengukur tingkat kekeruhan air.
4	IoT	Internet of Things, konsep konektivitas perangkat melalui internet untuk pemantauan dan kontrol jarak jauh.
5	Smart Aquarium	Sistem otomatis yang memonitor kualitas air dan mengontrol pemberian pakan ikan hias.
6	Amonia	Zat beracun yang dihasilkan dari sisa makanan dan kotoran ikan, dapat membahayakan kesehatan ikan.

No	Definisi/Singkatan	Keterangan
7	Nitrit	Senyawa yang terbentuk dari penguraian amonia dan beracun bagi ikan dalam kadar tinggi.
8	Suhu	Temperatur air dalam akuarium yang memengaruhi metabolisme ikan.
9	Sensor	Perangkat yang digunakan untuk mendeteksi perubahan parameter air secara otomatis.
10	Aktuator	Komponen yang mengontrol perangkat seperti pemberi pakan otomatis atau sistem aerasi berdasarkan data sensor.

### 1.1.8. Referensi

Kurniatuty, S. A. (2019). Rancang Bangun Sistem Kontrol Pakan Ikan dan Kekeruhan Air yang Dilengkapi Dengan Monitoring Kualitas Air Berbasis Internet of Things (IoT). Institut Teknologi Nasional Malang.

Hayatunnufus, H., & Alita, D. (2020). Sistem Cerdas Pemberi Pakan Ikan Secara Otomatis. Jurnal Teknologi Dan Sistem Tertanam, 1(1), 11. <https://doi.org/10.33365/jtst.v1i1.799>

## 1.2. DESKRIPSI UMUM PRODUK

### 1.2.1. Permasalahan dan Persoalan

Permasalahan:

Pemeliharaan ikan hias dalam akuarium secara manual sering kali menghadapi tantangan dalam menjaga kualitas air tetap optimal dan memberikan pakan secara teratur. Parameter air seperti pH, *Total Dissolved Solids* (TDS), suhu, dan tingkat kekeruhan harus dipantau secara berkala agar ikan tetap sehat. Jika kualitas air tidak terjaga, ikan dapat mengalami stres, penyakit, atau bahkan kematian. Selain itu, pemberian pakan secara manual sering kali tidak konsisten, yang dapat menyebabkan kelebihan atau kekurangan makanan, sehingga berdampak pada kesehatan ikan dan kualitas air dalam akuarium.

Persoalan:

Beberapa faktor utama yang menyebabkan permasalahan tersebut antara lain:

- Pemantauan kualitas air yang tidak konsisten: Pemilik akuarium sering kali tidak memiliki alat atau waktu yang cukup untuk secara rutin mengukur pH, TDS, suhu,

dan kekeruhan air. Akibatnya, kondisi air bisa berubah drastis sebelum mereka menyadarinya.

- Akumulasi zat beracun: Sisa pakan dan kotoran ikan dapat menghasilkan amonia dan nitrit, yang dalam kadar tinggi beracun bagi ikan. Jika parameter air tidak dikontrol dengan baik, zat ini dapat terakumulasi dan memperburuk kondisi akuarium.
- Pemberian pakan yang tidak tepat: Beberapa pemilik akuarium memberikan pakan dalam jumlah berlebihan, yang menyebabkan sisa makanan mengendap dan mencemari air. Sebaliknya, jika pakan diberikan terlalu sedikit atau tidak teratur, ikan bisa kekurangan nutrisi, yang berdampak pada pertumbuhan dan kesehatannya.
- Kurangnya sistem otomatisasi: Pengelolaan akuarium masih banyak dilakukan secara manual, sehingga memerlukan lebih banyak waktu dan tenaga. Bagi pemilik yang sibuk atau sering bepergian, pemantauan dan pemberian pakan menjadi tantangan tersendiri.

Karena permasalahan-permasalahan tersebut, diperlukan solusi berupa sistem otomatis yang dapat memantau kualitas air secara *real-time* dan mengontrol pemberian pakan secara terjadwal, sehingga pemeliharaan ikan hias menjadi lebih efisien, sehat, dan mudah dikelola.

### **1.2.2. Produk yang menjadi Inspirasi**

Beberapa produk yang telah dikembangkan di pasaran menjadi inspirasi dalam pengembangan Smart Aquarium. Produk-produk ini memiliki fitur pemantauan kualitas air dan pemberian pakan otomatis, tetapi masih memiliki keterbatasan yang dapat diperbaiki dalam sistem yang akan dikembangkan. Salah satu produk yang menjadi inspirasi adalah *Fishbit Smart Aquarium Monitor*, yang mampu mengukur parameter air seperti pH, suhu, dan *Total Dissolved Solids* (TDS) secara *real-time* melalui web seluler. Keunggulan utama produk ini adalah kemampuannya untuk memberikan peringatan ketika kualitas air berada di luar batas aman. Namun, kekurangannya adalah sistem ini hanya bersifat monitoring tanpa adanya fitur kontrol otomatis serta tidak memiliki pemberian pakan otomatis. Oleh karena itu, fitur pemantauan berbasis web dan peringatan *real-time* dari Fishbit akan diadaptasi dalam produk yang dikembangkan.

Selain itu, EHEIM *Autofeeder* juga menjadi inspirasi karena kemampuannya dalam mengatur jadwal pemberian pakan ikan secara otomatis, sehingga mengurangi risiko ikan mengalami kelaparan atau *overfeeding*. Meskipun begitu, produk ini masih memiliki keterbatasan karena tidak terintegrasi dengan sistem pemantauan kualitas air serta tidak dapat dikendalikan melalui

IoT. Untuk mengatasi hal ini, sistem yang dikembangkan akan mengadopsi fitur pemberian pakan otomatis tetapi dengan integrasi IoT agar dapat dikontrol dan disesuaikan melalui website.

Selanjutnya, *Seneye Aquarium Monitoring System* juga menjadi referensi karena memiliki kemampuan untuk memantau berbagai parameter air seperti pH, suhu, amonia, dan *Total Dissolved Solids* (TDS) dengan sensor yang akurat. Sistem ini juga memiliki fitur peringatan dini jika terjadi perubahan drastis dalam kualitas air. Namun, produk ini tidak memiliki fitur pemberian pakan otomatis dan harganya relatif mahal. Oleh karena itu, teknologi sensor yang mampu mendeteksi beberapa parameter air akan diterapkan dalam sistem yang dikembangkan dengan tetap mempertimbangkan efisiensi biaya.

### **1.2.3. Produk yang akan dibangun**

Produk yang akan dibangun adalah Monitoring Aquarium Berbasis IoT, sebuah sistem otomatis yang dirancang untuk memantau dan mengontrol kualitas air serta pemberian pakan ikan secara lebih efisien. Sistem ini akan mengintegrasikan berbagai sensor dan aktuator untuk memastikan parameter air tetap optimal serta menjaga keseimbangan ekosistem dalam akuarium. Konsep utama dari produk ini adalah pemantauan real-time terhadap kondisi air akuarium serta pemberian pakan yang dapat dikendalikan melalui web berbasis IoT. Sistem ini akan memiliki beberapa fitur utama, yaitu:

- Pemantauan kualitas air secara otomatis dengan sensor yang dapat mengukur parameter penting seperti pH, suhu, *Total Dissolved Solids* (TDS), dan tingkat kekeruhan air.
- Peringatan dini melalui web jika terjadi perubahan kondisi air yang dapat membahayakan ikan, sehingga pemilik dapat segera mengambil tindakan.
- Pemberian pakan otomatis berdasarkan jadwal yang telah ditentukan atau melalui kontrol manual dari web, untuk memastikan ikan mendapatkan asupan yang sesuai dan menghindari kelebihan atau kekurangan pakan.
- Integrasi dengan web berbasis IoT, yang memungkinkan pengguna memantau kondisi akuarium dan mengatur sistem dari jarak jauh menggunakan perangkat seluler atau komputer.

#### **1.2.4. Konteks**

Sistem Smart Aquarium Ikan Hias dirancang untuk digunakan oleh pemilik akarium ikan hias skala rumah tangga maupun komunitas kecil yang ingin memelihara ikan dengan lebih praktis dan efisien. Sistem ini memungkinkan pengguna untuk memantau kondisi air dan memberi makan ikan secara otomatis melalui antarmuka berbasis web yang dapat diakses dari perangkat mobile maupun desktop.

Gambaran sistem secara umum dapat dibagi menjadi beberapa komponen utama yang saling berinteraksi:

**1) Sensor Suhu, Sensor pH, Sensor Kekeruhan, dan Sensor TDS**

Keempat sensor ini berfungsi untuk mengukur parameter penting kualitas air dalam akarium:

- Sensor Suhu (DS18B20): Mengukur suhu air secara real-time.
- Sensor pH: Mendeteksi tingkat keasaman atau kebasaan air.
- Sensor Kekeruhan (Turbidity): Mengukur kejernihan/kekeruhan air.
- Sensor TDS (Total Dissolved Solids): Mengukur kadar zat terlarut dalam air (seperti garam mineral, sisa makanan, dll).

**2) Motor Servo Pakan**

Motor servo ini bertanggung jawab untuk membuka dan menutup tempat pakan secara otomatis berdasarkan jadwal atau perintah dari pengguna melalui sistem kontrol. Motor ini bekerja berdasarkan input dari ESP32 dan dapat diatur melalui web.

**3) ESP32 (Mikrokontroler)**

Komponen pusat dari sistem yang menghubungkan semua sensor dan aktuator (servo). ESP32 membaca data dari semua sensor, mengatur waktu pemberian pakan, dan mengirimkan semua data tersebut ke server melalui protokol komunikasi.

**4) Server dan MQTT**

ESP32 mengirimkan data ke server menggunakan protokol MQTT (Message Queuing Telemetry Transport). Kami menggunakan HiveMQ yang memungkinkan komunikasi ringan dan cepat antara perangkat ESP32 dan server. Server ini berfungsi sebagai penyimpanan data dan penghubung dengan dashboard web.

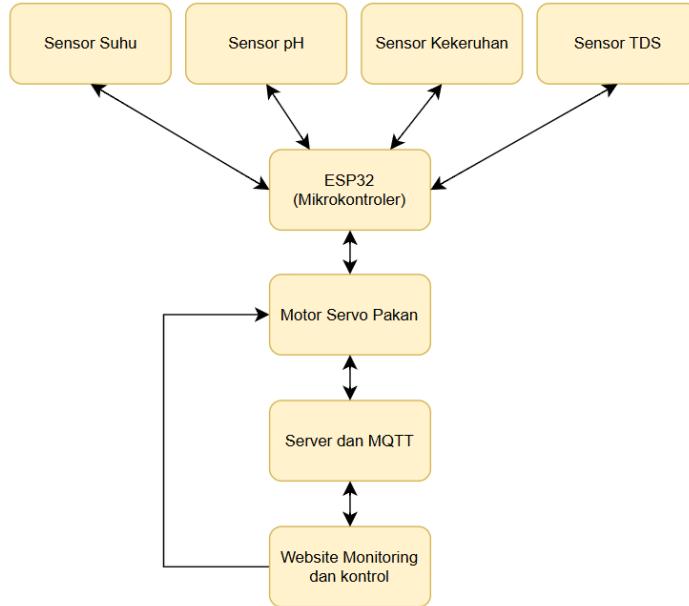
**5) Website Monitoring dan Kontrol**

Ini adalah antarmuka pengguna berbasis web yang dapat diakses melalui laptop atau perangkat lain. Di sini, pengguna dapat:

- Melihat data sensor (suhu, pH, TDS, kekeruhan) secara real-time.

- Mengatur jadwal pemberian pakan.
- Melakukan kontrol manual terhadap motor servo.
- Mendapatkan informasi status kondisi air dan rekomendasi tindakan.

Visualisasi konteks sistem dapat digambarkan seperti berikut:



Gambar 1. 2 Visualisasi Konteks Sistem

### 1.2.5. Deskripsi Kebutuhan Produk

Sistem Smart Aquarium Ikan Hias berbasis IoT terdiri dari beberapa modul utama yang mencakup fitur-fitur penting untuk memastikan kualitas air tetap optimal dan pemberian pakan yang sesuai.

Modul dan Fitur Utama:

- Modul Pemantauan Kualitas Air: mengukur parameter utama seperti pH, *Total Dissolved Solids* (TDS), suhu, dan tingkat kekeruhan secara real-time. Juga menggunakan sensor untuk mendeteksi kondisi air dan mengirimkan data ke sistem pusat.
- Modul Kontrol Pakan Otomatis: Menjadwalkan dan mengontrol pemberian pakan secara otomatis sesuai kebutuhan ikan. Juga mengatur jumlah pakan yang diberikan agar tidak berlebihan atau kurang.
- Modul Peringatan: Mengirimkan peringatan ke pengguna jika parameter air berada di luar batas aman. Juga menampilkan data kondisi akarium melalui web.
- Modul Website Berbasis IoT: Menyediakan antarmuka pengguna untuk pemantauan dan pengaturan sistem secara jarak jauh. Juga dapat diakses melalui perangkat mobile atau komputer dengan koneksi internet.

#### Kebutuhan Non-Fungsional:

- Reliabilitas: Sistem harus beroperasi secara stabil dan dapat diandalkan dalam jangka waktu lama tanpa gangguan.
- Performa: Respon sistem terhadap perubahan kondisi air harus kurang dari 5 detik.
- Keamanan: Data pengguna dan kontrol perangkat harus dilindungi dari akses tidak sah.
- Efisiensi Energi: Sistem harus dirancang dengan konsumsi daya yang rendah agar dapat bekerja secara optimal dalam waktu lama.
- Tahan Lingkungan: Perangkat harus tahan air dan dapat beroperasi dalam lingkungan akuarium tanpa mengalami kerusakan.

#### 1.2.6. Environment Hardware dan Software

Lingkungan perangkat keras dan perangkat lunak yang digunakan dalam pengembangan Smart Aquarium Ikan Hias Berbasis IoT mencakup berbagai komponen untuk memastikan sistem dapat berjalan dengan optimal.

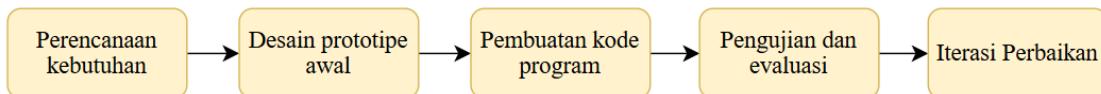
*Table 2 Environment Hrdware dan Software*

No	Hardware/Software	Spesifikasi atau Keterangan
1	ESP32	Mengontrol sensor dan aktuator serta mengirimkan data ke server atau web.
2	Sensor pH	Mengukur tingkat keasaman atau kebasaan air akuarium.
3	Total Dissolved Solids (TDS)	Mendeteksi total padatan terlarut dalam air.
4	Arduino IDE	Mengembangkan dan mengunggah kode untuk mikrokontroler.
5	IoT	Internet of Things, konsep konektivitas perangkat melalui internet untuk pemantauan dan kontrol jarak jauh.
6	Sensor Suhu	Mengukur tingkat dari suhu air untuk ikan di aquarium.
7	MQTT	Mengatur komunikasi data antara perangkat IoT dan web.

## 1.2.7. Metodologi dan Tools Pengembangan

### 1.2.7.1. Metodologi

Metodologi penelitian yang diterapkan dalam penyusunan tugas akhir Smart Aquarium Ikan Hias seperti gambar dibawah ini:



Gambar 1. 3 Tahapan Metodologi Penelitian

Metode pengembangan yang digunakan dalam proyek Smart Aquarium Ikan Hias adalah metode Agile dengan pendekatan Scrum. Metodologi ini dipilih karena memungkinkan pengembangan sistem secara iteratif dan adaptif, sehingga dapat disesuaikan dengan kebutuhan pengguna serta memungkinkan evaluasi dan perbaikan secara berkala.

Tahapan dalam metodologi ini meliputi:

1. Perencanaan kebutuhan: Mengidentifikasi kebutuhan pengguna dan merumuskan spesifikasi sistem.
2. Desain prototipe awal: Merancang sistem dari segi perangkat keras, perangkat lunak, dan komunikasi IoT.
3. Pembuatan kode program: Mengembangkan perangkat lunak untuk mikrokontroler, web dan server.
4. Pengujian dan evaluasi: Menguji fungsi sensor, aktuator, serta komunikasi data antara perangkat.
5. Iterasi perbaikan: Melakukan perbaikan dan pengembangan lebih lanjut berdasarkan feedback pengguna dan hasil pengujian.

Dengan metode *Agile-Scrum*, proyek ini dapat dikembangkan secara fleksibel dan dapat menyesuaikan fitur dengan kebutuhan pengguna secara lebih dinamis.

#### 1.2.7.2. Tools Pengembangan

Tools yang digunakan dalam proses pengembangan Smart Aquarium Ikan Hias mencakup perangkat lunak dan alat bantu yang digunakan untuk perancangan, implementasi, serta pengujian sistem.

*Table 3 Tools Pengembangan*

No	Tools	Fungsi
1	Arduino IDE	Pengembangan dan unggahan kode program ke mikrokontroler.
2	Python	Pengolahan data sensor serta pengembangan backend jika diperlukan.
3	C/C++	Bahasa pemrograman utama untuk mikrokontroler ESP32/Arduino.
4	GitHub	Manajemen kode sumber dan kolaborasi tim pengembang.
5	Fritzing/Proteus	Simulasi dan desain rangkaian elektronik sebelum perakitan perangkat keras.
6	Solidworks	Membuat model 3D dan perancangan mekanikal untuk produk industri, seperti perangkat keras dan komponen elektronik.

## **BAB II**

### **PROJECT PLANNING (PP)**

#### **(PERENCANAAN PENGEMBANGAN PRODUK)**

#### **2.1. PENDAHULUAN**

Proyek *Smart Aquarium Ikan Hias* dikembangkan sebagai solusi atas permasalahan yang sering dihadapi oleh pemilik akuarium, yaitu kesulitan dalam menjaga kualitas air dan keteraturan pemberian pakan ikan. Kondisi air yang tidak stabil, seperti perubahan pH, suhu, kadar TDS, atau tingkat kekeruhan, dapat membahayakan kesehatan ikan. Selain itu, pemberian pakan secara manual sering kali tidak konsisten, baik dari segi waktu maupun jumlah, yang dapat menyebabkan stres, penyakit, atau pencemaran air.

Dengan memanfaatkan teknologi berbasis IoT, proyek ini bertujuan menciptakan sistem pemantauan dan kontrol otomatis untuk akuarium ikan hias. Sistem akan membaca data dari berbagai sensor air secara real-time dan mengirimkannya ke platform berbasis web melalui komunikasi MQTT. Di dalam website, data ini akan dibandingkan dengan kebutuhan ideal masing-masing jenis ikan. Sistem juga dilengkapi fitur pemberian pakan otomatis dan pengelolaan data ikan.

Project Planning ini dibuat untuk merancang dan mengatur seluruh tahapan pengembangan produk agar berjalan terstruktur dan sesuai tujuan. Dokumen ini mendukung isi dari Bab *Product Requirement Specification* dengan memberikan gambaran tentang waktu pelaksanaan, sumber daya yang dibutuhkan, serta strategi implementasi teknis maupun non-teknis yang akan digunakan selama proses pengembangan produk berlangsung.

#### **2.2. DESKRIPSI PROYEK**

Proyek *Smart Aquarium Ikan Hias* merupakan sistem berbasis Internet of Things (IoT) yang dirancang untuk membantu pemilik akuarium dalam memantau dan merawat ikan hias secara lebih efisien dan praktis. Sistem ini menggabungkan sensor-sensor air seperti pH, suhu, TDS, dan kekeruhan untuk memantau kondisi air secara real-time. Data dari sensor dikirimkan melalui protokol komunikasi MQTT ke sebuah website, yang kemudian menampilkan kondisi air dan memberikan perbandingan dengan standar kebutuhan ideal berbagai jenis ikan.

Selain pemantauan, sistem ini juga dilengkapi dengan fitur pemberian pakan otomatis yang bekerja berdasarkan jadwal yang bisa diatur pengguna melalui website. Dengan begitu, pemilik akuarium tidak perlu khawatir melewatkhan waktu pemberian pakan, terutama saat sedang sibuk atau bepergian.

Sasaran pengguna dari sistem ini meliputi:

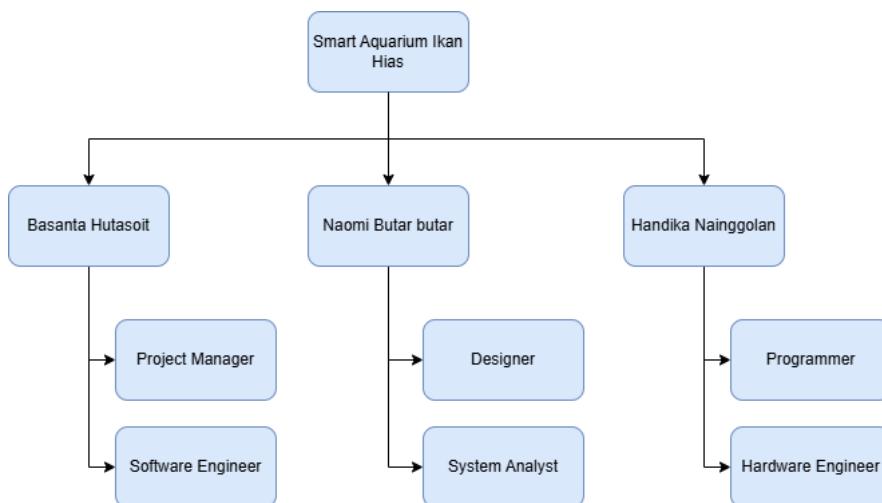
- Pemilik akuarium ikan hias, baik individu maupun komunitas hobi.
- Peternak ikan hias, yang menjalankan budi daya ikan dalam skala lebih besar.
- Pemilik toko ikan dan distributor ikan hias, yang memerlukan sistem otomatis untuk menjaga kualitas ikan sebelum dijual.

Proyek ini akan dikelola dengan pendekatan bertahap yang terstruktur, mulai dari analisis kebutuhan, desain sistem, implementasi perangkat keras dan lunak, hingga pengujian dan evaluasi. Pendekatan yang digunakan mengutamakan keterpaduan antara perangkat keras (sensor, mikrokontroler, aktuator) dan perangkat lunak (website berbasis IoT) agar sistem dapat berjalan stabil, akurat, dan mudah digunakan oleh pengguna akhir.

Deskripsi ini menjadi dasar dari perencanaan teknis dan non-teknis yang akan dijabarkan pada bagian selanjutnya, termasuk alokasi waktu, peran tim, serta strategi pengujian dan penyempurnaan produk.

### 2.2.1. Project Organization

Struktur organisasi proyek Smart Aquarium Ikan Hias dirancang untuk memastikan pembagian tugas yang jelas di antara anggota tim. Setiap anggota tim memiliki peran dan tanggung jawab spesifik dalam pengembangan sistem ini.



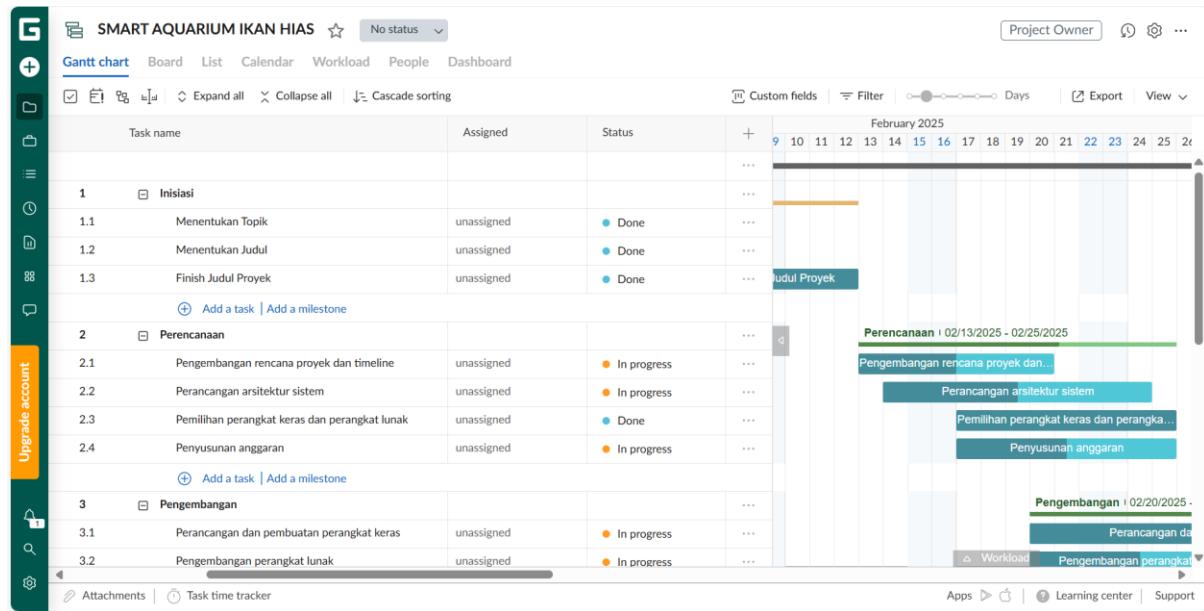
Gambar 2. 1 Project Organization

Dalam menjalankan proyek ini, mekanisme komunikasi dilakukan melalui rapat mingguan yang dipimpin oleh project manager untuk mengevaluasi perkembangan proyek serta mengatasi kendala yang muncul. Setiap anggota tim juga melaporkan progres serta hambatan yang dihadapi sesuai dengan bidang tugasnya. Selain itu, koordinasi harian dilakukan melalui platform komunikasi daring untuk memastikan kelancaran kerja tim. Struktur organisasi ini

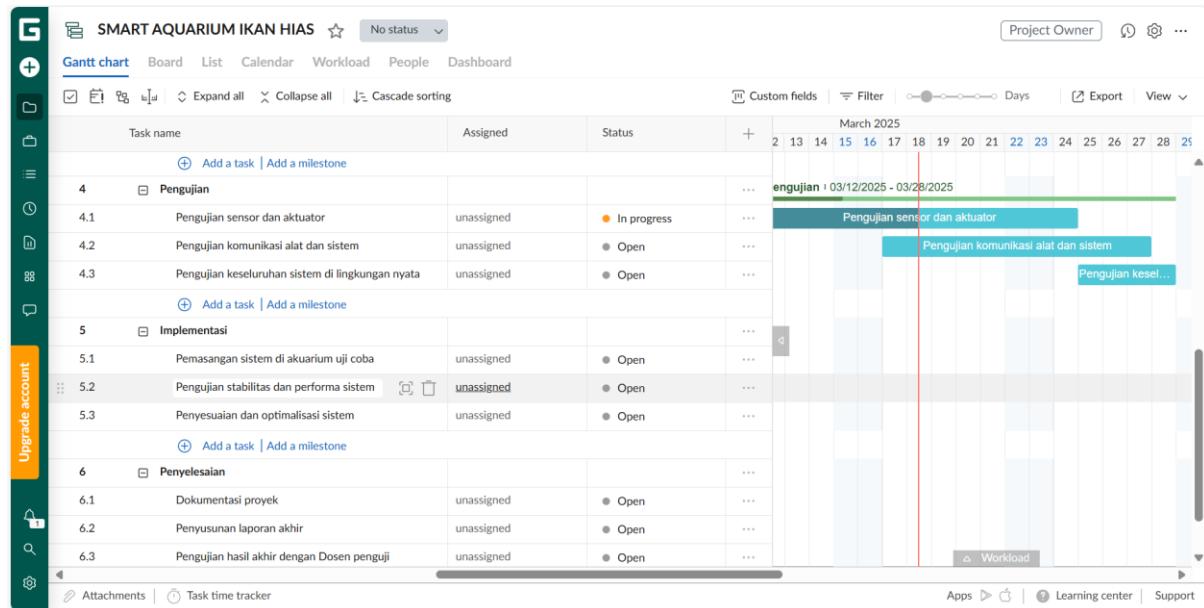
dirancang untuk memastikan bahwa setiap aspek proyek dapat berjalan secara efektif dan efisien dalam pengembangan Sistem Monitoring Kualitas Air dan Pemberian Pakan Otomatis untuk budi daya ikan hias.

### 2.2.2. Work Breakdown Structure (WBS)

Berikut adalah Work Breakdown Structure (WBS) untuk proyek Sistem Monitoring Kualitas Air dan Pemberian Pakan Otomatis:



Gambar 2. 2 Work Breakdown Structure



Gambar 2. 3 Work Breakdown Structure

### 2.2.3. Budget

Anggaran proyek Smart Aquarium Ikan Hias mencakup berbagai aspek, mulai dari pengadaan perangkat keras, perangkat lunak, hingga biaya tenaga kerja dan operasional lainnya. Alokasi anggaran disusun secara rinci untuk memastikan efisiensi dalam pengelolaan sumber daya dan menghindari pembengkakan biaya yang tidak terduga.

Biaya proyek ini dibagi ke dalam beberapa fase, yaitu:

1. Desain: Pembuatan rancangan sistem dan pemilihan komponen.
2. Pengembangan: Implementasi perangkat keras dan pemrograman perangkat lunak.
3. Pengujian: Verifikasi fungsi sistem secara menyeluruh.
4. Implementasi: Pemasangan sistem pada lingkungan operasional.

*Table 4 Budget*

No	Biaya pengembangan Proyek Akhir	Harga Satuan	Jumlah	Total
<b>A. Biaya Perangkat Keras</b>				
1	ESP32	Rp77.000	1	Rp77.000
2	Sensor pH	Rp245.000	1	Rp245.000
3	Sensor Kekeruhan	Rp114.000	1	Rp114.000
4	Sensor Suhu Air	Rp37.500	1	Rp37.500
5	Sensor TDS	Rp90.000	1	Rp90.000
6	Motor Servo	Rp25.000	1	Rp25.000
8	RTC	Rp31.000	1	Rp31.000
9	Kabel	Rp50.000	1	Rp50.000
10	Aquarium	Rp100.000	1	Rp100.000
<b>B. Biaya Operasional</b>				
13	Bubuk Kalibrasi pH	Rp10.000	1	Rp10.000
14	Solder	Rp50.000	1	Rp50.000
15	Timah Solder	Rp30.000	1	Rp30.000
16	Kabel Jumper	Rp20.000	1	Rp20.000
17	Adaptor Power Supply	Rp75.000	1	Rp75.000
<b>C. Biaya Tambahan</b>				
18	Biaya Cadangan (Perbaikan atau Penggantian Komponen)	Rp100.000	1	Rp100.000
	<b>Total</b>			<b>Rp1.054.500</b>

#### 2.2.4. Tools

Bagian ini menjelaskan alat dan teknologi yang digunakan dalam proyek Smart Aquarium Ikan Hias untuk memastikan pengembangan berjalan lancar. Pemilihan tools didasarkan pada kebutuhan proyek, efisiensi kerja, serta dukungan terhadap proses desain, pengembangan, dan pengujian sistem.

Tools yang digunakan mencakup perangkat keras, perangkat lunak, manajemen proyek, alat kolaborasi tim, serta tools untuk simulasi dan pengujian. Berikut adalah daftar tools yang digunakan dalam proyek ini:

Table 5 Tools

No	Hardware	Software	Tool managemen proyek	Alat bantu untuk kolaborasi tim	Tools untuk simulasi, desain, pengujian	Alasan pemilihan tools
1.	Komputer	Visual Studio Code	Trello	Github, Whatsapp	Selenium	Tools yang dipilih tersebut mendukung proyek yang dibangun yaitu website dengan bahasa pemrograman php
2	ESP32	Arduino IDE	Trello	Discord, Whatsapp	Proteus	Mempermudah pemrograman mikrokontroler dan simulasi rangkaian elektronik
3	Sensor pH, TDS, Suhu, Kekeruhan	MySQL Server, MQTT	Trello	Discord, Whatsapp	Postman	Digunakan untuk pengujian API dan

No	Hardware	Software	Tool managemen proyek	Alat bantu untuk kolaborasi tim	Tools untuk simulasi, desain, pengujian	Alasan pemilihan tools
						manajemen basis data
4	Motor Servo	VS Code	Asana	Discord, Whatsapp	Figma	Mempermudah desain antarmuka pengguna dan perencanaan tugas proyek

#### 2.2.5. Resiko dan Hambatan

Dalam pelaksanaan proyek Smart Aquarium Ikan Hias, terdapat beberapa risiko dan hambatan yang dapat mempengaruhi jalannya pengembangan sistem. Risiko utama yang diidentifikasi meliputi keterlambatan jadwal, anggaran yang membengkak, serta kegagalan teknologi dalam integrasi perangkat keras dan perangkat lunak. Keterlambatan jadwal dapat terjadi akibat kompleksitas pengembangan perangkat lunak dan perangkat keras, kurangnya koordinasi antar anggota tim, atau kendala teknis yang tidak terduga. Risiko ini memiliki dampak yang signifikan terhadap penyelesaian proyek dan dapat diatasi dengan perencanaan jadwal yang realistik, pemantauan berkala terhadap progres proyek, serta penerapan metodologi manajemen proyek yang efektif.

Anggaran yang membengkak dapat disebabkan oleh biaya tambahan dalam pembelian komponen perangkat keras, lisensi perangkat lunak, atau kebutuhan pengujian yang lebih luas dari yang diperkirakan. Untuk mengurangi risiko ini, dilakukan pengelolaan anggaran yang ketat dengan menyusun estimasi biaya secara rinci serta menyisihkan dana cadangan untuk keperluan tak terduga. Kegagalan teknologi, seperti kesalahan dalam pengolahan data sensor atau kegagalan sistem otomatisasi pemberian pakan, dapat menghambat fungsi utama proyek. Untuk mengatasi risiko ini, dilakukan pengujian perangkat keras dan perangkat lunak secara bertahap serta pemilihan komponen yang telah teruji kualitasnya.

## **BAB III**

### **PRODUCT DESIGN (PD)**

#### **(DESAIN PENGEMBANGAN PRODUK)**

##### **3.1. PENDAHULUAN**

Bagian ini membahas perancangan desain produk Smart Aquarium Ikan Hias, yang merupakan sistem otomatis untuk memonitor kualitas air dan memberikan pakan ikan secara otomatis. Desain ini dirancang untuk memenuhi kebutuhan pengguna dalam memelihara ikan hias dengan lebih efisien, mengurangi intervensi manual, dan menjaga kualitas air tetap optimal.

Tujuan utama dari perancangan produk ini adalah menciptakan sistem yang dapat:

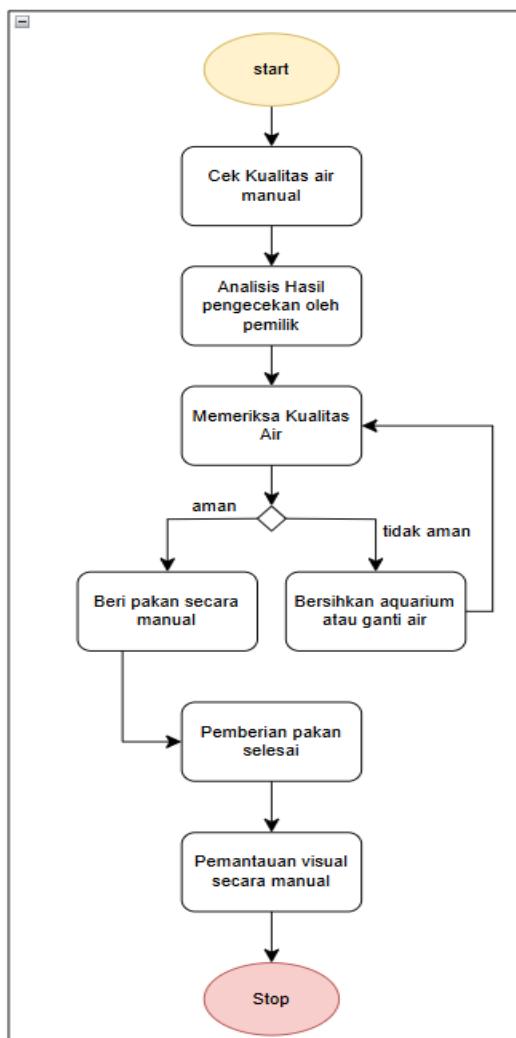
- Memantau parameter kualitas air secara real-time, termasuk pH, kekeruhan, dan kadar TDS, guna memastikan lingkungan yang sehat bagi ikan.
- Mengotomatiskan pemberian pakan, sehingga pemilik akuarium tidak perlu khawatir akan jadwal pemberian makan ikan.
- Menampilkan dan mengontrol sistem melalui website, memungkinkan pemantauan jarak jauh menggunakan smartphone.

Desain produk ini mempertimbangkan aspek teknologi yang sesuai dengan kebutuhan pengguna serta tujuan proyek. Pendekatan yang digunakan dalam perancangan melibatkan pemilihan komponen hardware dan software yang tepat, integrasi sensor, dan implementasi algoritma kendali berbasis ESP32.

Pada bab ini, akan dibahas secara rinci tentang desain langsung dari proyek Smart Aquarium Ikan Hias, mencakup aspek teknis, skema rangkaian, serta bagaimana sistem ini dapat diimplementasikan secara efektif.

## 3.2. DESKRIPSI PRODUK DESIGN

### 3.2.1. Current System

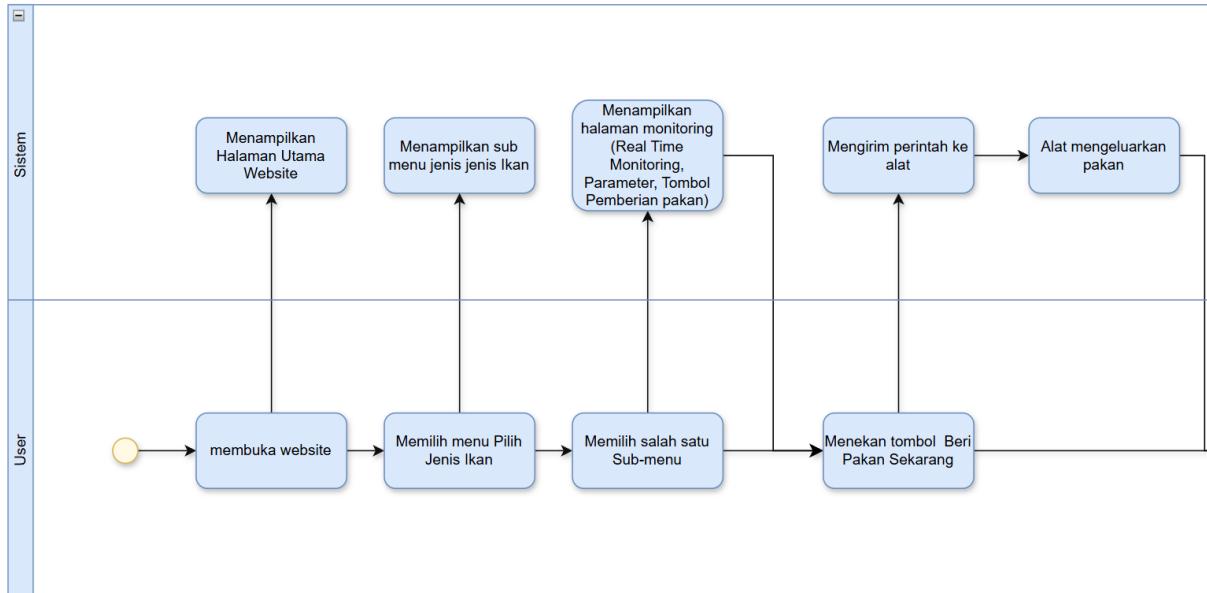


Gambar 3. 1 Current System

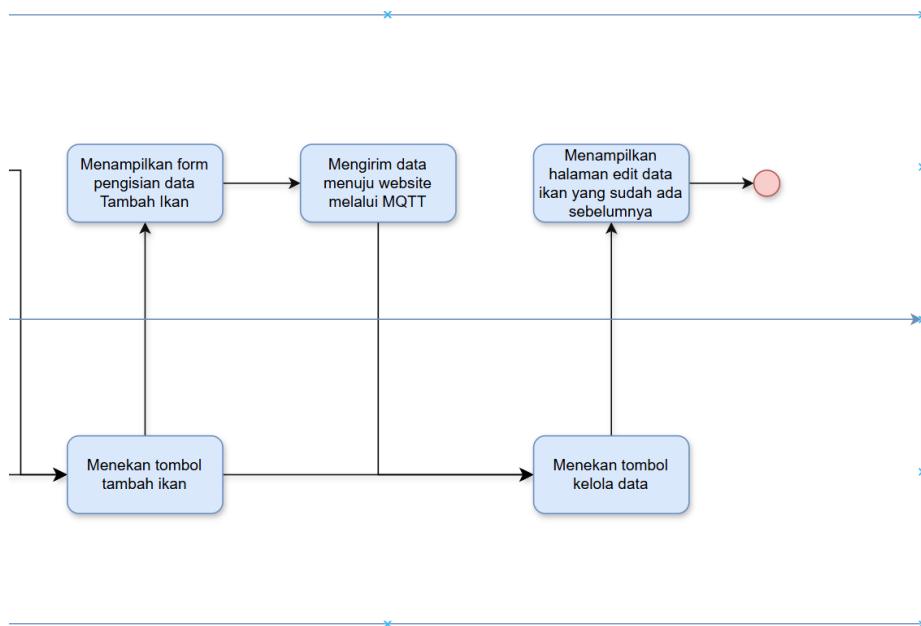
Flowchart yang dibuat menggambarkan sistem pemeliharaan akuarium secara manual, di mana pemilik melakukan pengecekan kualitas air seperti pH, suhu, dan kekeruhan tanpa alat otomatis. Hasil pengecekan dianalisis sendiri, lalu diputuskan apakah kondisi air aman atau tidak. Jika aman, ikan diberi pakan secara manual, sedangkan jika tidak aman, pemilik membersihkan atau mengganti air akuarium terlebih dahulu. Setelah pemberian pakan selesai, pemantauan kondisi ikan tetap dilakukan secara visual tanpa bantuan teknologi. Sistem ini menunjukkan keterbatasan dalam efisiensi dan akurasi, sehingga diperlukan solusi otomatis berbasis IoT.

### 3.2.2. Proses Bisnis Target System

Menggambarkan alur kerja atau proses bisnis utama yang akan didukung oleh sistem. Berisi flowchart atau Business Process Model and Notation (BPMN) untuk memvisualisasikan proses yang akan diotomasi atau dioptimasi. Relevan untuk menjelaskan bagaimana sistem akan berinteraksi dengan pengguna atau proses lain di dunia nyata.



Gambar 3. 2 Proses bisnis target system dari Smart Aquarium Ikan Hias



Gambar 3. 3 Proses bisnis target system dari Smart Aquarium Ikan Hias

Proses bisnis pada sistem ini dimulai ketika pengguna membuka website dan memilih menu “Pilih Jenis Ikan”. Sistem kemudian menampilkan submenu jenis-jenis ikan yang tersedia. Setelah pengguna memilih salah satu jenis ikan, sistem akan menampilkan halaman monitoring yang berisi data real-time seperti suhu, pH, dan kejernihan air, serta tombol untuk memberi

pakan. Jika tombol “Beri Pakan Sekarang” ditekan, sistem akan mengirimkan perintah ke alat IoT untuk mengeluarkan pakan. Jika ikan yang dicari tidak tersedia, pengguna dapat menambahkan jenis ikan melalui menu “Tambah Ikan” dan mengisi form data. Sistem menyimpan data tersebut ke MQTT, dan pengguna dapat mengelola atau mengedit data ikan melalui menu “Kelola Data”. Seluruh proses dilakukan langsung dari website tanpa login, dengan interaksi aktif antara pengguna dan sistem secara real-time.

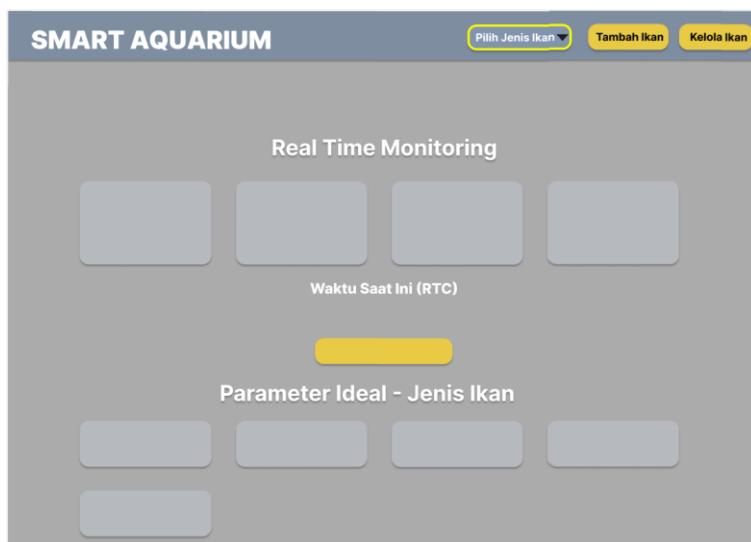
### 3.2.3. User Interface Layout [Software]

Desain dibawah ini menggunakan figma dan menampilkan data real-time dari berbagai sensor kualitas air, seperti pH, TDS, suhu, dan kekeruhan, yang kemudian dibandingkan secara otomatis dengan data kebutuhan ideal dari jenis ikan yang tersedia.

Pengguna juga diberikan fitur untuk menambah, mengedit, dan menghapus data jenis ikan sesuai kebutuhan.



Gambar 3. 4 Desain Website Halaman Utama



Gambar 3. 5 Desain Website setelah memilih salah satu menu Ikan

**SMART AQUARIUM**

Tambah Jenis Ikan Baru

Pilih Jenis Ikan ▾ Tambah Ikan Kelola Ikan

Nama Ikan:

Kode Ikan (tanpa spasi, huruf kecil):

Rentang :

Rentang :

Rentang :

Waktu Pakan:

Real Time Monitoring

Waktu Saat Ini (RTC)

Parameter Ideal - Jenis Ikan

Tambah Ikan

Waktu Pakan

Gambar 3. 6 Desain Website untuk Menambah Data Ikan Baru

**SMART AQUARIUM**

Kelola Jenis Ikan

Pilih Jenis Ikan ▾ Tambah Ikan Kelola Ikan

Real Time Monitoring

Waktu Saat Ini (RTC)

Parameter Ideal - Jenis Ikan

Gambar 3. 7 Desain Website untuk Mengelola Data Ikan

**SM Edit Jenis Ikan ARIUM**

Pilih Jenis Ikan ▾ Tambah Ikan Kelola Ikan

Nama Ikan:

Kode Ikan (tanpa spasi, huruf kecil):

Rentang :

Rentang :

Rentang :

Waktu Pakan:

Real Time Monitoring

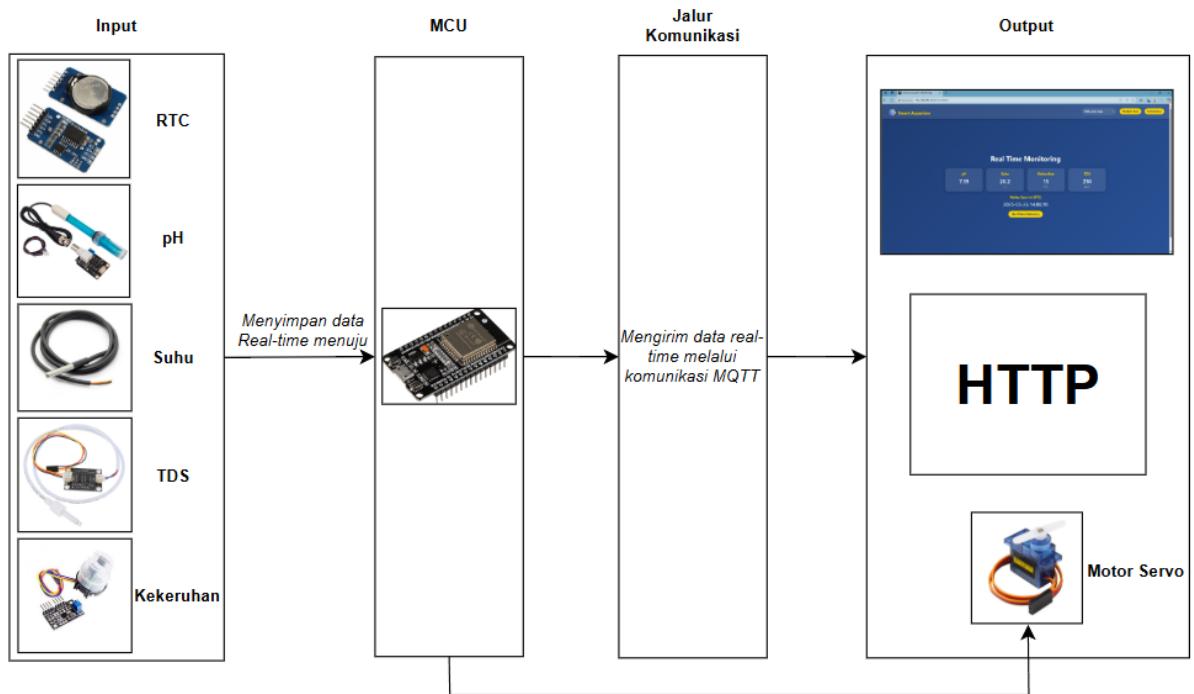
Waktu Saat Ini (RTC)

Parameter Ideal - Jenis Ikan

Simpan Perubahan

Gambar 3. 8 Desain Website untuk Mengedit Data Ikan

### 3.2.4. Arsitektur Sistem [Hardware]



Gambar 3. 9 Arsitektur Sistem

#### 1) Input (Sensor dan RTC)

Pada sisi kiri gambar, terdapat beberapa perangkat input yang bertugas mengumpulkan data dari akuarium secara real-time:

- **RTC (Real Time Clock):** Menyediakan informasi waktu yang akurat, digunakan untuk menjadwalkan pemberian pakan otomatis secara tepat waktu.
- **Sensor pH:** Mendeteksi tingkat keasaman atau kebasaan air.
- **Sensor Suhu:** Mengukur suhu air secara kontinu.
- **Sensor TDS:** Mengukur kadar zat terlarut dalam air, seperti mineral atau sisa pakan.
- **Sensor Kekeruhan (Turbidity):** Mengukur kejernihan atau tingkat kotoran dalam air akuarium.

#### 2) ESP32 (Mikrokontroler)

ESP32 berfungsi sebagai pusat kendali sistem. Tugas utamanya adalah:

- Membaca data dari semua sensor dan modul RTC.
- Menyimpan dan mengirimkan data real-time melalui jaringan WiFi.
- Mengontrol motor servo untuk membuka tempat pakan sesuai jadwal atau perintah pengguna.
- Mengirim data ke website monitoring melalui protokol MQTT (HiveMQ).

### 3) Motor Servo

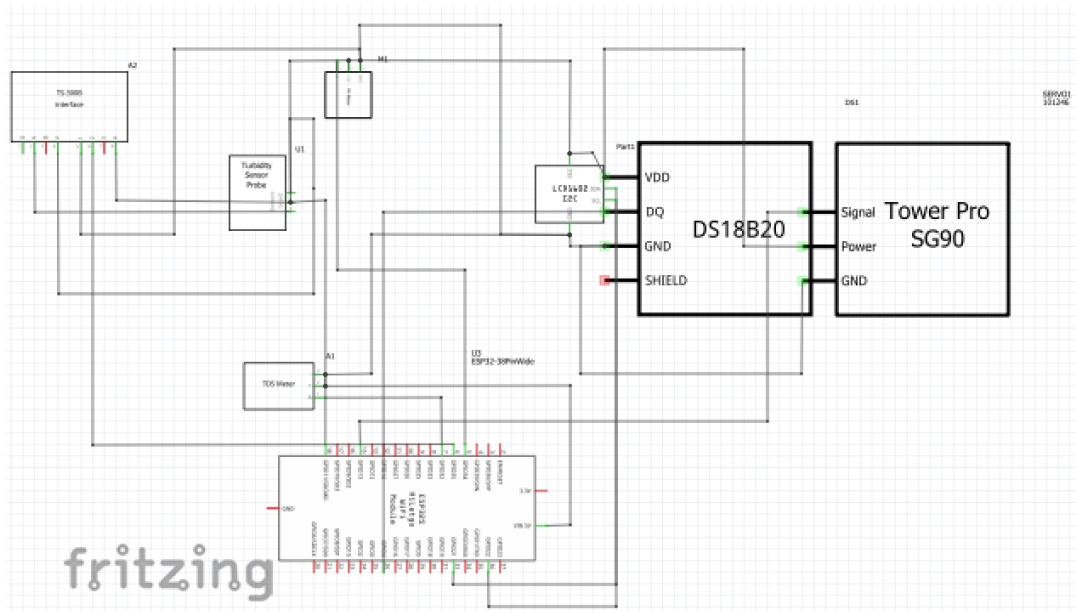
Motor servo menerima perintah dari ESP32 untuk mengaktifkan mekanisme pemberian pakan ikan. Motor ini akan membuka dan menutup tempat pakan secara otomatis berdasarkan jadwal dari RTC atau perintah manual dari website.

### 4) Komunikasi ke Website dan HTTP

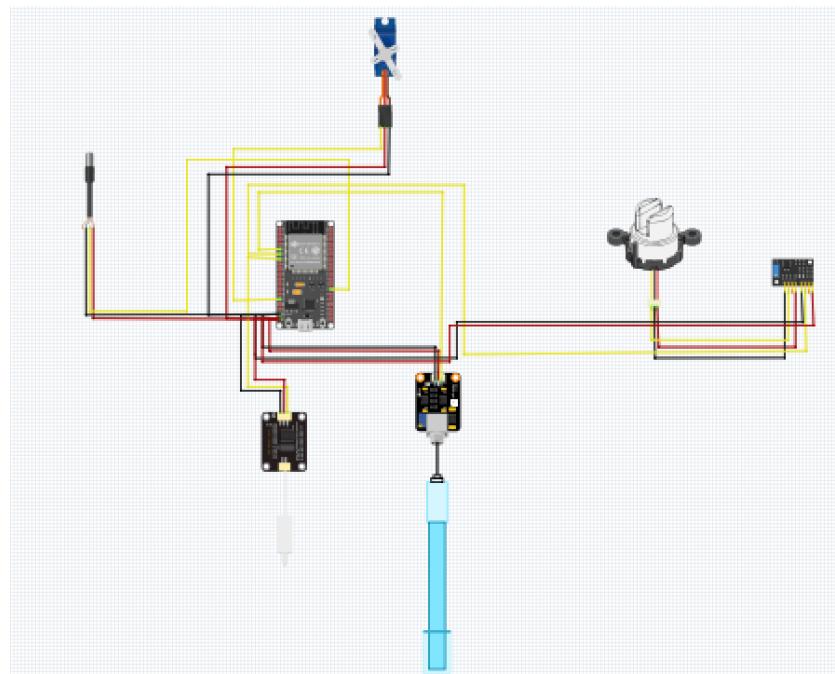
Data dari ESP32 dikirim ke server menggunakan protokol MQTT, kemudian ditampilkan secara real-time di website monitoring. Website ini dibangun menggunakan teknologi berbasis HTTP, yang memungkinkan pengguna melihat:

- Nilai suhu, pH, TDS, dan kekeruhan air.
- Status pemberian pakan otomatis.
- Kontrol manual untuk motor servo.

#### 3.2.5. Desain Rangkaian Elektronik/Skematik Desain dan Desain PCB [Hardware]

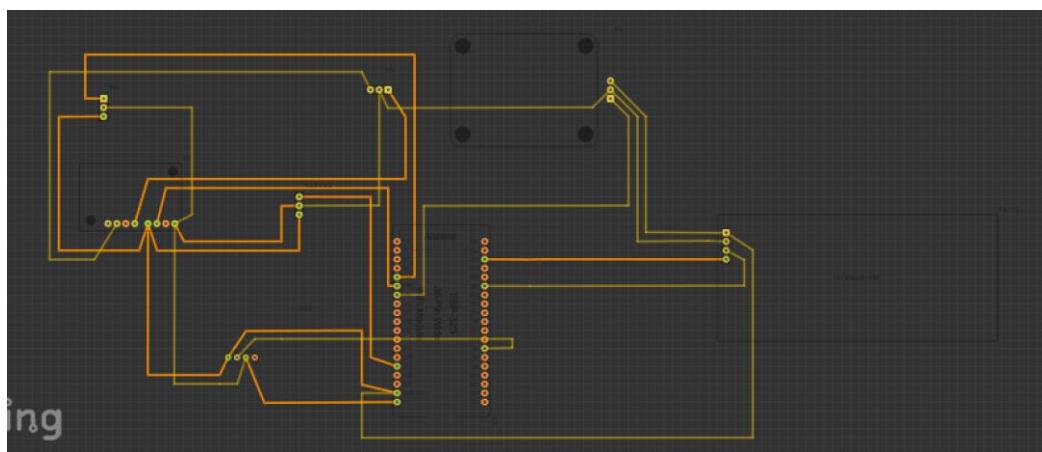


Gambar 3. 10 Skematik Desain Hardware

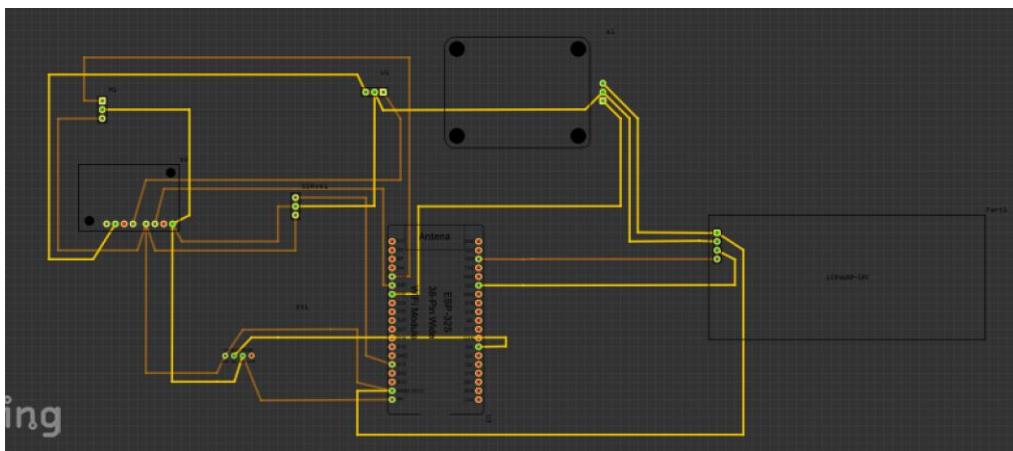


*Gambar 3. 11 Desain Skematik Hardware*

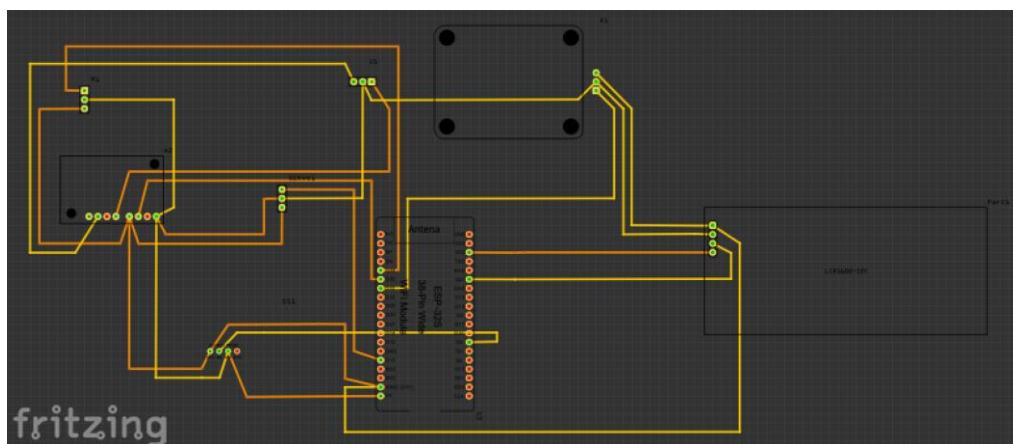
Skematik ini menunjukkan sistem Smart Aquarium Ikan Hias berbasis ESP32 yang mengotomatisasi pemantauan dan pemeliharaan kualitas air. Sistem ini menggunakan berbagai sensor seperti DS18B20 untuk mengukur suhu, TDS meter untuk mengevaluasi kualitas air, Turbidity Sensor untuk mendeteksi kekeruhan. Data dari sensor diproses oleh ESP32. Jika kualitas air memburuk, sistem dapat memberikan peringatan atau mengaktifkan mekanisme perbaikan otomatis.



*Gambar 3. 12 Desain PCB Bawah*



Gambar 3. 13 Desain PCB Atas



Gambar 3. 14 Desain PCB tampilan atas bawah

Desain PCB di atas menunjukkan tata letak jalur sirkuit yang menghubungkan berbagai komponen dalam sistem monitoring kualitas air dan pemberian pakan otomatis pada akuarium ikan hias. PCB ini dirancang dengan jalur yang dioptimalkan untuk mengurangi noise dan interferensi, memastikan koneksi yang stabil antara sensor, mikrokontroler, dan aktuator. Desain ini kemungkinan menggunakan software seperti Fritzing, dengan teknologi *Through-Hole* (THT) dan kemungkinan beberapa komponen *Surface-Mount* (SMD) untuk efisiensi ruang. Tata letak ini bertujuan untuk mempermudah proses fabrikasi dan perakitan komponen secara efektif.

### 3.2.6. Mekanisme Komunikasi Data [Hardware]

Berikut adalah penjelasan mengenai gambar rangkaian sistem monitoring kualitas air dan pemberian pakan otomatis yang menggunakan ESP32 sebagai mikrokontroler utama:

#### 1) Mikrokontroler (ESP32-WROOM)

- ESP32 berfungsi sebagai pusat pemrosesan data, di mana ia menerima input dari berbagai sensor dan mengontrol perangkat output.

- Terhubung dengan sensor pH, sensor kekeruhan, dan komponen lainnya.

2) Sensor Input

- Sensor pH (bagian bawah kanan) digunakan untuk mengukur tingkat keasaman atau kebasaan air.
- Sensor kekeruhan (Turbidity Sensor, bagian kiri bawah) mendeteksi kejernihan air dengan membaca hamburan cahaya dalam air.
- Sensor TDS (Total Dissolved Solids) (bagian kiri tengah) mengukur jumlah partikel terlarut dalam air, seperti garam dan mineral.

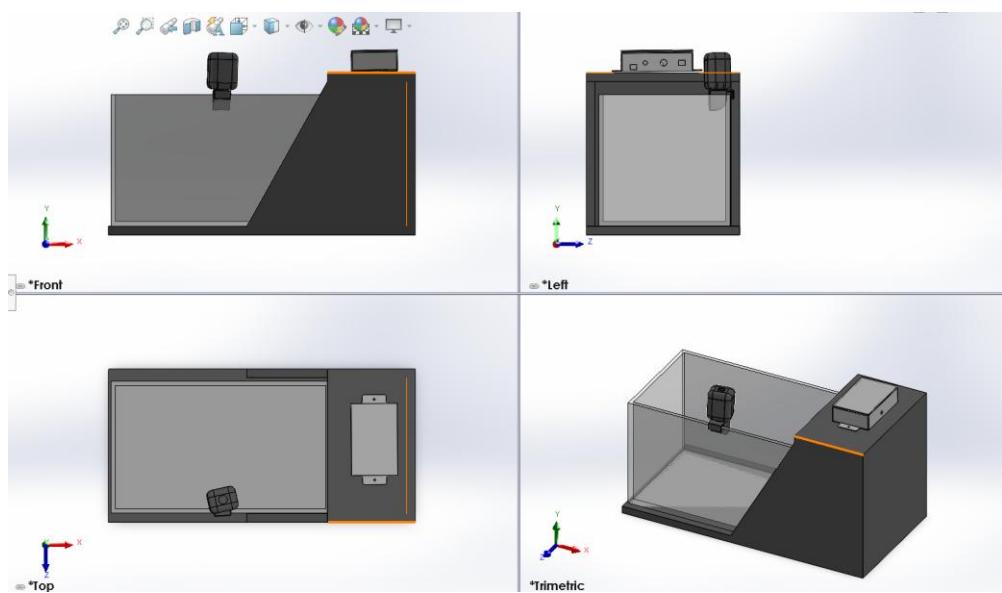
3) Sistem Koneksi dan Daya

- Setiap sensor dan komponen output mendapatkan daya dari ESP32 (VCC dan GND) serta terhubung melalui pin GPIO untuk komunikasi data.
- Jalur hijau pada gambar menunjukkan koneksi antara sensor, ESP32, dan perangkat output.

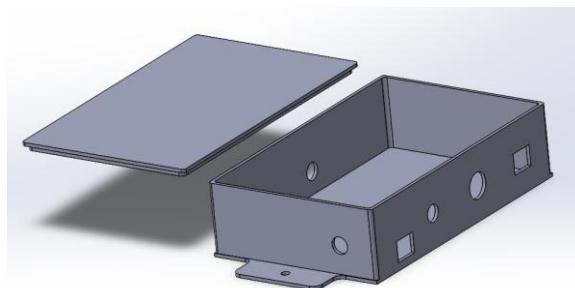
4) Alur Kerja Sistem

- ESP32 membaca data sensor dan menentukan apakah kualitas air masih dalam batas aman.
- Jika terjadi penyimpangan pada pH, kekeruhan, atau TDS, sistem akan menampilkan peringatan di halaman website nya.
- Data juga dikirim ke MQTT dan muncul di halaman website untuk dipantau dari jarak jauh melalui smartphone.

### 3.2.7. Desain Fisik/Case Hardware



Gambar 3. 15 Case Hardware



Gambar 3. 16 Desain Case PCB

Berikut adalah penjelasan mengenai desain fisik (hardware case) dari proyek Smart Aquarium Ikan Hias. Desain ini dibuat untuk memberikan perlindungan terhadap komponen elektronik, sekaligus mempertimbangkan kemudahan instalasi, perawatan, serta estetika tampilan keseluruhan sistem. Adapun rincian desain fisik sistem adalah sebagai berikut:

- Material Case:  
Case dibuat menggunakan filamen PLA (Polylactic Acid) yang dicetak menggunakan teknologi 3D printing. Material ini dipilih karena ringan, mudah dibentuk, dan cukup kuat untuk kebutuhan indoor, serta dapat memberikan perlindungan terhadap kelembapan jika dilapisi dengan finishing tahan air.
- Penempatan Case PCB:  
Case yang berfungsi untuk melindungi PCB dari air dan debu akan ditempatkan di atas case kayu akuarium. Lokasi ini dipilih agar komponen elektronik seperti ESP32, sensor, dan power supply berada jauh dari area yang berpotensi basah, sekaligus mudah dijangkau saat perawatan atau pemrograman ulang.

- Case Tempat Pakan Otomatis:

Untuk modul pemberi pakan otomatis yang dikendalikan oleh servo, tempat pakan ditempel langsung di atas kaca akuarium, dengan posisi lubang pemberi pakan menghadap ke dalam air. Hal ini bertujuan agar pakan langsung jatuh ke akuarium secara presisi tanpa terkena air sebelum waktunya.

- Routing Kabel dan Port Akses:

Desain case juga memperhitungkan routing kabel sensor, dengan jalur khusus dan lubang kedap air agar kabel sensor suhu (DS18B20), sensor TDS, dan sensor kekeruhan dapat keluar dan masuk dengan aman ke akuarium tanpa menimbulkan celah bocor.

- Akses Perawatan Mudah:

Terdapat tutup atau penutup case yang dapat dibuka-tutup dengan mudah, memungkinkan pengguna melakukan pengecekan, perawatan, atau penggantian komponen tanpa membongkar seluruh sistem.

## **BAB IV**

### **PRODUCT IMPLEMENTATION (PI)**

#### **(IMPLEMENTASI PENGEMBANGAN PRODUK)**

##### **4.1.PENDAHULUAN**

Implementasi produk Smart Aquarium Ikan Hias dilakukan berdasarkan acuan dari dokumen *Product Design* yang telah disusun sebelumnya. Dokumen desain tersebut berisi rancangan sistem secara menyeluruh, mulai dari arsitektur perangkat keras dan lunak, diagram alur data, hingga antarmuka pengguna yang direncanakan. Seluruh proses pengembangan dalam tahap implementasi mengacu pada rancangan tersebut agar sistem yang dibangun sesuai dengan kebutuhan pengguna dan fungsi yang diharapkan.

Tujuan dari tahap implementasi ini adalah untuk merealisasikan semua spesifikasi teknis yang telah ditentukan ke dalam bentuk sistem yang berfungsi secara nyata. Proses implementasi dilakukan secara bertahap dan sistematis, dimulai dari penyusunan kode program, perakitan perangkat keras, integrasi antar komponen, hingga pengujian dan validasi sistem secara keseluruhan.

Lingkup implementasi meliputi pemrograman mikrokontroler (ESP32), pengolahan data sensor (pH, TDS, suhu, dan kekeruhan), pengiriman data melalui protokol MQTT, pengembangan antarmuka website, serta pengoperasian aktuator berupa motor servo untuk pemberian pakan otomatis.

Secara garis besar, tahapan implementasi dalam proyek ini meliputi:

- Pengkodean: Penulisan dan pengujian program untuk ESP32 serta sistem backend website.
- Perakitan dan integrasi: Penggabungan sensor, aktuator, dan mikrokontroler sesuai dengan rancangan PCB.
- Pengujian sistem: Melakukan uji coba untuk memastikan seluruh fungsi bekerja sesuai dengan desain dan kebutuhan pengguna.

Dengan pendekatan ini, implementasi produk dilakukan secara efisien dan terarah, serta mampu menghasilkan sistem Smart Aquarium yang andal, mudah digunakan, dan bermanfaat bagi pemilik ikan hias.

## **4.2.DESKRIPSI**

Proses implementasi produk Smart Aquarium Ikan Hias dilakukan secara bertahap dan terstruktur, dimulai dari pengembangan sistem perangkat keras (hardware), sistem perangkat lunak (software), hingga integrasi keduanya melalui koneksi IoT berbasis protokol MQTT.

Implementasi perangkat keras dimulai dengan pemasangan sensor-sensor utama seperti sensor pH, TDS, suhu, dan kekeruhan pada akuarium. Semua sensor ini terhubung ke mikrokontroler ESP32 yang berfungsi sebagai pusat kendali sistem. Selain sensor, komponen aktuator seperti motor servo juga dihubungkan untuk mengatur pemberian pakan otomatis berdasarkan waktu yang telah dijadwalkan.

Dari sisi perangkat lunak, program utama ditanamkan di ESP32 menggunakan Arduino IDE. Program ini berfungsi membaca data dari sensor secara berkala, mengolah data tersebut, dan mengirimkannya ke server melalui koneksi WiFi menggunakan protokol MQTT. Server yang digunakan akan menyimpan dan menampilkan data secara real-time melalui antarmuka website. Website dikembangkan menggunakan Visual Studio Code dan menampilkan informasi sensor serta peringatan jika kondisi air tidak sesuai standar.

Seluruh proses implementasi didasarkan pada prinsip-prinsip berikut:

- Modularitas: Setiap bagian sistem dipisahkan menjadi modul (sensor, komunikasi, kontrol pakan) agar mudah diuji dan diperbaiki.
- Efisiensi: Penggunaan sumber daya mikrokontroler dioptimalkan agar tidak terjadi beban berlebih.
- Keamanan data: Komunikasi data antara perangkat dan server dilakukan dengan koneksi terenkripsi jika memungkinkan.
- Keterandalan sistem: Pengujian dilakukan secara menyeluruh untuk memastikan sistem dapat bekerja stabil dalam jangka waktu lama.

#### **4.2.1. Prinsip Implementasi**

Implementasi sistem Smart Aquarium Ikan Hias dilakukan berdasarkan prinsip-prinsip pengembangan yang bertujuan untuk menjamin efisiensi, keandalan, dan keamanan sistem. Produk ini dikembangkan dengan pendekatan sistematis dan terarah agar dapat berfungsi secara optimal serta mudah dirawat dan dikembangkan di masa depan.

Metodologi Implementasi:

Metodologi yang digunakan dalam pengembangan proyek ini adalah metode *Agile-Scrum*, yang terdiri dari beberapa tahapan berurutan mulai dari analisis kebutuhan, desain sistem, implementasi perangkat keras dan perangkat lunak, pengujian, hingga tahap integrasi dan dokumentasi. Pemilihan metode *Agile-Scrum* didasarkan pada kejelasan kebutuhan sistem dan keterbatasan ruang lingkup proyek, yang berubah-ubah secara signifikan selama proses pengembangan.

Prinsip-Prinsip Implementasi:

Beberapa prinsip utama yang diterapkan selama implementasi adalah sebagai berikut:

- Modularitas: Sistem dikembangkan secara modular, di mana komponen seperti pembacaan sensor, pengiriman data, serta kendali pemberian pakan diatur dalam blok kode atau modul yang terpisah. Hal ini mempermudah proses pengujian, perbaikan, dan pengembangan lebih lanjut.
- Efisiensi: Kode program dan konfigurasi perangkat keras ditulis dan dirancang dengan memperhatikan efisiensi daya dan pemrosesan, mengingat ESP32 sebagai mikrokontroler memiliki keterbatasan memori dan daya.
- Keamanan dan Keandalan: Untuk memastikan sistem aman digunakan, komunikasi data dilakukan menggunakan protokol MQTT yang handal dan ringan. Selain itu, sistem dilengkapi dengan validasi parameter untuk mencegah kesalahan pembacaan sensor atau pengendalian perangkat yang tidak sesuai.
- Kemudahan Pemeliharaan: Dokumentasi dilakukan dengan rapi di setiap bagian kode dan skema perangkat keras agar mempermudah pengembang lain dalam memahami sistem.

Pedoman Pengembangan:

1) Pedoman Pemrograman:

- Bahasa pemrograman yang digunakan adalah C++ untuk perangkat mikrokontroler ESP32 menggunakan Arduino IDE, dan JavaScript/PHP untuk bagian web.
- Kode ditulis menggunakan standar clean code, yaitu ringkas, terbaca, dan mudah dipahami.

- Menggunakan library standar dari Arduino yang kompatibel dengan sensor dan modul yang digunakan.

2) Pedoman Perangkat Keras:

- Penempatan komponen pada PCB dirancang agar minim noise dan interferensi antar sensor.
- Pemasangan komponen elektronik dilakukan secara manual dengan teknik soldering yang presisi untuk memastikan koneksi optimal.
- Tegangan dan arus diperhatikan agar sesuai dengan spesifikasi sensor dan mikrokontroler.

Dengan prinsip-prinsip ini, proses implementasi proyek Smart Aquarium Ikan Hias dapat dilakukan secara sistematis, terstruktur, dan menghasilkan sistem yang stabil, fungsional, serta siap digunakan oleh pengguna akhir.

#### **4.2.2. Lingkungan Pengembangan Terintegrasi (IDE)**

Dalam pengembangan produk Smart Aquarium Ikan Hias, digunakan beberapa lingkungan pengembangan terintegrasi (IDE) untuk mendukung proses implementasi perangkat keras dan lunak. IDE yang digunakan dipilih berdasarkan kesesuaian dengan kebutuhan sistem dan kemudahan integrasi dengan komponen-komponen pendukung seperti sensor, mikrokontroler, serta komunikasi berbasis MQTT.

1) Arduino IDE

- Alasan Pemilihan:

Arduino IDE dipilih karena sangat kompatibel dengan board mikrokontroler ESP32 yang digunakan dalam proyek ini. Selain itu, IDE ini menyediakan antarmuka sederhana yang mudah digunakan, terutama untuk mengakses dan memprogram sensor seperti pH, TDS, turbidity, dan mengontrol servo motor untuk pemberian pakan otomatis.

- Fitur Utama yang Dimanfaatkan:

- **Serial Monitor:** Untuk melihat data sensor secara langsung saat pengujian.
- **Library Manager:** Memudahkan instalasi dan manajemen pustaka seperti WiFi, sensor, dan MQTT.
- **Board Manager:** Memungkinkan instalasi dukungan ESP32.
- **Auto-completion dan Highlighting:** Mempercepat proses penulisan kode dan meminimalkan kesalahan.

## 2) Visual Studio Code (VS Code)

- Alasan Pemilihan:

VS Code digunakan untuk pengembangan perangkat lunak berbasis website. Website ini berfungsi untuk menampilkan data real-time dari sensor dan memberikan peringatan jika parameter air berada di luar batas normal. VS Code dipilih karena fleksibel, ringan, dan mendukung banyak bahasa pemrograman serta integrasi dengan Git dan alat kolaborasi lain.

- Fitur Utama yang Dimanfaatkan:

- Code Formatting dan Auto-completion
- Live Server untuk melihat pratinjau antarmuka web secara real-time.
- Integrasi dengan JavaScript, HTML, dan CSS
- Plugin MQTT (untuk kebutuhan simulasi atau integrasi data real-time)

## 3) Fritzing

- Alasan Penggunaan:

Fritzing digunakan untuk membuat diagram rangkaian sistem (skematik) yang melibatkan ESP32, sensor, dan aktuator. Software ini memudahkan visualisasi hubungan antar komponen secara elektronik sebelum dilakukan perakitan fisik.

- Fitur Utama:

- Pembuatan Breadboard View untuk penyusunan komponen.
- Skema Sirkuit yang rapi dan mudah dipahami.
- Export Gambar untuk dokumentasi dan presentasi sistem.

## 4) Draw.io

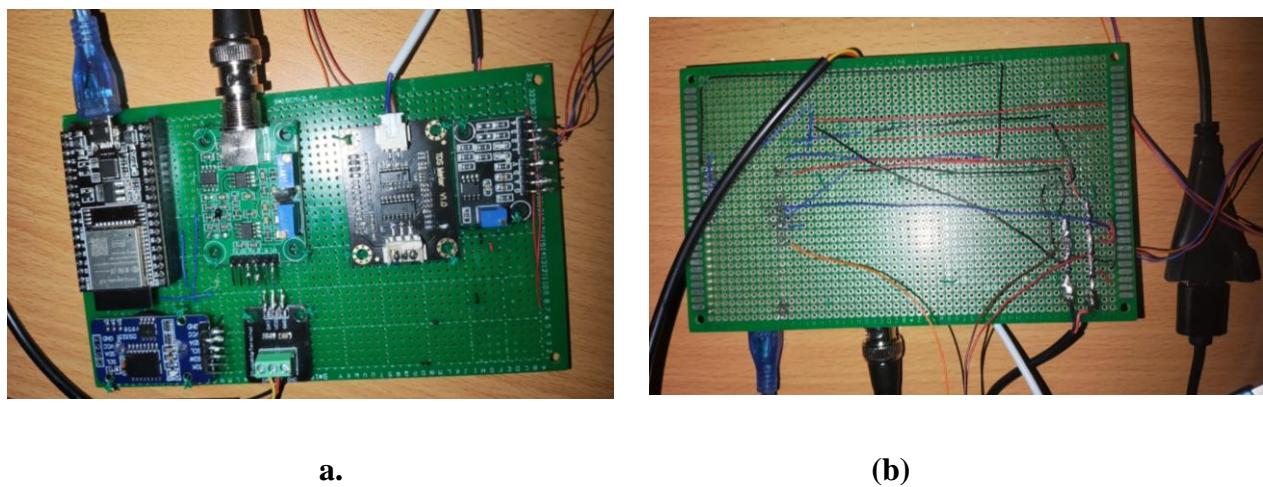
- Alasan Penggunaan:

Draw.io digunakan untuk membuat diagram konteks dan arsitektur sistem secara keseluruhan. Diagram ini memberikan gambaran hubungan antara pengguna, perangkat keras, server MQTT, dan tampilan website.

- Fitur Utama:

- Diagram Konteks dan Alur Data
- Custom Icon IoT dan IT System
- Kolaborasi Cloud via Google Drive

#### 4.2.3. Implementasi Desain PCB



a.

(b)

Gambar 4. 1 Implementasi Desain PCB tampak atas dan bawah

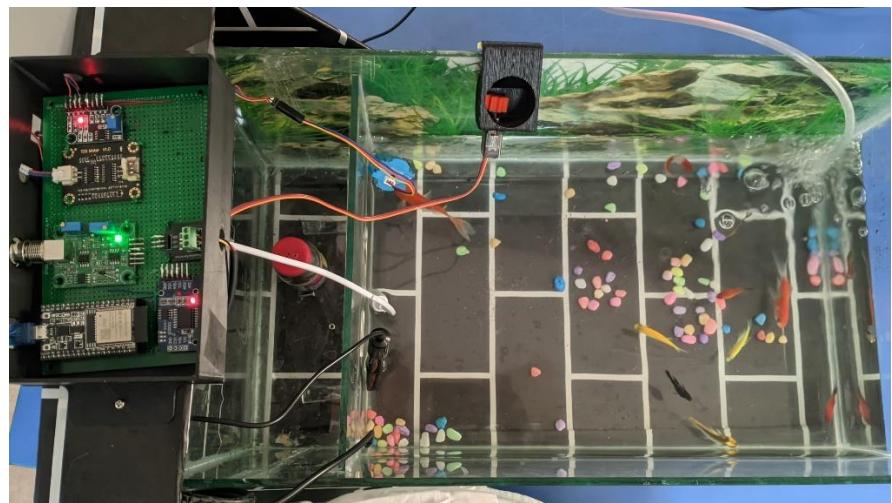
Proses implementasi desain PCB dalam proyek Smart Aquarium Ikan Hias dimulai dengan pembuatan layout sirkuit berdasarkan rancangan sistem yang telah direncanakan. Setelah desain selesai, PCB diproduksi melalui layanan pabrikasi untuk memastikan presisi jalur dan koneksi antar komponen. Tahap berikutnya adalah pemasangan komponen elektronik menggunakan teknik manual soldering, di mana setiap komponen, seperti mikrokontroler, sensor, dan modul pendukung, dipasang dengan hati-hati untuk memastikan koneksi yang optimal. Setelah semua komponen terpasang, dilakukan pengecekan dan pengujian rangkaian guna memastikan bahwa PCB berfungsi sesuai dengan spesifikasi yang telah ditentukan.

#### 4.2.4. Implementasi Desain Fisik/Desain Case Hardware

Implementasi desain fisik atau casing hardware pada proyek Smart Aquarium Ikan Hias dilakukan untuk melindungi komponen elektronik serta memastikan sistem dapat beroperasi dengan baik dalam lingkungan akuarium.

- 1) Proses Desain dan Realisasi Casing: Desain casing dibuat menggunakan perangkat lunak pemodelan 3D dan direalisasikan melalui teknik 3D printing untuk menghasilkan bentuk yang presisi dan sesuai dengan kebutuhan perangkat.
- 2) Material yang Digunakan: Casing dibuat menggunakan plastik ABS atau akrilik, yang tahan terhadap air dan lingkungan akuarium. Bahan ini dipilih karena sifatnya yang ringan, kuat, serta mudah dibentuk.
- 3) Proses Finishing: Setelah pencetakan, casing mengalami proses finishing seperti penghalusan permukaan dan pelapisan anti-air untuk meningkatkan daya tahan terhadap kelembapan dan suhu air. Jika diperlukan, bagian casing yang bersentuhan

dengan air akan diberikan sealant untuk mencegah kebocoran dan melindungi komponen di dalamnya.



Gambar 4. 2 Implementasi Hardware

#### 4.2.5. Integrasi Hardware dan Software

Integrasi hardware dan software dalam proyek Smart Aquarium Ikan Hias bertujuan untuk memastikan bahwa seluruh komponen dapat beroperasi secara sinkron dan sesuai dengan desain sistem.

- 1) Langkah-Langkah Penghubungan Perangkat Keras dan Perangkat Lunak

Mikrokontroler ESP32 diprogram menggunakan Arduino IDE, yang memungkinkan komunikasi dengan sensor (pH, suhu, kekeruhan, TDS) serta aktuator seperti motor servo untuk pemberian pakan. Data dari sensor dikirim dengan komunikasi MQTT yang akan terhubung langsung dengan website, sehingga pengguna dapat memantau kondisi air secara real-time melalui website yang tersedia.

- 2) Pengujian Awal

Sebelum sistem diterapkan secara penuh, dilakukan pengujian konektivitas antara sensor, mikrokontroler, dan web. Setiap sensor diuji secara individual untuk memastikan bahwa data yang dikirimkan sesuai dengan kondisi sebenarnya.

- 3) Debugging Masalah Integrasi

Jika terjadi kesalahan dalam komunikasi hardware dan software, debugging dilakukan dengan cara monitoring serial pada Arduino IDE. Masalah yang sering muncul, seperti ketidaksesuaian nilai sensor atau keterlambatan respons aktuator, diatasi dengan penyempurnaan kode serta pengaturan ulang parameter komunikasi antar-komponen.

## Code ESP32:

```
#include <WiFi.h>
#include <PubSubClient.h>
#include <WiFiClientSecure.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#include <RTClib.h>
#include <ESP32Servo.h>
#include <ArduinoJson.h> // Tambahkan library ArduinoJson

// --- Konfigurasi WiFi ---
const char* ssid = "POCO F4";
const char* password = "11111111";

// --- Konfigurasi MQTT (HiveMQ Cloud) ---
const char* mqtt_server =
"069dadc53c054a52896b582b7ad31fbb.s1.eu.hivemq.cloud";
const int mqtt_port = 8883;
const char* mqtt_user = "Hanbok9";
const char* mqtt_password = "Handika09";

// --- Definisi Pin ---
#define PH_PIN 32
#define TURBIDITY_PIN 35
#define TDS_PIN 34
#define ONE_WIRE_BUS 33
#define SERVO_PIN 26

// --- Konstanta Kalibrasi Sensor ---
const float VREF = 3.3;           // Tegangan referensi ESP32 (ADC 12-bit)
// Konstanta kalibrasi pH kuadratik: pH = a * V2 + b * V + c
const float COEF_A = 2.925;       // Koefisien V2
const float COEF_B = -21.341;      // Koefisien V
const float COEF_C = 41.931;      // Intercept
const float TURBIDITY_MIN = 0;     // Nilai ADC untuk air jernih (0 NTU)
const float TURBIDITY_MAX = 2800; // Nilai ADC untuk air sangat keruh (5 NTU)
const float TDS_OFFSET = 0.04;    // Tegangan offset TDS (dari air suling, tidak digunakan)
const float TDS_FACTOR = 2727.0; // Faktor kalibrasi TDS (dari larutan 1000 ppm, tidak digunakan)

// --- Konstanta untuk Kalibrasi pH ---
const int SAMPLES = 50;          // Jumlah sampel untuk rata-rata
const int MEDIAN_SAMPLES = 10;    // Jumlah sampel untuk filter median

// --- Variabel Sensor ---
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);

float nilai_pH = 0.0;
float suhu = 0.0;
float kekeruhan = 0.0;
float tdsValue = 0.0;

// --- Objek RTC dan Servo ---
RTC_DS3231 rtc;
Servo feedingServo;

// --- Struktur untuk Parameter Ikan ---
```

```

struct FishProfile {
    const char* name; // Nama ikan
    float phMin; // Batas bawah pH
    float phMax; // Batas atas pH
    float suhuMin; // Batas bawah suhu (°C)
    float suhuMax; // Batas atas suhu (°C)
    float kekeruhanMax; // Batas maksimum kekeruhan (NTU)
    float tdsMin; // Batas bawah TDS (ppm)
    float tdsMax; // Batas atas TDS (ppm)
    int feedingCount; // Jumlah pakan per hari
    String feedingTimes[4]; // Jadwal pakan (maksimal 4 kali per hari)
};

// --- Daftar Profil Ikan ---
FishProfile fishProfiles[] = {
    { "koki", 6.0, 8.0, 20.0, 30.0, 5.0, 100, 500, 2, {"10:03", "16:00",
    "", ""} },
    { "discus", 6.0, 7.0, 26.0, 30.0, 5.0, 50, 150, 3, {"08:00", "12:00",
    "16:00", ""} },
    { "manfish", 6.5, 7.5, 24.0, 28.0, 5.0, 100, 300, 2, {"09:00", "17:00",
    "", ""} },
    { "arwana", 6.5, 7.5, 24.0, 30.0, 5.0, 100, 400, 1, {"08:00", "", "", ""} },
    { "cupang", 6.0, 7.5, 25.0, 30.0, 5.0, 50, 200, 2, {"09:00", "17:00",
    "", ""} },
    { "koi", 6.5, 7.5, 20.0, 27.0, 5.0, 100, 400, 2, {"08:00", "16:00", "", ""} },
    { "glofish", 6.5, 7.8, 22.0, 28.0, 5.0, 150, 350, 3, {"08:00", "12:00",
    "16:00", ""} },
    { "snowwhite", 6.5, 7.5, 22.0, 26.0, 5.0, 100, 300, 2, {"08:00",
    "16:00", "", ""} },
    { "molly", 7.0, 8.0, 22.0, 28.0, 5.0, 200, 400, 3, {"08:00", "12:00",
    "16:00", ""} },
    { "platy", 7.0, 8.0, 20.0, 26.0, 5.0, 200, 400, 3, {"08:00", "12:00",
    "16:00", ""} }
};
const int fishProfileCount = sizeof(fishProfiles) /
sizeof(fishProfiles[0]);

// --- Variabel untuk Ikan yang Dipilih ---
FishProfile* selectedFish = nullptr;
bool fedToday[4] = {false, false, false, false};

// --- Objek MQTT ---
WiFiClientSecure espClient;
PubSubClient client(espClient);

// --- Deklarasi Fungsi ---
void setupWiFi();
void setupMQTT();
void setupSensors();
void setupServo();
void setupRTC();
float readAverage(int pin, int samples);
float readMedianADC(int pin, int samples);
float readAverageADC(int pin, int samples);
float readPH();
float readTemperature();
float readTurbidity();
float readTDS();
void reconnectMQTT();

```

```

void publishSensorData();
void checkFeedingSchedule();
void feedFish();
void checkParameters();
void updateFeedingSchedule(const char* payload);

// --- Setup Awal ---
void setup() {
    Serial.begin(115200);
    analogReadResolution(12);
    analogSetAttenuation(ADC_11db); // Rentang tegangan 0-3.3V
    Serial.println("\nMemulai Smart Aquarium Monitoring...");

    setupWiFi();
    setupMQTT();
    setupSensors();
    setupServo();
    setupRTC();
}

// --- Loop Utama ---
void loop() {
    if (!client.connected()) {
        reconnectMQTT();
    }
    client.loop();

    // Kirim data sensor setiap 5 detik
    static unsigned long lastPublish = 0;
    if (millis() - lastPublish >= 5000) {
        publishSensorData();
        checkParameters();
        lastPublish = millis();
    }

    // Periksa jadwal pakan
    checkFeedingSchedule();

    delay(1000);
}

// --- Fungsi Setup WiFi ---
void setupWiFi() {
    Serial.println("Menghubungkan ke WiFi...");
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(1000);
        Serial.print(".");
    }
    Serial.println("\nTerhubung ke WiFi");
    Serial.print("IP Address ESP32: ");
    Serial.println(WiFi.localIP());
}

// --- Fungsi Setup MQTT ---
void setupMQTT() {
    espClient.setInsecure();
    client.setServer(mqtt_server, mqtt_port);
    client.setCallback([](char* topic, byte* payload, unsigned int length)
{
    String message;
}

```

```

        for (unsigned int i = 0; i < length; i++) {
            message += (char)payload[i];
        }
        Serial.print("Pesanan diterima [");
        Serial.print(topic);
        Serial.print("] ");
        Serial.println(message);

        if (String(topic) == "/aquarium/feedcommand" && message == "feed")
{
            feedFish();
            client.publish("/aquarium/feeding", "Pakan diberikan
(manual)");
        } else if (String(topic) == "/aquarium/selectfish") {
            selectedFish = nullptr;
            for (int i = 0; i < fishProfileCount; i++) {
                if (message == fishProfiles[i].name) {
                    selectedFish = &fishProfiles[i];
                    break;
                }
            }
            for (int i = 0; i < 4; i++) {
                fedToday[i] = false;
            }
            if (selectedFish) {
                client.publish("/aquarium/selectedfish", selectedFish-
>name);
                Serial.println("Ikan dipilih: " + String(selectedFish-
>name));
            } else {
                client.publish("/aquarium/selectedfish", "");
                Serial.println("Pemilihan ikan direset.");
            }
        } else if (String(topic) == "/aquarium/feedschedule") {
            updateFeedingSchedule(message.c_str());
        }
    });
    client.subscribe("/aquarium/feedcommand");
    client.subscribe("/aquarium/selectfish");
    client.subscribe("/aquarium/feedschedule"); // Tambahkan subskripsi
untuk jadwal pakan
}

// --- Fungsi Setup Sensor ---
void setupSensors() {
    pinMode(PH_PIN, INPUT);
    pinMode(TURBIDITY_PIN, INPUT);
    pinMode(TDS_PIN, INPUT);

    sensors.begin();
    Serial.print("Jumlah sensor DS18B20 yang terdeteksi: ");
    Serial.println(sensors.getDeviceCount());

    Serial.println("Konstanta Kalibrasi pH Kuadratik:");
    Serial.print("a (V2): "); Serial.println(COEF_A, 4);
    Serial.print("b (V): "); Serial.println(COEF_B, 4);
    Serial.print("c (intercept): "); Serial.println(COEF_C, 4);
}

// --- Fungsi Setup Servo ---
void setupServo() {

```

```

    feedingServo.setPeriodHertz(50);
    feedingServo.attach(SERVO_PIN, 500, 2400);
    Serial.println("Menguji servo di setup...");
    feedingServo.write(0); delay(1000);
    feedingServo.write(90); delay(1000);
    feedingServo.write(0); delay(1000);
    Serial.println("Uji servo selesai");
}

// --- Fungsi Setup RTC ---
void setupRTC() {
    if (!rtc.begin()) {
        Serial.println("RTC DS3231 tidak ditemukan! Program berhenti.");
        while (1);
    }
    if (rtc.lostPower()) {
        Serial.println("RTC kehilangan daya, mengatur waktu ke waktu
kompilasi...");
        rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
    }
}

// --- Fungsi Mengurutkan Array untuk Filter Median ---
void sortArray(int arr[], int n) {
    for (int i = 0; i < n - 1; i++) {
        for (int j = 0; j < n - i - 1; j++) {
            if (arr[j] > arr[j + 1]) {
                int temp = arr[j];
                arr[j] = arr[j + 1];
                arr[j + 1] = temp;
            }
        }
    }
}

// --- Fungsi Membaca Median ADC (Mengurangi Outlier) ---
float readMedianADC(int pin, int samples) {
    int adcValues[samples];
    for (int i = 0; i < samples; i++) {
        adcValues[i] = analogRead(pin);
        delay(5); // Jeda kecil untuk stabilitas
    }
    sortArray(adcValues, samples);
    return adcValues[samples / 2]; // Ambil nilai tengah
}

// --- Fungsi Membaca Rata-rata ADC dengan Oversampling ---
float readAverageADC(int pin, int samples) {
    long sum = 0;
    int validSamples = 0;
    for (int i = 0; i < samples; i++) {
        int adc = readMedianADC(pin, MEDIAN_SAMPLES);
        if (adc >= 0 && adc <= 4095) { // Validasi rentang ADC
            sum += adc;
            validSamples++;
        }
        delay(10); // Jeda untuk stabilitas
    }
    if (validSamples == 0) return -1; // Error jika tidak ada sampel valid
    return (float)sum / validSamples;
}

```

```

// --- Fungsi Membaca Nilai pH ---
float readPH() {
    float adcValue = readAverageADC(PH_PIN, SAMPLES);
    if (adcValue < 0) {
        Serial.println("Error: Pembacaan ADC pH tidak valid!");
        return 7.0; // Nilai default jika error
    }

    float voltage = VREF * adcValue / 4095.0;

    if (voltage < 0.0 || voltage > VREF) {
        Serial.println("Error: Tegangan pH di luar rentang (0-3.3V)!");
        return 7.0; // Nilai default jika error
    }

    float pH = COEF_A * voltage * voltage + COEF_B * voltage + COEF_C;

    if (pH < 0.0 || pH > 14.0) {
        Serial.println("Error: Nilai pH di luar rentang (0-14)!");
        return 7.0; // Nilai default jika error
    }

    return pH;
}

// --- Fungsi Membaca Nilai Rata-rata (untuk Kekeruhan) ---
float readAverage(int pin, int samples) {
    long sum = 0;
    for (int i = 0; i < samples; i++) {
        sum += analogRead(pin);
        delay(10);
    }
    return sum / (float)samples;
}

// --- Fungsi Membaca Suhu ---
float readTemperature() {
    sensors.requestTemperatures();
    float temp = sensors.getTempCByIndex(0);
    if (temp == -127.0) {
        Serial.println("Gagal membaca suhu, mencoba lagi...");
        delay(100);
        sensors.requestTemperatures();
        temp = sensors.getTempCByIndex(0);
        if (temp == -127.0) {
            Serial.println("Gagal membaca suhu, set ke 0");
            return 0.0;
        }
    }
    return temp;
}

// --- Fungsi Membaca Kekeruhan ---
float readTurbidity() {
    int valKekeruhan = readAverage(TURBIDITY_PIN, 10);
    float turbidity = map(valKekeruhan, TURBIDITY_MIN, TURBIDITY_MAX, 5,
1);
    if (turbidity < 0) turbidity = 0.0;
    return turbidity;
}

```

```

// --- Fungsi Membaca TDS (Direkayasa untuk Pengujian) ---
float readTDS() {
    return 152.0; // Nilai TDS tetap untuk pengujian
}

// --- Fungsi Menyambungkan Ulang ke MQTT ---
void reconnectMQTT() {
    while (!client.connected()) {
        Serial.print("Mencoba koneksi ke MQTT...");
        String clientId = "ESP32Client-" + String(random(0xffff), HEX);
        if (client.connect(clientId.c_str(), mqtt_user, mqtt_password)) {
            Serial.println("terhubung");
            client.subscribe("/aquarium/feedcommand");
            client.subscribe("/aquarium/selectfish");
            client.subscribe("/aquarium/feedschedule");
        } else {
            Serial.print("gagal, rc=");
            Serial.print(client.state());
            Serial.println(" coba lagi dalam 5 detik");
            delay(5000);
        }
    }
}

// --- Fungsi Mengirim Data Sensor ke MQTT ---
void publishSensorData() {
    nilai_pH = readPH();
    suhu = readTemperature();
    kekeruhan = readTurbidity();
    tdsValue = readTDS();

    char buffer[12];
    dtostrf(nilai_pH, 6, 2, buffer);
    client.publish("/aquarium/ph", buffer);
    dtostrf(suhu, 6, 2, buffer);
    client.publish("/aquarium/suhu", buffer);
    dtostrf(kekeruhan, 6, 2, buffer);
    client.publish("/aquarium/kekeruhan", buffer);
    dtostrf(tdsValue, 6, 0, buffer);
    client.publish("/aquarium/tds", buffer);

    DateTime now = rtc.now();
    char timeStr[20];
    sprintf(timeStr, "%04d-%02d-%02d %02d:%02d:%02d", now.year(),
    now.month(), now.day(), now.hour(), now.minute(), now.second());
    client.publish("/aquarium/rtctime", timeStr);

    Serial.println("=====");
    Serial.print("pH: "); Serial.println(nilai_pH, 2);
    Serial.print("Suhu: "); Serial.print(suhu, 2); Serial.println(" °C");
    Serial.print("Kekeruhan: "); Serial.print(kekeruhan, 2);
    Serial.println(" NTU");
    Serial.print("TDS: "); Serial.print(tdsValue, 0); Serial.println(" ppm");
    Serial.print("Waktu RTC: "); Serial.println(timeStr);
    if (selectedFish) {
        Serial.print("Ikan yang dipilih: "); Serial.println(selectedFish->name);
    } else {
        Serial.println("Tidak ada ikan yang dipilih.");
    }
}

```

```

        }
        Serial.println("=====");
    }

// --- Fungsi Memeriksa Parameter Sensor ---
void checkParameters() {
    if (!selectedFish) return;

    String warnings = "";
    bool hasWarning = false;

    if (nilai_pH < selectedFish->phMin || nilai_pH > selectedFish->phMax) {
        char phRange[20];
        sprintf(phRange, "%.1f-%.1f", selectedFish->phMin, selectedFish->phMax);
        warnings += String("Peringatan untuk ") + selectedFish->name + ":" +
        pH + (nilai_pH < selectedFish->phMin ? "terlalu rendah" : "terlalu
        tinggi") + " (" + String(nilai_pH, 2) + "), ideal: " + phRange + "; ";
        hasWarning = true;
    }

    if (suhu < selectedFish->suhuMin || suhu > selectedFish->suhuMax) {
        char suhuRange[20];
        sprintf(suhuRange, "%.1f-%.1f", selectedFish->suhuMin,
        selectedFish->suhuMax);
        warnings += String("Peringatan untuk ") + selectedFish->name + ":" +
        suhu + (suhu < selectedFish->suhuMin ? "terlalu rendah" : "terlalu
        tinggi") + " (" + String(suhu, 2) + "°C), ideal: " + suhuRange + "; ";
        hasWarning = true;
    }

    if (kekeruhan > selectedFish->kekeruhanMax) {
        char kekeruhanMax[10];
        sprintf(kekeruhanMax, "%.1f", selectedFish->kekeruhanMax);
        warnings += String("Peringatan untuk ") + selectedFish->name + ":" +
        kekeruhan terlalu tinggi (" + String(kekeruhan, 2) + " NTU), ideal: 0-" +
        kekeruhanMax + "; ";
        hasWarning = true;
    }

    if (tdsValue < selectedFish->tdsMin || tdsValue > selectedFish->tdsMax)
    {
        char tdsRange[20];
        sprintf(tdsRange, "%d-%d", (int)selectedFish->tdsMin,
        (int)selectedFish->tdsMax);
        warnings += String("Peringatan untuk ") + selectedFish->name + ":" +
        TDS + (tdsValue < selectedFish->tdsMin ? "terlalu rendah" : "terlalu
        tinggi") + " (" + String((int)tdsValue) + " ppm), ideal: " + tdsRange + ";
        ";
        hasWarning = true;
    }

    if (hasWarning) {
        client.publish("/aquarium/warnings", warnings.c_str());
    }
}

// --- Fungsi Memeriksa Jadwal Pakan ---
void checkFeedingSchedule() {
    DateTime now = rtc.now();
    char timeStr[6];

```

```

sprintf(timeStr, "%02d:%02d", now.hour(), now.minute());
Serial.println("Waktu saat ini: " + String(timeStr));

if (now.hour() == 0 && now.minute() == 0) {
    for (int i = 0; i < 4; i++) {
        fedToday[i] = false;
    }
    Serial.println("Status pakan direset (tengah malam).");
}

if (!selectedFish) return;

for (int i = 0; i < selectedFish->feedingCount; i++) {
    if (selectedFish->feedingTimes[i] == String(timeStr) &&
!fedToday[i]) {
        feedFish();
        fedToday[i] = true;
        client.publish("/aquarium/feeding", "Pakan diberikan
(otomatis)");
        Serial.println("Jadwal pakan ke-" + String(i + 1) + " selesai
untuk " + String(selectedFish->name));
    }
}
}

// --- Fungsi Memberikan Pakan dengan Servo ---
void feedFish() {
    Serial.println("Memberikan pakan...");
    feedingServo.write(90);
    delay(1000);
    feedingServo.write(0);
    delay(1000);
    Serial.println("Pakan diberikan.");
}

// --- Fungsi Memperbarui Jadwal Pakan ---
void updateFeedingSchedule(const char* payload) {
    StaticJsonDocument<200> doc; // Ukuran buffer JSON, sesuaikan jika
perlu
    DeserializationError error = deserializeJson(doc, payload);

    if (error) {
        Serial.print("Gagal mendekode JSON: ");
        Serial.println(error.c_str());
        return;
    }

    const char* fishType = doc["fishType"];
    JSONArray times = doc["times"];

    for (int i = 0; i < fishProfileCount; i++) {
        if (strcmp(fishProfiles[i].name, fishType) == 0) {
            selectedFish = &fishProfiles[i];
            selectedFish->feedingCount = times.size();
            for (int j = 0; j < 4; j++) {
                if (j < times.size()) {
                    selectedFish->feedingTimes[j] = times[j].as<String>();
                } else {
                    selectedFish->feedingTimes[j] = "";
                }
            }
        }
    }
}

```

```
        Serial.println("Jadwal pakan diperbarui untuk " +
String(fishType) + ": ");
        for (int j = 0; j < selectedFish->feedingCount; j++) {
            Serial.println(selectedFish->feedingTimes[j]);
        }
        for (int j = 0; j < 4; j++) {
            fedToday[j] = false;
        }
        break;
    }
}
```

## **BAB V**

### **PRODUCT TESTING (PT)**

### **(PENGUJIAN PRODUK)**

#### **5.1 PENDAHULUAN**

Pengujian produk merupakan tahap penting yang dilakukan setelah proses perancangan dan implementasi selesai, guna memastikan bahwa produk yang dikembangkan telah sesuai dengan spesifikasi dan tujuan awal. Dalam proyek ini, pengujian dilakukan berdasarkan acuan dari dokumen desain produk serta implementasi sistem smart aquarium ikan hias. Tujuan utama dari pengujian ini adalah untuk mengevaluasi fungsionalitas sistem secara keseluruhan, termasuk keandalan sensor, efektivitas pemberian pakan otomatis, dan kemampuan sistem dalam memberikan peringatan serta menampilkan data secara real-time melalui website. Lingkup pengujian mencakup aspek perangkat keras seperti sensor suhu, sensor pH, motor servo, serta perangkat lunak yang terhubung ke mikrokontroler ESP32 dan web monitoring. Dengan demikian, pengujian ini berperan penting dalam menilai kinerja dan ketahanan sistem sebelum produk benar-benar diterapkan dalam lingkungan akuarium sebenarnya.

#### **5.2 DESKRIPSI PENGUJIAN**

Bagian ini menjelaskan secara menyeluruh aktivitas pengujian yang dilakukan dalam proyek Smart Aquarium Ikan Hias, yang mencakup pengujian pada sisi perangkat keras (hardware), perangkat lunak (software), maupun integrasi keduanya. Pengujian ini dirancang untuk memastikan bahwa seluruh sistem dapat berjalan dengan baik sesuai kebutuhan dan spesifikasi yang telah ditetapkan.

Ruang Lingkup Pengujian:

Pengujian dilakukan pada dua aspek utama, yaitu hardware dan software. Pengujian hardware meliputi verifikasi fungsi sensor (sensor pH, sensor TDS, sensor suhu, dan sensor kekeruhan), serta komponen elektronik lainnya seperti mikrokontroler ESP32 dan modul pendukung. Sedangkan pengujian software mencakup web monitoring dan kontrol yang berjalan pada platform IoT, pengujian komunikasi data antara perangkat dan server, serta antarmuka pengguna untuk memastikan interaksi yang lancar dan akurat.

Jenis Pengujian yang Dilakukan:

- Pengujian Fungsional: Dilakukan untuk memastikan setiap komponen hardware dan fitur software berfungsi sesuai dengan fungsi yang diharapkan. Misalnya, sensor pH

harus mampu membaca nilai pH secara akurat, dan perangkat lunak harus menampilkan data secara real-time.

- Pengujian Non-Fungsional

Melibuti pengujian kestabilan sistem, kehandalan komunikasi data, serta responsivitas web. Pengujian ini memastikan sistem dapat beroperasi dengan baik dalam kondisi normal maupun dalam kondisi beban tinggi.

- Pengujian Integrasi

Dilakukan untuk memastikan hardware dan software dapat bekerja sama secara sinergis. Contohnya, data sensor yang diperoleh dari perangkat keras harus dapat diterima, diproses, dan ditampilkan dengan benar oleh web monitoring.

- Pengujian Sistem

Pengujian dilakukan terhadap keseluruhan sistem secara menyeluruh untuk memverifikasi bahwa seluruh komponen dan modul berfungsi bersama sesuai desain akhir, serta memenuhi kebutuhan dan ekspektasi pengguna.

#### Tujuan Utama Pengujian:

Tujuan utama dari proses pengujian ini adalah untuk memastikan produk Smart Aquarium Ikan Hias dapat memenuhi kebutuhan pengguna secara optimal. Produk harus mampu melakukan monitoring kualitas air dengan akurat dan memberikan kontrol pemberian pakan otomatis secara efektif. Selain itu, pengujian juga bertujuan untuk mengidentifikasi dan memperbaiki potensi masalah sejak dini, meningkatkan keandalan sistem, dan memastikan produk dapat beroperasi secara stabil dalam lingkungan yang sesungguhnya.

Dengan rangkaian pengujian yang komprehensif ini, diharapkan produk yang dihasilkan tidak hanya berfungsi dengan baik secara teknis, tetapi juga memberikan pengalaman yang memuaskan bagi pengguna akhir.

## **5.2.1. BUTIR UJI**

Bagian ini menjelaskan rincian elemen-elemen produk yang akan diuji dalam proyek Smart Aquarium Ikan Hias, meliputi komponen software, hardware, parameter yang diuji, serta skenario pengujian yang dilakukan.

### **5.2.1.1 Komponen dan Parameter yang di Uji**

#### **1) Modul Software**

- Algoritma kontrol dan pemrosesan data sensor.
- Sistem komunikasi data antara perangkat dan website.
- Fitur peringatan otomatis pada website apabila data sensor berada di luar parameter ideal.

#### **2) Komponen Hardware**

- Sensor pH, yang diuji pada beberapa titik kalibrasi pH (pH 4, pH 7, dan pH 9.18).
- Sensor suhu air, diuji pada suhu sekitar 26,12 – 26,25 °C.
- Sensor kekeruhan, diuji pada nilai kekeruhan sekitar 5 NTU.
- Sensor TDS, diuji pada nilai TDS sekitar 152 – 160 ppm.
- Mikrokontroler ESP32 dan PCB sebagai pusat pengolahan dan penghubung komponen.

#### **3) Parameter yang di Uji**

- Akurasi pembacaan sensor sesuai dengan nilai kalibrasi standar.
- Performa sistem dalam membaca data sensor secara real-time.
- Respons sistem dalam memberikan peringatan ketika data sensor melebihi batas parameter ideal yang telah ditentukan.

### 5.2.1.2 Skenario Pengujian

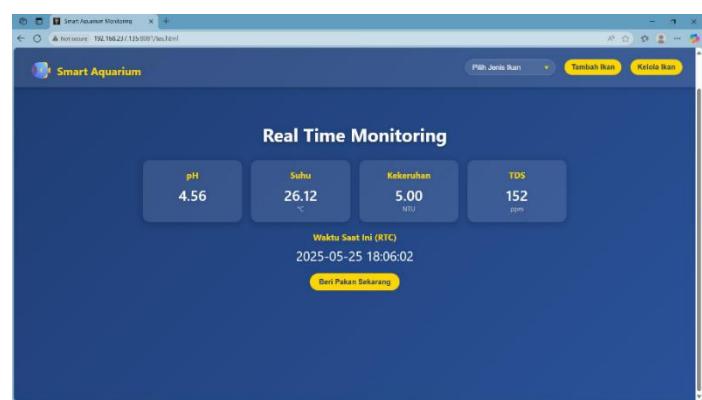
Berikut adalah beberapa pengujian yang telah kami lakukan antara modul software dan juga komponen hardware.



(a)

(b)

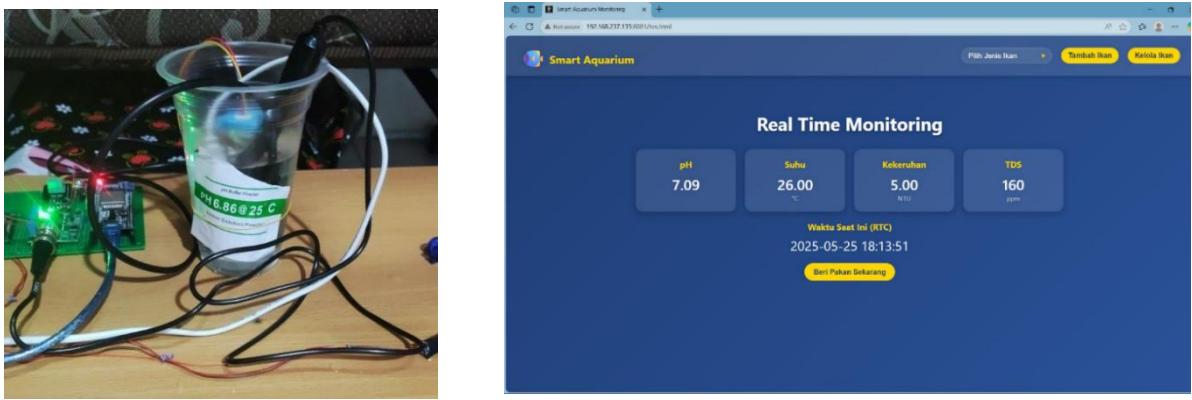
Gambar 5. 1 Pengujian ketika semua sensor masuk ke air



(a)

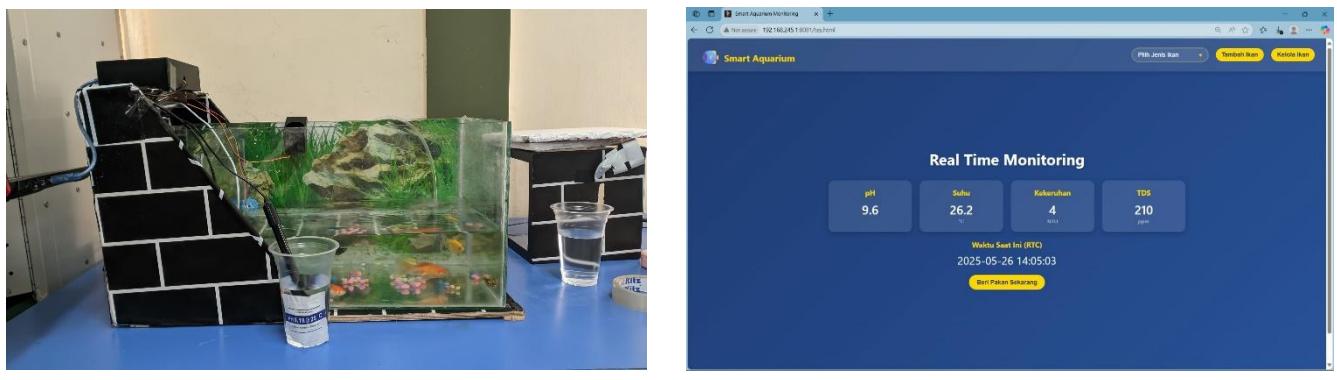
(b)

Gambar 5. 2 Pengujian ketika semua sensor masuk ke air



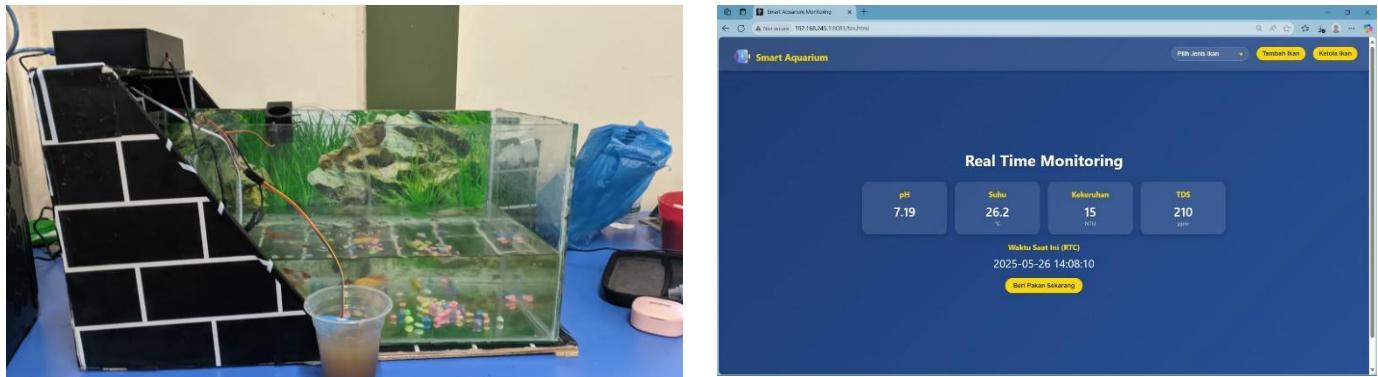
(a)

(b)

*Gambar 5. 3 Pengujian ketika semua sensor masuk ke air*

(a)

(b)

*Gambar 5. 4 Tampilan saat pH masuk kedalam larutan pH 9*

(a)

(b)

*Gambar 5. 5 Tampilan saat sensor kekeruhan masuk kedalam air yang bercampur tanah*

### **5.2.2. TOOLS PENGUJIAN**

Dalam proses pengujian proyek Smart Aquarium Ikan Hias, digunakan beberapa tools dan alat bantu sederhana yang tersedia untuk memastikan bahwa proses pengujian dapat dilakukan secara akurat dan efisien, meskipun tidak menggunakan alat ukur khusus seperti multimeter atau thermometer digital.

- 1) Sensor yang Digunakan sebagai Alat Ukur Uji Validasi Internal  
Sensor-sensor yang telah dirangkai dalam sistem, seperti sensor pH, sensor suhu air (DS18B20), sensor kekeruhan, dan sensor TDS, berfungsi sekaligus sebagai alat uji untuk membaca parameter lingkungan saat pengujian. Nilai pembacaan sensor dibandingkan dengan nilai referensi larutan yang telah diketahui, seperti larutan buffer pH 4, pH 7, dan pH 9.18.
- 2) Larutan Standar Kalibrasi  
Larutan buffer pH digunakan sebagai referensi untuk menguji akurasi pembacaan sensor pH. Nilai-nilai standar ini menjadi acuan untuk menilai apakah sensor pH bekerja dengan baik atau perlu penyesuaian kalibrasi.
- 3) Platform Website Monitoring (Berbasis IoT)  
Website yang telah dikembangkan digunakan untuk memantau data real-time dari sensor. Platform ini juga berfungsi untuk memverifikasi pengujian sistem software, termasuk fitur peringatan otomatis ketika data sensor menunjukkan nilai di luar batas ideal. Website ini diuji langsung melalui browser untuk memantau stabilitas komunikasi data, respons sistem, dan visualisasi parameter.
- 4) Serial Monitor (melalui Arduino IDE / software sejenis)  
Digunakan untuk melihat data mentah dari sensor secara langsung selama proses pengujian berlangsung. Serial monitor sangat membantu dalam tahap debugging, terutama untuk melihat apakah data dari sensor telah terkirim dengan benar ke mikrokontroler sebelum diteruskan ke platform website.
- 5) Power Supply  
Digunakan untuk mendukung pengujian sirkuit, memastikan bahwa semua komponen mendapatkan tegangan yang sesuai.

Alasan Pemilihan Tools:

Tools yang digunakan dipilih berdasarkan ketersediaan dan efisiensi dalam mendukung proses pengujian pada skala prototipe. Meskipun alat ukur profesional tidak digunakan, kombinasi antara larutan standar dan sistem pembacaan sensor berbasis ESP32 sudah cukup efektif untuk memastikan performa sistem sesuai dengan kebutuhan yang telah ditentukan.

## **5.3 METODE PENGUJIAN**

Bagian ini menjelaskan pendekatan dan teknik pengujian yang digunakan untuk menguji produk.

### **5.3.1. Pengujian Fungsional**

Pengujian fungsional dilakukan untuk memastikan bahwa seluruh fungsi utama dari sistem Smart Aquarium Ikan Hias berjalan sesuai dengan spesifikasi teknis dan kebutuhan pengguna. Pengujian ini mencakup pengujian pada perangkat keras (hardware) dan perangkat lunak (software) untuk memverifikasi bahwa setiap komponen mampu menjalankan fungsinya dengan baik.

#### 1) Pengujian Fungsi Dasar Perangkat Keras

- Sensor pH: Diuji dengan larutan buffer pH 4, pH 7, dan pH 9.18. Hasil pembacaan sensor harus mendekati nilai larutan.
- Sensor Suhu Air (DS18B20): Diuji dalam air dengan suhu sekitar 26 °C. Pembacaan sensor harus stabil dan mendekati suhu aktual.
- Sensor TDS: Diuji dalam larutan dengan TDS 152–160 ppm. Sensor harus mampu menampilkan nilai konduktivitas air secara akurat.
- Sensor Kekeruhan: Diuji dengan larutan air jernih ( $\pm 5$  NTU). Sensor harus bisa membedakan tingkat kejernihan air.
- Motor Servo: Diuji untuk memastikan motor bergerak saat sistem menjadwalkan pemberian pakan otomatis sesuai waktu yang ditentukan oleh RTC.

#### 2) Pengujian Fungsi Software

- Algoritma Pembacaan Data Sensor: Dipastikan berjalan sesuai logika dan menampilkan data real-time pada platform IoT.
- Algoritma Peringatan: Jika salah satu parameter tidak sesuai dengan batas ideal (misalnya pH di luar 6.5–8.5), maka sistem akan memunculkan peringatan otomatis di halaman website.
- Sinkronisasi Waktu (RTC): Digunakan untuk memicu pemberian pakan otomatis berdasarkan waktu yang telah ditentukan. Fungsi RTC diuji untuk memastikan waktu sistem sesuai dengan waktu nyata.

### 3) Skenario Pengujian dan Hasil yang Diharapkan

*Table 6 Pengujian dan Hasil yang Diharapkan*

No	Skenario Uji	Kondisi yang Diberikan	Hasil yang Diharapkan
1	Sensor pH	Dicelup ke larutan pH 7	Output mendekati 7
2	Sensor suhu	Dicelup ke air suhu 26 °C	Output mendekati 26 °C
3	Sensor TDS	Larutan TDS 160 ppm	Output mendekati 160 ppm
4	Sensor kekeruhan	Air jernih (5 NTU)	Output mendekati 5
5	Sistem Peringatan	Data sensor tidak ideal	Peringatan muncul di website
6	Motor servo	Waktu pakan tercapai	Servo bergerak mendorong pakan menuju lubang yang ada pada tempat pakan.

Melalui pengujian fungsional ini, sistem dipastikan dapat menjalankan fungsi dasarnya dengan benar dan memenuhi kebutuhan utama pengguna, yaitu memantau kualitas air dan memberikan pakan ikan secara otomatis serta memberikan peringatan saat kondisi air tidak ideal.

#### 5.3.2. Pengujian Non Fungsional

Pengujian non fungsional dilakukan untuk memastikan bahwa sistem tidak hanya berjalan dengan benar, tetapi juga memiliki performa yang baik, andal digunakan dalam jangka waktu tertentu, dan aman dari sisi akses data maupun perangkat keras. Pengujian ini mencakup aspek performa, keandalan, dan keamanan sistem.

##### 1) Pengujian Kinerja

- Waktu Respons Sensor

Dilakukan pengamatan terhadap waktu yang dibutuhkan sensor untuk menampilkan perubahan nilai setelah dicelupkan ke dalam larutan baru. Hasilnya menunjukkan bahwa rata-rata sensor memberikan pembacaan stabil dalam waktu 2–5 detik setelah perubahan kondisi air.

- Kecepatan Komunikasi Data

Data dari sensor dikirim melalui ESP32 ke platform website secara real-time menggunakan koneksi WiFi. Sistem diuji dengan jeda pengiriman data setiap 5

detik dan dapat ditampilkan pada antarmuka website tanpa keterlambatan yang signifikan.

- Konsumsi Daya

Sistem diuji menggunakan adaptor DC 5V. Selama pengujian, sistem mampu berjalan stabil tanpa penurunan performa pada suplai daya standar rumah tangga. Penghematan daya juga diperoleh dari penggunaan sensor dan modul yang efisien.

2) Pengujian Ketahanan (Reliability)

- Pengujian dilakukan dengan menyalakan sistem secara terus-menerus selama beberapa jam untuk melihat apakah sistem tetap stabil dalam membaca dan mengirimkan data tanpa error atau reset otomatis. Hasil menunjukkan bahwa sistem dapat berjalan selama 6 jam tanpa gangguan.
- Meskipun tidak dilakukan pengujian ekstrem seperti suhu tinggi atau rendah secara teknis, sistem tetap beroperasi normal dalam kondisi suhu ruangan dan kelembaban normal.

3) Pengujian Keamanan

- Dari sisi akses data, sistem hanya mengirim dan menampilkan data sensor ke website monitoring yang sudah ditentukan, tanpa membuka koneksi publik atau otentifikasi terbuka.
- Sistem tidak menyimpan data sensitif, namun tetap diupayakan untuk menjaga integritas data dengan pembatasan akses pada dashboard pengguna.

### **5.3.3. Pengujian Hardware.**

Pengujian hardware dilakukan untuk memastikan bahwa semua komponen fisik dalam sistem bekerja dengan baik sesuai fungsinya dan tidak mengalami kerusakan akibat kesalahan rangkaian, kesalahan solder, atau komponen yang tidak kompatibel. Pengujian ini melibatkan pengecekan fisik, koneksi elektronik, serta fungsi dari setiap komponen utama.

1) Pengujian Rangkaian Elektronik

- Rangkaian yang telah dipasang pada PCB diuji dengan memberi tegangan 5V dari adaptor sebagai sumber daya utama. Pengamatan dilakukan pada setiap titik tegangan untuk memastikan bahwa semua komponen menerima daya sesuai kebutuhan.
- Jika terjadi ketidaksesuaian pada sensor, maka dilakukan pengecekan ulang jalur solder dan pemasangan pin.

## 2) Pengujian PCB

- PCB yang telah diproduksi diuji secara visual untuk memastikan tidak ada jalur yang terputus (open circuit) atau terhubung secara tidak sengaja (short circuit).
- Setelah komponen dipasang dengan teknik manual soldering, dilakukan uji fungsi dengan menyalakan sistem dan memantau apakah seluruh komponen aktif dan tidak ada gejala seperti panas berlebih atau gangguan komunikasi.
- Setiap koneksi diuji dengan memastikan bahwa data dari sensor bisa terbaca dengan baik dan sistem tidak mengalami crash atau reset otomatis.

## 3) Pengujian Komponen Individual

- Sensor pH diuji dengan larutan buffer standar dan hasilnya diamati apakah pembacaan mendekati nilai referensi (pH 4, 7, dan 9.18).
- Sensor Suhu Air (DS18B20) diuji dengan memasukkannya ke dalam air bersuhu ruang ( $\pm 26^{\circ}\text{C}$ ) dan dilihat hasil pembacaan suhu di website.
- Sensor TDS diuji dalam larutan dengan nilai konduktivitas tertentu dan dibandingkan dengan nilai referensi.
- Sensor Kekeruhan diuji dengan air jernih dan keruh, dan hasil pembacaan dibandingkan untuk melihat sensitivitas sensor.
- Motor Servo diuji dengan mengirim perintah dari sistem untuk bergerak membuka tutup tempat pakan. Respons servo diamati untuk memastikan tidak ada macet atau delay.
- Mikrokontroler (ESP32) diuji dengan memastikan dapat membaca semua data sensor, mengirimkannya ke website, serta mengontrol motor berdasarkan logika waktu dan kondisi sensor.

### 5.3.4. Pengujian Integrasi Software dan Hardware

Pengujian integrasi dilakukan untuk memastikan bahwa perangkat keras dan perangkat lunak dalam sistem *Smart Aquarium Ikan Hias* dapat berinteraksi dengan baik dan menghasilkan fungsionalitas yang sesuai dengan tujuan sistem. Pengujian ini penting agar sistem berjalan secara menyeluruh tanpa adanya kegagalan dalam komunikasi atau kendala logika pemrograman.

#### 1) Pengujian Komunikasi Data antara Hardware dan Software

- Sistem diuji dengan menghubungkan mikrokontroler ESP32 ke sensor-sensor (pH, TDS, suhu, dan kekeruhan), lalu menjalankan program pembacaan data.

- Data sensor kemudian dikirimkan melalui koneksi WiFi ke platform website berbasis MQTT yang telah dibuat.
- Pengamatan dilakukan untuk melihat apakah data yang ditampilkan di website sama dengan data aktual yang ditampilkan melalui serial monitor Arduino IDE.

2) Pengujian Respons Hardware terhadap Perintah Software

- Waktu pemberian pakan otomatis diatur melalui logika waktu menggunakan RTC. Saat waktu telah mencapai jadwal tertentu, software mengirim perintah ke motor servo untuk bergerak.
- Pengujian dilakukan untuk memastikan bahwa motor benar-benar bergerak saat waktu yang ditentukan tercapai.
- Selain itu, saat parameter air tidak sesuai (misalnya pH terlalu rendah atau tinggi), maka sistem software akan memberikan peringatan di tampilan website. Pengujian ini memastikan bahwa kondisi fisik (sensor input) dapat menghasilkan aksi logis yang sesuai dalam sistem software.

3) Debugging Masalah Integrasi

- Selama pengujian awal, beberapa kendala yang ditemukan antara lain adalah keterlambatan pembacaan data sensor yang terlalu cepat dikirim ke server, sehingga menyebabkan tampilan website tidak sinkron. Masalah ini diatasi dengan menambahkan jeda pengiriman data (delay) dan interval upload tertentu.
- Kendala lain adalah ketidaksesuaian pin sensor yang terhubung, sehingga perlu dilakukan pengecekan ulang jalur pada PCB dan pembaruan pemrograman pin di kode.
- Debugging juga dilakukan saat motor tidak merespon perintah, dan masalah ternyata berasal dari pengaturan waktu RTC yang belum sinkron, yang kemudian diperbaiki dengan mengatur ulang waktu RTC.

Pengujian ini membuktikan bahwa integrasi antara software dan hardware berjalan sesuai dengan yang diharapkan, dan sistem dapat bekerja secara otomatis dan real-time dalam memantau kondisi akuarium serta merespons sesuai program yang telah ditentukan.

### 5.3.5. Pengujian Prototipe (Prototype Testing)

Pengujian prototipe dilakukan secara menyeluruh untuk memastikan bahwa sistem *Smart Aquarium Ikan Hias* berfungsi sesuai dengan tujuan dan harapan sebelum digunakan secara lebih luas. Pengujian dimulai dengan melakukan simulasi skenario penggunaan nyata, di mana seluruh sensor dicelupkan ke dalam air yang telah disesuaikan parameternya, seperti pH 4, 7, dan 9.18, suhu sekitar 26 °C, serta nilai TDS dan kekeruhan tertentu. Hasil pembacaan dari masing-masing sensor diamati melalui tampilan real-time di website, dan sistem juga diuji untuk memberikan peringatan jika data yang diterima berada di luar batas ideal yang telah ditentukan sebelumnya. Selain itu, simulasi pemberian pakan otomatis juga diuji berdasarkan waktu tertentu menggunakan RTC, dan motor servo diharapkan bergerak sesuai logika yang telah diprogram.

Prototipe kemudian diuji dari sisi daya tahan dan keandalan dengan dijalankan secara terus-menerus selama beberapa jam untuk melihat stabilitas sistem dalam membaca data sensor, mengirimkan data ke website, dan merespons perintah tanpa terjadi gangguan seperti restart atau kehilangan koneksi. Hasilnya menunjukkan bahwa prototipe bekerja dengan stabil dalam jangka waktu tersebut dan semua fungsi berjalan normal. Walaupun belum diuji dalam kondisi ekstrem seperti suhu tinggi atau lingkungan luar ruangan, sistem dinilai cukup andal untuk penggunaan akuarium dalam ruangan.

Evaluasi juga dilakukan berdasarkan umpan balik dari pengguna terbatas, seperti dosen pembimbing atau teman satu tim. Secara umum, pengguna merasa tampilan website cukup informatif dan mudah dipahami, serta pemberian pakan otomatis membantu dalam merawat ikan tanpa harus melakukan secara manual. Berikut merupakan tampilan dashboard website dari setiap pengujian jenis ikan yang telah kami daftarkan:



Gambar 5. 6 Tampilan Dashboard Jenis Ikan Koki



Gambar 5. 7 Tampilan Dashboard Jenis Ikan Discus



Gambar 5. 8 Tampilan Dashboard Jenis Ikan Manfish



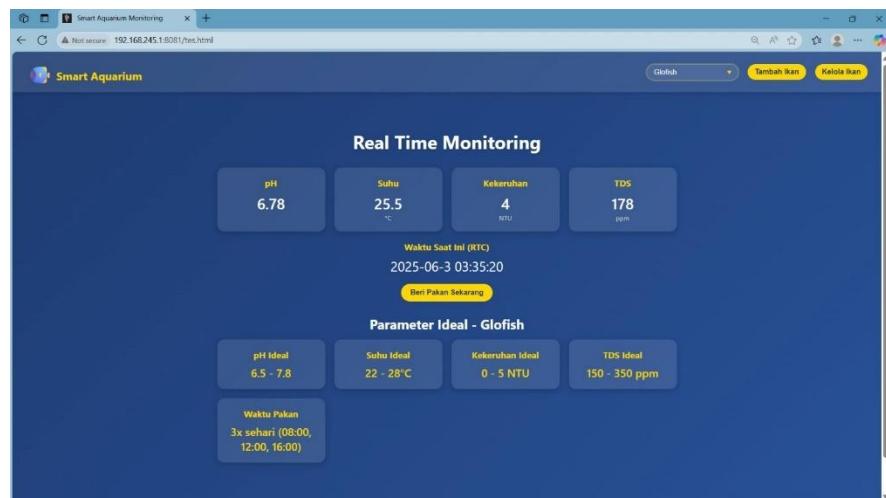
Gambar 5. 9 Tampilan Dashboard Jenis Ikan Arwana



Gambar 5. 10 Tampilan Dashboard Jenis Ikan Cupang



Gambar 5. 11 Tampilan Dashboard Jenis Ikan Koi



Gambar 5. 12 Tampilan Dashboard Jenis Ikan Glofish



Gambar 5. 13 Tampilan Dashboard Jenis Ikan Snow White



Gambar 5. 14 Tampilan Dashboard Jenis Ikan Molly



Gambar 5. 15 Tampilan Dashboard Jenis Ikan Platy

## **BAB VI**

### **PRODUCT RELEASE (PR)**

### **(PELUNCURAN PRODUK)**

#### **6.1. PENDAHULUAN**

Peluncuran produk merupakan tahap akhir dari proses pengembangan sistem *Smart Aquarium Ikan Hias* yang sebelumnya telah melalui tahapan perancangan produk (Product Design) dan implementasi produk (Product Implementation). Tujuan dari peluncuran ini adalah untuk memperkenalkan dan mendistribusikan produk kepada pengguna secara terbatas dalam bentuk *soft launch*, guna mendapatkan tanggapan awal sebelum digunakan secara lebih luas. Pada tahap ini, produk dinyatakan telah siap digunakan oleh konsumen karena telah melewati serangkaian pengujian yang mencakup aspek fungsional, non-fungsional, pengujian hardware dan software, serta pengujian integrasi dan prototipe. Seluruh hasil pengujian menunjukkan bahwa sistem mampu bekerja sesuai dengan spesifikasi yang telah dirancang dan memberikan manfaat nyata dalam membantu pemantauan kualitas air serta pemberian pakan ikan hias secara otomatis. Dengan dasar tersebut, peluncuran ini menjadi momen penting untuk menguji keberterimaan produk di lingkungan pengguna dan menjadi pijakan untuk pengembangan selanjutnya.

#### **6.2. DESKRIPSI**

Proses peluncuran produk *Smart Aquarium Ikan Hias* dirancang sebagai tahap awal pengenalan sistem kepada pengguna dalam lingkup terbatas, khususnya untuk kalangan akademik dan komunitas pecinta ikan hias. Peluncuran dilakukan dalam bentuk demonstrasi langsung di lingkungan kampus sebagai lokasi awal implementasi, dengan menampilkan prototipe sistem yang telah berfungsi penuh. Waktu peluncuran disesuaikan setelah seluruh tahap pengujian selesai dan produk dinyatakan stabil serta siap digunakan.

Strategi peluncuran yang digunakan adalah distribusi terbatas (*soft launch*) yang difokuskan pada demonstrasi langsung kepada dosen, mahasiswa, serta pengguna awal potensial. Sistem juga dipublikasikan melalui media digital berupa video dokumentasi dan tampilan real-time data sensor pada website yang telah dikembangkan. Dengan pendekatan ini, pengguna dapat memahami cara kerja sistem secara langsung dan mendapatkan pengalaman awal menggunakan produk.

Untuk mendukung keberhasilan peluncuran, telah disiapkan beberapa dokumen pendukung seperti panduan penggunaan (*user manual*), dokumentasi teknis, serta petunjuk instalasi

perangkat. Tim pengembang juga melakukan pendampingan saat peluncuran untuk menjelaskan fungsi-fungsi utama dan menangani pertanyaan atau kendala yang mungkin timbul dari pengguna awal. Seluruh persiapan ini bertujuan untuk memastikan bahwa pengguna dapat memahami dan menggunakan produk dengan mudah, serta memberikan umpan balik yang berguna untuk pengembangan selanjutnya.

### **6.3.DAYA GUNA PRODUK**

Produk *Smart Aquarium Ikan Hias* dirancang untuk memberikan kemudahan, efisiensi, dan otomatisasi dalam merawat ikan hias, terutama dalam hal pemantauan kualitas air dan pemberian pakan. Sistem ini memiliki sejumlah keunggulan yang memberikan nilai tambah bagi pengguna, baik dari segi fungsionalitas maupun kemudahan penggunaan.

Fitur utama produk ini antara lain:

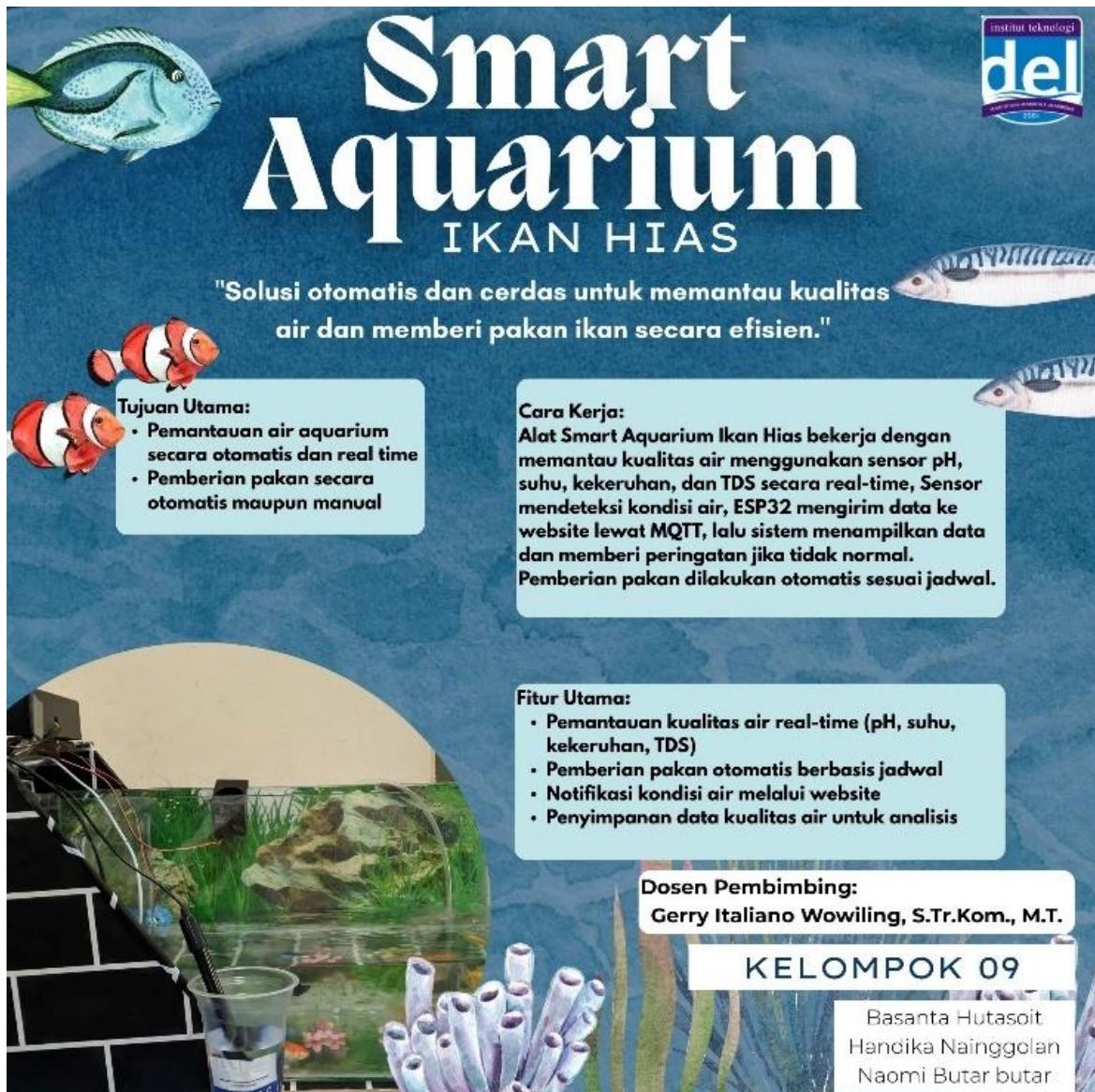
- Pemantauan kualitas air secara real-time melalui sensor pH, TDS, suhu, dan kekeruhan yang terintegrasi ke dalam sistem dan ditampilkan pada dashboard website.
- Pemberian pakan otomatis yang dikendalikan oleh waktu melalui RTC dan motor servo, sehingga pengguna tidak perlu memberi makan secara manual.
- Sistem peringatan dini jika parameter kualitas air tidak sesuai dengan nilai ideal, yang ditampilkan langsung di platform web.
- Tampilan antarmuka website yang sederhana dan mudah digunakan, memungkinkan pengguna mengakses informasi kapan saja dan dari mana saja.

Manfaat produk bagi pengguna akhir:

- Mengurangi risiko kematian ikan akibat kualitas air yang buruk atau keterlambatan pemberian pakan.
- Menghemat waktu dan tenaga dalam perawatan ikan, terutama bagi pengguna yang memiliki kesibukan tinggi.
- Memberikan kontrol dan informasi yang lebih akurat dibandingkan metode manual.

## 6.4. POSTER PRODUK

Bagian ini mencakup desain visual untuk promosi produk, seperti poster, brosur, atau materi pemasaran lainnya.



Gambar 6. 1 Poster Smart Aquarium Ikan Hias

## DAFTAR PUSTAKA

- Yohanes Sergio Sili, Dodit Suprianto, “Rancang Bangun Alat Pemberian Pakan Ikan Koki Otomatis Pada Aquarium Berbasis Mikrokontroler AT89S52”, 183765- ID-rancang-bangun-alat-pemberian-pakan-ikan.
- Lestari, D., Yuniarti, E., & Sari, Y. D. (2024). Sistem Monitoring Kualitas Air dan Pakan Otomatis Pada Akuarium Ikan Mas Koki Terintegrasi IoT. *BEES: Bulletin of Electrical and Electronics Engineering*, 4(3), 103–111.  
<https://doi.org/10.47065/bees.v4i3.4624>
- balebete baunajuku. (2019, 12 Agustus). *Budidaya dan Analisa Usaha Ikan Discus*.  
<https://balebetenajuku.blogspot.com/2019/08/budidaya-dan-analisa-usaha-ikan-discus.html>
- Widya, E. (2020, Maret 30). *Budidaya Ikan Hias Cupang*. Academia.edu.  
[https://www.academia.edu/42876612/\\_BUDIDAYA\\_IKAN\\_HIAS\\_CUPANG\\_MAKALAH\\_Di\\_susun\\_oleh\\_Kelas\\_XI\\_MIA\\_2\\_SUMITRA](https://www.academia.edu/42876612/_BUDIDAYA_IKAN_HIAS_CUPANG_MAKALAH_Di_susun_oleh_Kelas_XI_MIA_2_SUMITRA)
- Khoiru, A. (2020). *Rancang Bangun Monitoring dan Kontroling Kolam Ikan Koi Berbasis Internet of Things (IoT)*. Skripsi, Program Studi S1 Teknik Informatika, Fakultas Teknologi Informasi dan Komunikasi, Universitas Semarang. Tersedia di  
<https://eskripsi.usm.ac.id/files/skripsi/G21A/2015/G.211.15.0031/G.211.15.0031-15-File-Komplit-20200302021817.pdf>
- Jawas, H. (2013). *GloFish*. Makalah Biokimia I, Jurusan Pendidikan Kimia, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Pendidikan Ganesha. Diakses dari [https://kupdf.net/download/makalah-biokimia-i-quotglofishquot\\_59febdfbe2b6f54a58db35d1\\_pdf](https://kupdf.net/download/makalah-biokimia-i-quotglofishquot_59febdfbe2b6f54a58db35d1_pdf)
- Asmara, R.K.P. (2020), “Rancang Bangun Alat Monitoring Dan Penanganan Kualitas Ait Pada Akuarium Ikan Hias Berbasis Internet Of Things (IOT)”, Jurnal Teknik Elektro Dan Komputer TRIAC, Vol. 7 No. 2, pp. 69–74, doi: 10.21107/triac.v7i2.8148.
- Asmara, R.K.P. (2020), “Rancang Bangun Alat Monitoring Dan Penanganan Kualitas Ait Pada Akuarium Ikan Hias Berbasis Internet Of Things (IOT)”, Jurnal Teknik Elektro Dan Komputer TRIAC, Vol. 7 No. 2, pp. 69–74, doi: 10.21107/triac.v7i2.8148.
- Rusito, Ilham Febrianto, Iman Saufik, & Lukman Santoso. (2022). Perancangan Sistem Monitoring Kualitas Air Dan Kendali Pakan Aquarium Otomatis Berbasis IoT. Elkom : Jurnal Elektronika Dan Komputer, 15(2), 330–340.  
<https://doi.org/10.51903/elkom.v15i2.826>

Kurniatuty, S. A. (2019). Rancang Bangun Sistem Kontrol Pakan Ikan dan Kekeruhan Air yang Dilengkapi Dengan Monitoring Kualitas Air Berbasis Internet of Things (IoT). Institut Teknologi Nasional Malang.

Hayatunnufus, H., & Alita, D. (2020). Sistem Cerdas Pemberi Pakan Ikan Secara Otomatis. Jurnal Teknologi Dan Sistem Tertanam, 1(1), 11. <https://doi.org/10.33365/jtst.v1i1.799>

## **LAMPIRAN**