# **[HALAMAN SAMPUL LAPORAN PROYEK AKHIR](#_heading=h.gjdgxs)**

**LAPORAN PROYEK AKHIR**

****

**LAPORAN PENGEMBANGAN PRODUK PROYEK AKHIR**

**Prototype Alat Pembersih Sampah Mengapung Berbasis Mikrokontroler di Tepi Dalam Danau Toba**

**Disusun Oleh:**

|  |  |  |
| --- | --- | --- |
| **13323003** | **:** | **Ripandy Saragih** |
| **13323021** | **:** | **Grace Yosheva** |
| **13323042** | **:** | **Romaito A. Silaen** |

**PROGRAM STUDI D3 TEKNOLOGI KOMPUTER**

**FAKULTAS VOKASI**

# [**HALAMAN JUDUL LAPORAN PROYEK AKHIR**](#_heading=h.30j0zll)

**LAPORAN PROYEK AKHIR II**

****

**LAPORAN PENGEMBANGAN PRODUK PROYEK AKHIR**

**Prototype Alat Pembersih Sampah Mengapung Berbasis Mikrokontroler di Tepi Dalam Danau Toba**

**Disusun Oleh:**

|  |  |  |
| --- | --- | --- |
| **13323003** | **:** | **Ripandy Putra Juamanda Saragih** |
| **13323021** | **:** | **Grace Yosheva** |
| **13323042** | **:** | **Romaito A. Silaen** |

**PROGRAM STUDI D3 TEKNOLOGI KOMPUTER**

**FAKULTAS VOKASI**

**INSTITUT TEKNOLOGI DEL**

**2025**

**DAFTAR ISI**

[**HALAMAN SAMPUL LAPORAN PROYEK AKHIR** 1](#_Toc199164955)

[**HALAMAN JUDUL LAPORAN PROYEK AKHIR** 2](#_Toc199164956)

[**DAFTAR ISI** 4](#_Toc199164957)

[**DAFTAR TABEL** 16](#_Toc199164958)

[**DAFTAR GAMBAR** 17](#_Toc199164959)

[**DAFTAR RUMUS** 18](#_Toc199164960)

[**BAB I PRODUCT REQUIREMENT SPECIFICATION (SPESIFIKASI KEBUTUHAN PRODUK)** 19](#_Toc199164961)

[**1.1.** **PENDAHULUAN** 19](#_Toc199164962)

[1.1.1. Tujuan Penulisan Dokumen 19](#_Toc199164966)

[1.1.2. Latar Belakang Produk 20](#_Toc199164968)

[1.1.3. Tujuan Produk 21](#_Toc199164969)

[1.1.4. Lingkup Produk 21](#_Toc199164976)

[1.1.5. Manfaat Produk 22](#_Toc199164984)

[1.1.6. Definisi dan Singkatan 22](#_Toc199164991)

[1.1.7. Referensi 23](#_Toc199164993)

[**1.2.** **DESKRIPSI UMUM PRODUK** 24](#_Toc199164999)

[1.2.1. Permasalahan dan Persoalan 24](#_Toc199165001)

[1.2.2. Produk yang menjadi Inspirasi 25](#_Toc199165009)

[1.2.3. Produk yang akan dibangun 26](#_Toc199165014)

[1.2.4. Konteks 27](#_Toc199165016)

[1.2.5. Deskripsi Kebutuhan Produk 28](#_Toc199165019)

[1.2.6. Environment Hardware dan Software 28](#_Toc199165029)

[1.2.7. Metodologi dan Tools Pengembangan 29](#_Toc199165032)

[1.2.7.1. Metodologi 29](#_Toc199165034)

[1.2.7.2. Tools Pengembangan 30](#_Toc199165035)

[**BAB II PROJECT PLANNING (PP) (PERENCANAAN PENGEMBANGAN PRODUK)** 32](#_Toc199165036)

[**2.1. PENDAHULUAN** 32](#_Toc199165037)

[**2.2. DESKRIPSI PROYEK** 32](#_Toc199165040)

[2.2.1. Project Organization 33](#_Toc199165042)

[2.2.2. Work Breakdown Structure (WBS) 34](#_Toc199165043)

[2.2.3. Budget 36](#_Toc199165044)

[2.2.4. Tools 38](#_Toc199165045)

[2.2.5. Resiko dan Hambatan 40](#_Toc199165046)

[2.2.5.1 Daftar Resiko Utama 40](#_Toc199165048)

[2.2.5.2. Hambatan Eksternal dan Strategi Antisipasi 41](#_Toc199165049)

[**BAB III PRODUCT DESIGN (PD) (DESAIN PENGEMBANGAN PRODUK)** 43](#_Toc199165050)

[**3.1. PENDAHULUAN** 43](#_Toc199165051)

[**3.2. DESKRIPSI PRODUK DESIGN** 44](#_Toc199165061)

[3.2.1. Proses Bisnis Current System 44](#_Toc199165062)

[3.2.2. Proses Bisnis Target System 44](#_Toc199165064)

[3.2.3. User Interface Layout [Software] 45](#_Toc199165066)

[3.2.4. Arsitektur Sistem [Hardware] 47](#_Toc199165067)

[3.2.5. Desain Rangkaian Elektronik/Skematik Desain dan Desain PCB [Hardware] 48](#_Toc199165069)

[3.2.5.1. Disain Skematik 48](#_Toc199165070)

[3.2.5.2. Disain PCB 49](#_Toc199165076)

[3.2.6. Mekanisme Komunikasi Data [Hardware] 51](#_Toc199165077)

[3.2.7. Desain Fisik/Case Hardware 53](#_Toc199165078)

[3.2.7.1. Desain Fisik Keseluruhan 53](#_Toc199165079)

[3.2.7.2. Sayap Kiri dan Kanan 54](#_Toc199165082)

[3.2.7.3. Bagian Atas 55](#_Toc199165085)

[3.2.7.4. Dinding Samping Kiri dan Kanan 56](#_Toc199165086)

[3.2.7.5. Baling-Baling 57](#_Toc199165088)

[3.2.7.6. Pipa Kiri dan Kanan 58](#_Toc199165090)

[3.2.7.7. Penahan Jaring Atas Kiri dan Kanan 59](#_Toc199165093)

[3.2.7.8. Pipa yang Disambung 60](#_Toc199165096)

[3.2.7.9. Case Bagian Badan untuk Komponen Elektronika 61](#_Toc199165097)

[3.2.7.10. Tutup Case 62](#_Toc199165098)

[**BAB IV** 64](#_Toc199165099)

[**PRODUCT IMPLEMENTATION (PI)** 64](#_Toc199165100)

[**(IMPLEMENTASI PENGEMBANGAN PRODUK)** 64](#_Toc199165101)

[**4.1. PENDAHULUAN** 64](#_Toc199165102)

[**4.2. DESKRIPSI** 64](#_Toc199165103)

[4.2.1. Prinsip Implementasi 64](#_Toc199165104)

[4.2.2. Lingkungan Pengembangan Terintegrasi (IDE) 64](#_Toc199165105)

[4.2.3. Implementasi Desain PCB 65](#_Toc199165106)

[4.2.4. Implementasi Desain Fisik/Desain Case Hardware 65](#_Toc199165107)

[4.2.5. Integrasi Hardware dan Software 66](#_Toc199165108)

[4.2.6. Pengujian dan Evaluasi Sistem 75](#_Toc199165109)

[**BAB V** 76](#_Toc199165110)

[**PENGUJIAN PRODUK (PRODUCT TESTING)** 76](#_Toc199165111)

[**5.1. PENDAHULUAN** 76](#_Toc199165112)

[**5.2. DESKRIPSI PENGUJIAN** 76](#_Toc199165113)

[5.2.1 Pengujian Motor DC 76](#_Toc199165114)

[5.2.2 Pengujian Ketahanan Baterai 77](#_Toc199165115)

[5.2.3 Pegujian Motor BLDC 78](#_Toc199165116)

[5.2.4 Tabel Pengujian Sensor Ultrasonik 80](#_Toc199165117)

[**5.3. METODE PENGUJIAN** 81](#_Toc199165118)

[**5.3.1. Pengujian Fungsional** 81](#_Toc199165119)

[**5.3.2. Pengujian Non-Fungsional** 83](#_Toc199165120)

[**5.3.3. Pengujian Hardware** 84](#_Toc199165121)

[**5.3.4. Pengujian Integrasi Software dan Hardware** 84](#_Toc199165122)

[**5.3.5. Pengujian Prototipe** 85](#_Toc199165123)

[**5.4. KESIMPULAN** 86](#_Toc199165124)

[**BAB VI** 88](#_Toc199165125)

[**PRODUCT RELEASE (PR)** 88](#_Toc199165126)

[**PELUNCURAN PRODUK** 88](#_Toc199165127)

[**6.1. PENDAHULUAN** 88](#_Toc199165128)

[**6.2. DESKRIPSI** 88](#_Toc199165132)

[**6.3. DAYA GUNA PRODUK** 89](#_Toc199165133)

[**6.4. POSTER PRODUK** 91](#_Toc199165134)

[**DAFTAR PUSTAKA** 93](#_Toc199165135)

**DAFTAR TABEL**

[Tabel 1 Daftar Definisi dan Singkatan 12](#_Toc199054330)

[Tabel 2 Environmentx Hardware dan Software 18](#_Toc199054331)

[Tabel 3 Budget yang diperlukan 25](#_Toc199054332)

[Tabel 4 Tools Hardware 27](#_Toc199054333)

[Tabel 5 Tools Software 28](#_Toc199054334)

[Tabel 6 Daftar Resiko Utama 29](#_Toc199054335)

[Tabel 7 Pengujian dan Evaluasi Sistem 58](#_Toc199054336)

[Tabel 8 Hasil Pengujian Kecepatan Motor DC 59](#_Toc199054337)

[Tabel 9 Hasil Pengujian Ketahanan Baterai 61](#_Toc199054338)

[Tabel 10 Hasil Pengujian Motor BLDC Kiri 62](#_Toc199054339)

[Tabel 11 Hasil Pengujian Motor BLDC Kanan 62](#_Toc199054340)

[Tabel 12 Hasil Pengujian Sensor Ultrasonik 64](#_Toc199054341)

[Tabel 13 Hasil Pengujian Fungsional Motor Dc 65](#_Toc199054342)

[Tabel 14 Hasil Pengujian Non-Fungsional Ketahanan Baterai 66](#_Toc199054343)

[Tabel 15 Hasil Pengujian Hardware Motor BLDC 67](#_Toc199054344)

[Tabel 16 Hasil Pengujian Integrasi 67](#_Toc199054345)

[Tabel 17 Hasil Pengujian Prototipe 68](#_Toc199054346)

# **DAFTAR GAMBAR**

[Gambar 1 Blok Diagram 17](#_Toc199396788)

[Gambar 2 Project Organization 24](#_Toc199396789)

[Gambar 3 Work Breakdown Structure 25](#_Toc199396790)

[Gambar 4 Current System 33](#_Toc199396791)

[Gambar 5 Target System 34](#_Toc199396792)

[Gambar 6 User Interface Layout 35](#_Toc199396793)

[Gambar 7 Arsitektur Sistem [Hardware] 38](#_Toc199396794)

[Gambar 8 Desain Skematik 39](#_Toc199396795)

[Gambar 9 Rangkaian Mikrokontroller Esp8266 Terhubung ke Sumber Arus 41](#_Toc199396796)

[Gambar 10 Rangkaian Mikrokontroller Esp8266 Terhubung ke Motor DC 41](#_Toc199396797)

[Gambar 11 Rangkaian Esp8266 Terhubung ke Sensor Ultrasonik 42](#_Toc199396798)

[Gambar 12 Rangkaian Keseluruhan 42](#_Toc199396799)

[Gambar 13 Mekanisme Komunikasi Data [Hardware] 43](#_Toc199396800)

[Gambar 15 Desain Fisik Keseluruhan 44](#_Toc199396801)

[Gambar 16 Desain Sayap Kiri dan Kanan 45](#_Toc199396802)

[Gambar 17 Desain Bagian Atas 46](#_Toc199396803)

[Gambar 18 Desain Dinding Samping Kiri dan Kanan 47](#_Toc199396804)

[Gambar 19 Desain Baling - Baling 48](#_Toc199396805)

[Gambar 20 Desain Pipa Utama 49](#_Toc199396806)

[Gambar 21 Desain Penahan Jaring Atas Kiri dan Kanan 50](#_Toc199396807)

[Gambar 22 Desain Pipa yang Disambung 51](#_Toc199396808)

[Gambar 23 Desain Case Bagian Badan 52](#_Toc199396809)

[Gambar 24 Tutup Case 53](#_Toc199396810)

[Gambar 25 Poster Produk 82](#_Toc199396811)

# **DAFTAR RUMUS**

[Rumus 1 Pengujian Ketahanan Baterai 60](#_Toc199054660)

[Rumus 2 Durasi Daya Tahan Baterai 60](#_Toc199054661)

[Rumus 3 Polaristas Tegangan 63](#_Toc199054662)

[Rumus 4 Hukum Ohm 63](#_Toc199054663)

[Rumus 5 Kecepatan Rotasi Motor 65](#_Toc199054664)

[Rumus 6 Daya Baterai 66](#_Toc199054665)

[Rumus 7 Waktu Operasi 69](#_Toc199054666)

# **BAB I PRODUCT REQUIREMENT SPECIFICATION (SPESIFIKASI KEBUTUHAN PRODUK)**

# **PENDAHULUAN**

## Danau Toba merupakan salah satu danau terbesar dan terdalam di dunia, yang terletak di Provinsi Sumatra Utara, Indonesia. Selain memiliki keindahan alam yang luar biasa, Danau Toba juga berfungsi sebagai sumber daya alam yang penting, baik dari segi ekosistem maupun potensi pariwisata. Namun, dalam beberapa tahun terakhir, Danau Toba menghadapi masalah serius terkait pencemaran sampah, baik sampah plastik, sisa-sisa makanan, maupun limbah lainnya yang mengapung di permukaan air atau terdampar di sepanjang bibir danau. Sampah-sampah ini tidak hanya merusak keindahan alam, tetapi juga dapat mempengaruhi kualitas air dan mengancam keberlanjutan ekosistem yang ada, seperti ikan dan tanaman air.

## Pencemaran sampah di Danau Toba juga berhubungan dengan tingginya volume wisatawan yang berkunjung dan kurangnya kesadaran masyarakat serta pengunjung tentang pentingnya menjaga kebersihan dan kelestarian lingkungan danau. Oleh karena itu, untuk menjaga keberlanjutan Danau Toba sebagai destinasi wisata alam dan pelestarian ekosistem, sangat diperlukan upaya untuk mengurangi dan membersihkan sampah yang ada di danau secara berkelanjutan. Salah satu solusi yang potensial adalah dengan mengembangkan teknologi pembersihan sampah yang efisien, ramah lingkungan, dan tidak mengganggu keseimbangan ekosistem air.

## Dalam konteks ini, *prototype* sistem pembersih sampah menjadi pilihan yang tepat. Sistem ini dirancang untuk mengurangi keterlibatan manusia dalam proses pembersihan sampah, menggunakan teknologi yang dapat bekerja secara efisien dan terintegrasi dengan antarmuka berbasis web yang mudah diakses melalui jaringan *Wi-Fi* lokal yang disediakan oleh perangkat.

## Tujuan Penulisan Dokumen

### Dokumen ini bertujuan untuk :

1. Menguraikan kebutuhan produk secara terstruktur dan sistematis.
2. Memberikan panduan bagi tim pengembang dalam implementasi sistem.
3. Memastikan seluruh pihak yang terlibat memahami tujuan proyek dengan jelas.
4. Meningkatkan efisiensi proses pembersihan sampah di tepi danau.
5. Mengembangkan solusi berbasis teknologi yang dapat bekerja secara manual dalam mengumpulkan sampah.
6. Mendukung upaya konservasi lingkungan dengan penerapan teknologi ramah lingkungan.

Dokumen ini menjadi referensi utama dalam pengembangan, implementasi, dan evaluasi *Prototype Alat Pembersih Sampah Mengapung Berbasis Mikrokontroler di Tepi Danau Toba*, memastikan kesesuaian antara kebuuhan pengguna dan produk yang akan dihasilkan

## Latar Belakang Produk

Pencemaran lingkungan di sekitar Danau Toba menjadi permasalahan serius akibat akumulasi sampah di area tepi dan perairan danau. Sampah plastik dan organik yang menumpuk dapat mencemari ekosistem air, mengganggu kehidupan biota, serta mengurangi daya tarik wisata. Jika tidak segera ditangani, pencemaran ini dapat berdampak negatif pada ekologi dan ekonomi masyarakat sekitar. Beberapa permasalahan utama yang menjadi latar belakang pengembangan sistem ini adalah:

* Kesulitan dalam membersihkan sampah secara manual di tepi danau : Proses manual membutuhkan banyak tenaga kerja dan waktu.
* Kurangnya tenaga kerja yang tersedia untuk melakukan pembersihan secara berkala : Tidak ada sistem yang mendukung proses pembersihan secara efisien.
* Tidak adanya sistem yang dapat membantu pengelolaan sampah di perairan dangkal : Sampah yang terbawa arus sering kali sulit dijangkau oleh tenaga manusia.

Dengan adanya *Prototype Alat Pembersih Sampah Mengapung Berbasis Mikrokontroler di Tepi Dalam Danau Toba*, diharapkan proses pembersihan sampah dapat dilakukan secara lebih efisien dan efektif. Sistem ini dirancang untuk mengambil sampah menggunakan mekanisme jaring yang digerakkan oleh kapal kecil. Pengendalian arah kapal dilakukan melalui antarmuka web yang diakses melalui jaringan *Wi-Fi* lokal yang disediakan oleh *ESP8266* dengan *IP address 192.168.4.1*, sehingga pengguna dapat menggerakkan kapal ke kiri, kanan, maju, atau mundur untuk mendekati sampah. Sistem ini juga dilengkapi dengan 2 sensor ultrasonik untuk mendeteksi jarak kapal dengan sampah atau hambatan, memberikan informasi jarak pada antarmuka *web* untuk membantu navigasi manual. Tombol *on/off* sistem berada langsung di kapal, memungkinkan aktivasi dan deaktivasi sistem secara manual. Teknologi ini diharapkan dapat berkontribusi dalam menjaga kebersihan tepi Danau Toba dan mendukung upaya pelestarian lingkungan di sekitarnya.

## Tujuan Produk

### Tujuan dari pengembangan ***Prototype Alat Pembersih Sampah Mengapung Berbasis Mikrokontroler di Tepi Dalam Danau Toba*** adalah:

### Menciptakan sistem berbasis *IoT* yang dapat mendeteksi dan mengumpulkan sampah secara manual di area tepi Danau Toba.

### Mengurangi ketergantungan pada pembersihan manual yang membutuhkan banyak tenaga kerja dan waktu.

### Memungkinkan pengguna untuk mengoperasikan sistem melalui kontrol berbasis *web*, termasuk tombol *on/off* dan pengaturan arah gerak (ke kiri dan kanan).

### Mendukung upaya konservasi lingkungan dengan penerapan teknologi yang efisien dan ramah lingkungan.

1. Menyediakan informasi jarak dari sensor ultrasonik pada antarmuka *web* untuk membantu navigasi manual.

## Lingkup Produk

### Ruang lingkup dalam pengembangan ***Prototype Alat Pembersih Sampah Mengapung Berbasis Mikrokontroler di Tepi Dalam Danau Toba*** mencakup:

### Pengumpulan Sampah Manual: Sistem menggunakan jaring untuk menangkap dan mengangkut sampah ke atas permukaan, dengan kapal yang dikendalikan secara manual oleh pengguna untuk mendekati sampah.

### Penggunaan Sensor Ultrasonik: Dua sensor ultrasonik digunakan untuk mendeteksi jarak kapal dengan sampah atau hambatan, memberikan informasi jarak pada antarmuka *web* untuk membantu pengguna dalam navigasi.

### Batasan Penggunaan: Sistem ini dirancang untuk beroperasi di tepi perairan Danau Toba.

### Pengguna Akhir: Sistem ini akan digunakan oleh pemerintah daerah, komunitas lingkungan, dan pengelola wisata untuk menjaga kebersihan Danau Toba.

### Sistem Kontrol Berbasis Web: Pengendalian kapal dilakukan melalui antarmuka *web* yang diakses melalui jaringan *Wi-Fi* lokal yang disediakan oleh *ESP8266* dengan *IP address 192.168.4.1*, memungkinkan pengguna untuk menggerakkan kapal ke arah yang diinginkan untuk mendekati sampah. Tombol *on/off* sistem berada langsung di kapal untuk aktivasi manual.

### Dengan adanya batasan ini, sistem yang dikembangkan dapat berfungsi secara optimal sesuai dengan kebutuhan dan kondisi lingkungan Danau Toba.

## Manfaat Produk

### Produk ini diharapkan memberikan manfaat dari berbagai aspek, termasuk teknis, ekonomis, dan lingkungan:

### **Teknis** : Memastikan pembersihan sampah di tepi danau berjalan secara manual dengan pemantauan dan kontrol berbasis *web*.

### **Ekonomis** : Mengurangi biaya operasional yang dikeluarkan untuk tenaga kerja dalam membersihkan sampah secara manual.

### **Lingkungan** : Membantu menjaga kebersihan Danau Toba dan melestarikan ekosistem air dengan mengurangi pencemaran sampah.

### **Pengguna** : Memberikan kemudahan bagi pengelola wisata dan komunitas lingkungan dalam mengawasi dan mengontrol proses pembersihan sampah dengan teknologi yang efisien.

### Dengan manfaat ini, ***Prototype Alat Pembersih Sampah Mengapung Berbasis Mikrokontroler di Tepi Dalam Danau Toba*** dapat menjadi solusi yang inovatif dalam menjaga kebersihan dan kelestarian lingkungan di sekitar Danau Toba.

## Definisi dan Singkatan

### Bagian ini berisi daftar istilah teknis dan singkatan yang digunakan dalam dokumen ini untuk memberikan pemahaman yang lebih jelas kepada pembaca.

Tabel 1 Daftar Definisi dan Singkatan

| **No** | **Definisi/Singkatan** | **Keterangan** |
| --- | --- | --- |
| 1 | Motor DC | Motor ini digunakan untuk menggerakkan berbagai perangkat seperti kapal pembersih sampah. |
| 2 | pH | Potensial Hidrogen, ukuran keasaman atau kebasaan larutan. |
| 3 | IoT | *Internet of Things*, konsep konektivitas perangkat melalui internet untuk pengawasan dan kontrol jarak jauh. |
| 4 | Web Monitoring | Sistem pemantauan berbasis *web* yang memungkinkan pengguna mengontrol dan melihat data secara *real-time*. |
| 5 | Wemos ESP8266 | Mikrokontroler berbasis *WiFi* yang digunakan untuk komunikasi *IoT*. |
| 6 | WiFi | Teknologi komunikasi nirkabel yang digunakan untuk konektivitas sistem. |
| 7 | Hardware | Komponen fisik dari sistem yang digunakan dalam *prototype*. |
| 8 | Mikrokontroler | Unit pemrosesan utama dalam sistem elektronik yang digunakan untuk mengontrol perangkat. |
| 9 | Aktuator | Perangkat mekanis yang mengubah sinyal listrik menjadi gerakan fisik dalam sistem manual. |

## Referensi

1. Motor DC

* *Almira Fanny, A., & Erwin, J. (2023)* : Menggunakan motor DC sebagai penggerak utama kapal pembersih kolam yang dikendalikan secara manual melalui remote kontrol.
* *Antony, K. et al. (2024)* : Motor DC berfungsi sebagai aktuator penggerak robot autonomus untuk bergerak dan bermanuver di kolam ikan hias.
* *Makassar, P. A. T. I. (2022)* : Motor DC digunakan untuk menggerakkan robot di permukaan kolam, sebagai bagian dari sistem pembersih sampah.
* *Wijaya, A., & Juliadi, D. (2021)* :Motor DC digunakan untuk pergerakan robot pembersih lantai, dikendalikan melalui aplikasi Android.

1. Sensor Ultrasonik

* *Antony, K. et al. (2024)* : Menggunakan sensor ultrasonik untuk membantu navigasi robot dalam kolam, khususnya menghindari hambatan.
* *Makassar, P. A. T. I. (2022)* :Kemungkinan menggunakan sensor ultrasonik sebagai pendeteksi jarak, meskipun tidak dijelaskan secara rinci.

# **DESKRIPSI UMUM PRODUK**

## Bagian ini menjelaskan gambaran umum tentang produk yang dikembangkan, mencakup latar belakang, tujuan, serta cakupan sistem yang akan dibangun. Deskripsi ini bertujuan untuk memberikan pemahaman awal mengenai alasan pengembangan produk, permasalahan yang dihadapi, serta solusi yang ditawarkan. Dengan adanya sistem ini, diharapkan dapat memberikan efisiensi dalam proses pembersihan sampah di tepi Danau Toba melalui pendekatan berbasis teknologi *IoT*.

## Permasalahan dan Persoalan

### Bagian ini menguraikan latar belakang masalah atau kebutuhan yang mendasari pengembangan produk. Fokus utama adalah menjelaskan permasalahan yang dihadapi pengguna serta alasan mengapa solusi ini dibutuhkan.

### *Permasalahan* **:** Pengelolaan kebersihan di tepi Danau Toba masih dilakukan secara manual, yang memerlukan tenaga kerja yang cukup banyak dan waktu yang lama. Sampah yang menumpuk di perairan dan tepi danau dapat mencemari lingkungan, mengganggu ekosistem air, serta menurunkan daya tarik wisata.

### *Persoalan* **:** Ketidakefisienan dalam proses pembersihan disebabkan oleh beberapa faktor, antara lain:

### Kurangnya tenaga kerja : Pembersihan manual membutuhkan banyak tenaga kerja dan sulit dilakukan secara rutin, terutama di area yang luas.

### Aksesibilitas yang terbatas : Beberapa area di tepi danau sulit dijangkau oleh petugas kebersihan, sehingga sampah menumpuk dalam jangka waktu lama.

### Kurangnya monitoring real-time : Tidak adanya sistem pemantauan otomatis menyebabkan sulitnya mendeteksi lokasi sampah yang menumpuk secara cepat.

### Ketergantungan pada metode konvensional : Penggunaan alat tradisional kurang efektif dalam menangani volume sampah yang terus meningkat akibat aktivitas wisata dan masyarakat sekitar.

Untuk mengatasi permasalahan ini, dikembangkan *Prototipe Sistem Pembersih Sampah Berbasis IoT* yang memungkinkan pengendalian perangkat secara manual melalui web. Sistem ini menggunakan *ESP8266* sebagai *Access Point* yang menyediakan jaringan *Wi-Fi* lokal. Pengguna dapat mengakses antarmuka web melalui *IP address 192.168.4.1* untuk mengendalikan kapal dalam mengumpulkan sampah. Dua sensor ultrasonik digunakan untuk memberikan informasi jarak yang ditampilkan pada antarmuka web, membantu pengguna dalam navigasi. Dengan adanya sistem ini, diharapkan pengelolaan sampah di tepi Danau Toba dapat dilakukan dengan lebih efisien, mengurangi beban tenaga kerja, serta meningkatkan kebersihan lingkungan secara berkelanjutan.

## Produk yang menjadi Inspirasi

### Dalam pengembangan proyek ini, beberapa produk serupa telah menjadi inspirasi. Salah satu produk yang relevan adalah **"*Smart Trash Klasifikasi Sampah Otomatis Dengan Sensor Proximity Berbasis Arduino*"** yang dikembangkan oleh Anas, Hikmah, dan Aprilia (2023). Sistem ini menggunakan sensor proximity untuk mendeteksi keberadaan objek dan mengklasifikasikan sampah secara otomatis. Dari penelitian ini, terlihat bahwa sensor proximity memiliki keunggulan dalam mendeteksi objek tanpa memerlukan kontak langsung, yang dapat diterapkan dalam proyek ini untuk meningkatkan efisiensi sistem.

### Selain itu, penelitian oleh ***Antony, Hernando, Caniago, Adi, dan Aritonang (2024)*** tentang **"*Rancang Bangun Robot Autonomus Pembersih Kolam Ikan Hias*"** juga menjadi referensi dalam pengembangan proyek ini. Robot tersebut menggunakan sistem otomatis untuk mendeteksi dan membersihkan kolam ikan secara efisien. Kekurangan dari penelitian ini adalah kurangnya integrasi dengan sistem berbasis web untuk pemantauan dan kontrol secara langsung oleh pengguna. Oleh karena itu, dalam proyek ini akan dikembangkan fitur kontrol melalui web, sehingga pengguna dapat mengoperasikan sistem secara lebih fleksibel.

### Penelitian lainnya yang menjadi inspirasi adalah **"*Rancang Bangun Sistem Keamanan Otomatis dan Pengendali Mobil Listrik Berbasis Internet of Things*"** oleh *Zega, Fauzi, dan Mardiana (2023)*. Studi ini menunjukkan bagaimana IoT dapat digunakan untuk mengontrol perangkat secara jarak jauh, yang mendukung konsep sistem yang dikembangkan dalam proyek ini, yaitu pengoperasian perangkat melalui web.

### Berdasarkan penelitian-penelitian tersebut, proyek ini mengadaptasi beberapa fitur unggulan seperti penggunaan sensor proximity untuk deteksi objek dan pemantauan serta pengendalian perangkat melalui web. Dengan mengatasi keterbatasan produk-produk sebelumnya, diharapkan sistem yang dikembangkan lebih efisien dan mudah digunakan.

## Produk yang akan dibangun

### *Prototype* yang akan dibangun adalah sistem pembersih sampah otomatis berbasis *IoT* di Tepi Danau Toba dengan fitur utama sebagai berikut:

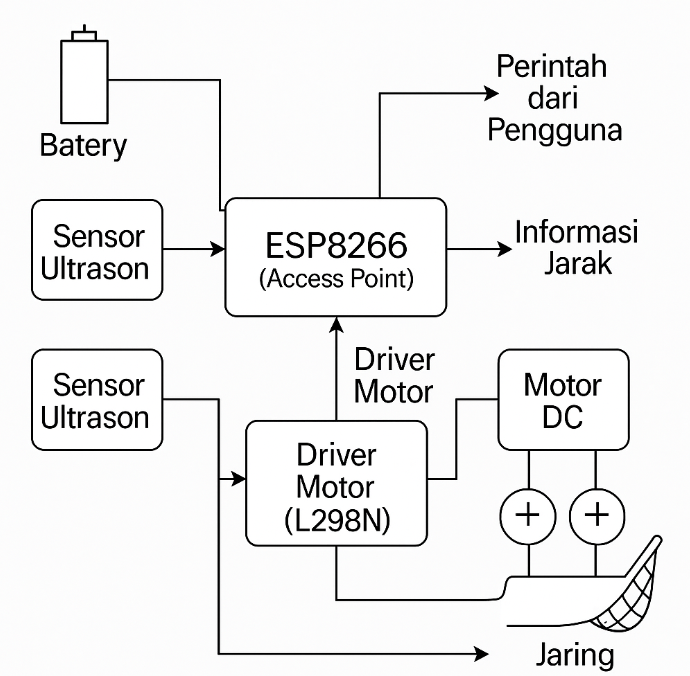
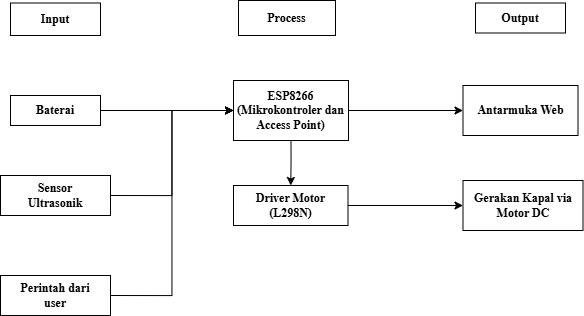
1. **Pengendalian jarak jauh melalui *web*** : Menggunakan *NodeMCU ESP8266* sebagai *Access Point* yang menyediakan jaringan *Wi-Fi* lokal dengan *IP address 192.168.4.1*, memungkinkan pengguna menggerakkan kapal (maju, mundur, kiri, kanan) untuk mendekati sampah.
2. **Penggunaan Sensor Ultrasonik** : Dua sensor ultrasonik untuk mendeteksi jarak kapal dengan sampah atau hambatan, menampilkan informasi jarak pada antarmuka *web* untuk membantu navigasi manual.
3. **Sistem on/off manual di kapal** : Memungkinkan pengguna mengaktifkan atau mematikan sistem secara langsung.
4. **Desain modular** : Memungkinkan perangkat dapat diperbaiki atau dimodifikasi dengan mudah sesuai kebutuhan pengguna.

Nilai tambah dari prototipe ini adalah kemudahan kontrol melalui antarmuka *web* yang diakses melalui jaringan *Wi-Fi* lokal, ditambah informasi jarak dari sensor ultrasonik untuk navigasi yang lebih akurat, sehingga pengguna dapat mengoperasikan kapal menggunakan perangkat mobile atau komputer tanpa ketergantungan pada server eksternal.

## Konteks

### Sistem ini akan diterapkan di tepi Danau Toba, khususnya di area yang sering mengalami penumpukan sampah atau lumut di permukaan air. Dengan kondisi lingkungan yang luas dan sulit dijangkau secara manual, penggunaan prototipe pembersih ini akan membantu mengurangi pencemaran air dan meningkatkan efisiensi pembersihan

### Berikut merupakan diagram blok untuk perancangan alat yang dapat disimpulkan dari berbagai referensi yang ada:



Gambar 1 Blok Diagram



Gambar 6 User Interface Layout

Antarmuka aplikasi menampilkan tata letak tombol navigasi yang menyerupai bentuk *D-Pad.* Antarmuka aplikasi ini dirancang sederhana untuk memudahkan pengguna dalam

Gambar diatas menunjukkan alur sederhana sistem ini:

1. *Input* : Meliputi Baterai (menyediakan daya untuk sistem, tahan hingga 1 jam 10 menit di air), Sensor Ultrasonik (mendeteksi jarak ke sampah atau hambatan dengan mengukur waktu pantulan gelombang ultrasonik, mulai deteksi pada 15 cm), dan Perintah Pengguna (dikirim melalui antarmuka web via jaringan Wi-Fi lokal dengan IP 192.168.4.1).
2. *Process* : Dilakukan oleh ESP8266 (otak sistem yang mengolah data jarak dari sensor ultrasonik, menyediakan jaringan Wi-Fi lokal sebagai Access Point dengan SSID 'Group-07' dan kata sandi '12345678', serta mengatur driver motor berdasarkan perintah dari web).
3. *Output* : Hasilnya adalah Gerakan Kapal via Motor DC (menggerakkan kapal untuk mengumpulkan sampah dengan jaring berdasarkan perintah maju, mundur, kiri, kanan, atau stop) dan Data Jarak di Antarmuka Web (menampilkan jarak secara real-time untuk navigasi manual).

## Sistem ini bekerja dengan sensor ultrasonik yang mengukur jarak ke sampah, ESP8266 yang memproses informasi dan mengendalikan motor melalui driver berdasarkan perintah pengguna via web, serta baterai yang memberi daya, sehingga kapal dapat membersihkan sampah secara efisien di tepi Danau Toba.

## Deskripsi Kebutuhan Produk

### Kebutuhan fungsional:

### Pemantauan kondisi kapal melalui antarmuka *web*, termasuk informasi jarak dari sensor ultrasonik.

### Kontrol pergerakan kapal secara jarak jauh (maju, mundur, kiri, kanan) untuk mendekati sampah.

### Sistem pembersihan berbasis motor DC untuk mengangkat sampah dari permukaan air menggunakan jaring.

### Penggunaan dua sensor ultrasonik untuk mendeteksi jarak kapal dengan sampah atau hambatan.

### Kebutuhan non-fungsional:

### Respon cepat, kurang dari 5 detik untuk perintah dari *web*.

### Desain tahan air, agar dapat beroperasi di lingkungan tepi Danau Toba dengan kondisi air yang bervariasi.

### Daya tahan tinggi, agar dapat beroperasi dalam waktu lama tanpa sering mengalami gangguan atau kerusakan.

## Environment Hardware dan Software

### Lingkungan perangkat keras dan perangkat lunak merupakan bagian penting dalam pengembangan, pengujian, dan implementasi sistem prototipe yang akan dibangun. Perangkat keras mencakup komponen fisik seperti mikrokontroler, sensor, dan aktuator, sedangkan perangkat lunak mencakup tools pemrograman, platform pengendalian, serta komunikasi data.

### Pada sistem ini, berbagai komponen perangkat keras dan perangkat lunak digunakan untuk memastikan bahwa prototipe dapat berfungsi secara optimal. Berikut adalah daftar *environment hardware* dan *software* yang digunakan:

Tabel 2 Environmentx Hardware dan Software

| **No** | **Hardware/Software** | **Spesifikasi atau Keterangan** |
| --- | --- | --- |
| 1 | Mikrokontroler (ESP8266) | Mengontrol sensor dan aktuator serta mengirimkan data ke server atau aplikasi berbasis *web*. |
| 2 | Sensor Ultrasonik | Digunakan untuk mendeteksi jarak antara prototipe dengan sampah. |
| 3 | Motor DC | Berfungsi sebagai penggerak utama prototipe di dalam air. |
| 5 | Wi-Fi (ESP8266) | Memungkinkan konektivitas prototipe dengan server berbasis *web* untuk pemantauan dan kontrol jarak jauh. |
| 6 | Web Server | Digunakan untuk mengontrol pergerakan prototipe dan memantau statusnya melalui tampilan *web*. |
| 7 | Arduino IDE | Digunakan untuk mengembangkan dan mengunggah kode ke mikrokontroler ESP8266. |
| 8 | VS Code | Editor kode yang digunakan dalam pengembangan sistem berbasis *web* untuk prototipe. |
| 9 | SolidWorks | Digunakan untuk merancang desain fisik prototipe sebelum pembuatan. |

## Metodologi dan Tools Pengembangan

### Bagian ini menjelaskan metodologi yang digunakan dalam pengembangan sistem serta tools yang mendukung proses tersebut. Pemilihan metodologi bertujuan untuk memastikan sistem dikembangkan secara efisien dan adaptif sesuai dengan kebutuhan proyek.

### Metodologi

Metode yang digunakan dalam proyek akhir ini adalah metode ***Agile***, yang memungkinkan pengembangan sistem secara iteratif dan adaptif. *Agile* dipilih karena fleksibilitasnya dalam menangani perubahan kebutuhan dan memungkinkan pengembang untuk terus meningkatkan sistem berdasarkan umpan balik yang diterima.

Tahapan utama dalam metodologi *Agile* yang diterapkan dalam proyek ini meliputi:

1. Perencanaan kebutuhan : Mengidentifikasi kebutuhan sistem dan menentukan ruang lingkup proyek.
2. Desain prototipe awal : Merancang sistem awal, termasuk desain perangkat keras dan antarmuka *web*.
3. Membuat kode program : Mengembangkan perangkat lunak dan perangkat keras secara bertahap dengan pengujian berkala.
4. Pengujian dan evaluasi : Melakukan pengujian pada setiap iterasi untuk memastikan sistem bekerja sesuai dengan spesifikasi.
5. Iterasi perbaikan berdasarkan *feedback* : Memperbaiki dan meningkatkan sistem berdasarkan hasil pengujian dan umpan balik pengguna.

Metode *Agile* memastikan bahwa pengembangan sistem dapat dilakukan secara dinamis dengan perbaikan berkelanjutan sehingga sistem yang dihasilkan lebih sesuai dengan kebutuhan pengguna dan lingkungan operasionalnya.

### Tools Pengembangan

Dalam proses pengembangan sistem ini, diperlukan berbagai perangkat lunak dan alat bantu untuk membantu dalam perancangan, implementasi, pengujian, serta validasi sistem. *Tools* pengembangan ini digunakan selama tahap pembuatan sistem dan bukan merupakan bagian dari produk akhir. Pemilihan *tools* dilakukan berdasarkan kebutuhan proyek, baik untuk desain perangkat keras maupun pengembangan sistem berbasis *web* yang digunakan untuk pemantauan dan kontrol.

Berikut adalah daftar *tools* yang digunakan dalam pengembangan prototipe sistem pembersih sampah di tepi Danau Toba:

1. ***Fritzing/Proteus*** : Digunakan untuk merancang dan mensimulasikan rangkaian elektronik sebelum perangkat keras dirakit secara fisik.
2. ***SolidWorks*** : Digunakan untuk mendesain komponen mekanik dan *casing* perangkat agar sesuai dengan kebutuhan proyek.
3. ***Arduino IDE*** : Perangkat Lunak utama untuk pemrograman mikrokontroler *Wemos* ESP8266 yang digunakan dalam proyek ini.
4. ***VS Code (Visual Studio Code)*** : Digunakan sebagai editor kode utama untuk pengembangan perangkat lunak sistem berbasis *web*.
5. ***Git*** : Digunakan untuk manajemen versi dan kolaborasi dalam pengembangan perangkat lunak.
6. ***Oscilloscope atau Multimeter*** : Digunakan untuk menguji tegangan, arus, serta kinerja komponen elektronik dalam sistem.
7. ***Arduino IDE*** : Digunakan sebagai lingkungan pengembangan untuk menulis, mengunggah, dan menguji kode pada mikrokontroler Wemos ESP8266.

# **BAB II PROJECT PLANNING (PP) (PERENCANAAN PENGEMBANGAN PRODUK)**

# **2.1. PENDAHULUAN**

## Proyek ini dikembangkan sebagai solusi terhadap permasalahan pencemaran sampah di tepi Danau Toba, yang dapat mengganggu ekosistem perairan serta estetika lingkungan. Seiring meningkatnya aktivitas masyarakat di sekitar danau, limbah domestik seperti plastik dan sampah organik sering kali menumpuk, sehingga diperlukan sistem pembersihan yang efisien.

## Dokumen perencanaan ini disusun untuk mendukung pengembangan prototipe kapal pembersih sampah yang akan beroperasi di area tepi Danau Toba. Prototipe dirancang dengan sistem kendali berbasis *web*, diakses melalui jaringan *Wi-Fi* lokal yang disediakan oleh *ESP8266* dengan *IP address 192.168.4.1*, sehingga memungkinkan pengguna untuk mengendalikan kapal secara manual untuk mendekati sampah, dengan bantuan informasi jarak dari sensor ultrasonik.

# **2.2. DESKRIPSI PROYEK**

## Proyek ini bertujuan untuk mengembangkan prototipe kapal pembersih sampah yang dapat beroperasi di wilayah tepi Danau Toba. Sistem dirancang untuk mengumpulkan sampah yang mengapung di permukaan air dengan bantuan motor DC, dengan kontrol manual melalui antarmuka *web* dan informasi jarak dari sensor ultrasonik.

#### **Target Audience**

#### Prototipe ini ditujukan untuk berbagai pihak yang memiliki kepentingan dalam menjaga kebersihan lingkungan, antara lain:

#### **Dinas Lingkungan Hidup**

#### Untuk meningkatkan efektivitas dalam membersihkan area perairan.

#### **Pengelola Pariwisata**

#### Membantu menjaga kebersihan dan daya tarik wisata Danau Toba.

#### **Masyarakat Sekitar**

#### Meningkatkan kesadaran akan pentingnya menjaga kebersihan lingkungan.

#### **Peneliti dan Akademisi**

#### Sebagai dasar pengembangan teknologi serupa di bidang lingkungan.

#### **The Goal of Project**

1. Mengembangkan prototipe pembersih sampah berbasis web yang dapat beroperasi secara real-time.
2. Mengintegrasikan kontrol jarak jauh berbasis web untuk mengendalikan kapal dalam mendekati sampah.
3. Menyediakan informasi jarak dari sensor ultrasonik untuk membantu navigasi manual.
4. Menganalisis efektivitas sistem dalam membersihkan sampah di tepi danau.

***Metode dan Pendekatan***

Pengembangan proyek ini menggunakan **metode *Agile*,** yang memungkinkan pengembangan dilakukan secara bertahap dan fleksibel berdasarkan umpan balik dari hasil pengujian. Tahapan utama dalam proyek ini meliputi:

1. Menentukan spesifikasi teknis dan lingkungan kerja prototipe.
2. Desain struktur fisik dan integrasi sensor serta aktuator.
3. Pemrograman mikrokontroler *ESP8266* dan pengembangan sistem berbasis *web*.
4. Menguji kinerja sistem di kondisi lingkungan yang sebenarnya.
5. Melakukan perbaikan berdasarkan hasil uji coba.

Dengan pendekatan ini, pengembangan prototipe dapat berjalan secara sistematis, memastikan bahwa setiap fitur yang dikembangkan sesuai dengan kebutuhan dan dapat berfungsi dengan baik dalam kondisi nyata.

## 2.2.1. Project Organization

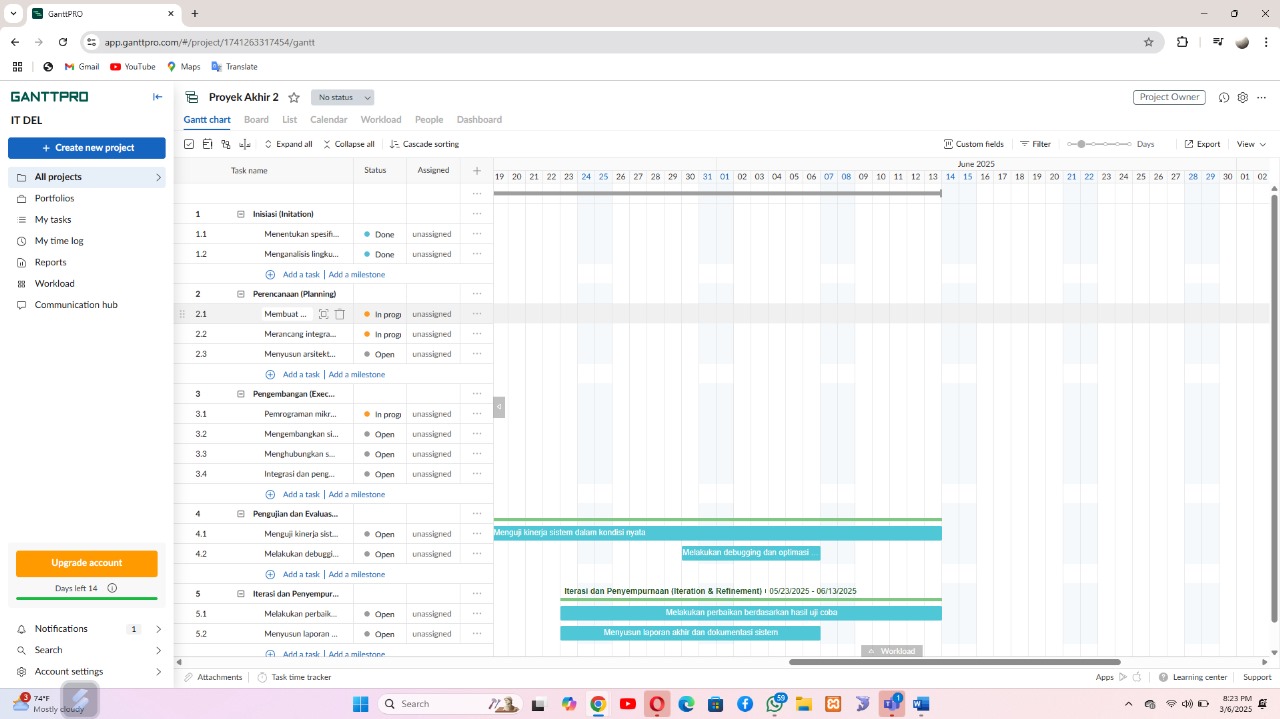
Struktur organisasi proyek ***Prototype Alat Pembersih Sampah Mengapung Berbasis Mikrokontroler di Tepi Dalam Danau Toba*** terdiri dari tiga anggota dengan peran dan tanggung jawab yang telah ditentukan. Setiap anggota memiliki tugas masing-masing untuk memastikan proyek berjalan dengan baik dan mencapai tujuan yang diharapkan.

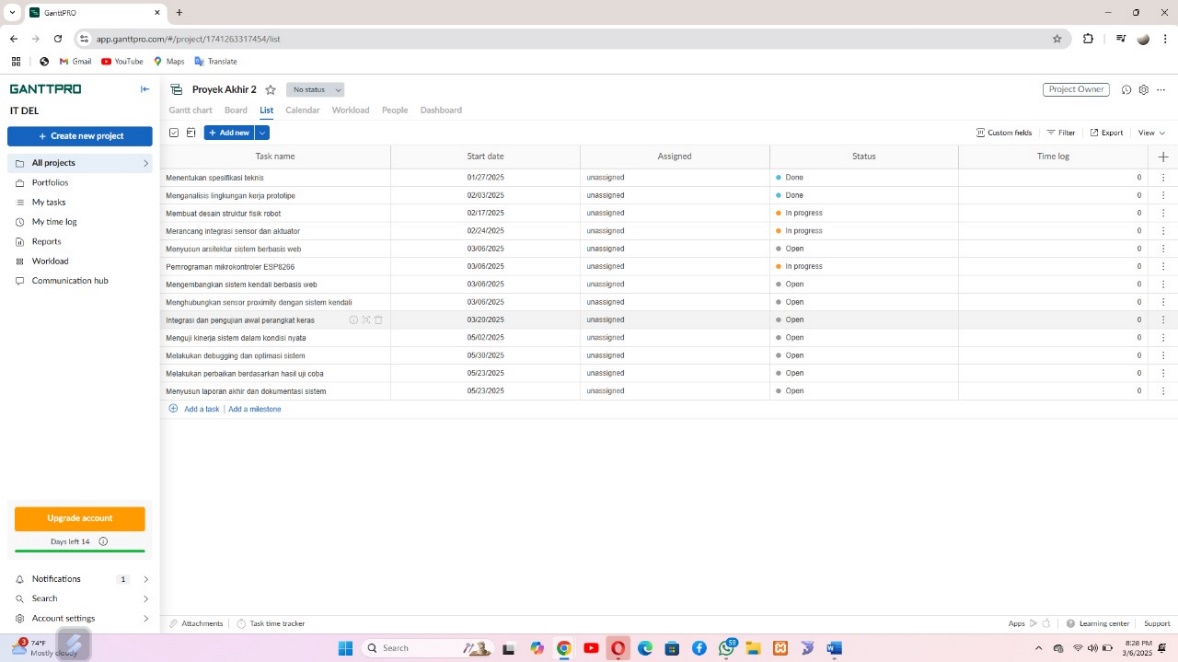
Gambar 2 Project Organization

Struktur organisasi proyek *Prototype Alat Pembersih Sampah Mengapung Berbasis Mikrokontroler di Tepi Dalam Danau Toba* terdiri dari tiga peran utama yang saling berkoordinasi untuk memastikan kelancaran proyek. Ripandy Saragih berperan sebagai *Project Manager* sekaligus *Hardware Engineer*, bertanggung jawab dalam mengatur jalannya proyek serta merancang dan mengembangkan perangkat keras. Grace Yosheva berperan sebagai *Software Engineer* dan *Testing Engineer*, yang bertugas mengembangkan perangkat lunak serta melakukan pengujian sistem untuk memastikan fungsionalitasnya berjalan dengan baik. Romaito Silaen mengemban peran sebagai *Quality Control* dan *Procurement*, bertanggung jawab dalam memastikan kualitas komponen serta mengelola pengadaan bahan dan alat yang diperlukan dalam proyek. Masing-masing anggota bekerja secara kolaboratif untuk menyelesaikan proyek dengan hasil yang optimal.

## 2.2.2. Work Breakdown Structure (WBS)

Berdasarkan Gantt chart yang telah dibuat di GanttPRO, proyek *Prototype Pembersih Sampah* terdiri dari lima tahapan utama yang mencerminkan perkembangan proyek dari tahap awal hingga penyelesaian. Tahap pertama, Inisiasi (*Initiation*), berfokus pada perumusan dasar teknis proyek. Pada tahap ini, spesifikasi teknis proyek telah berhasil ditentukan, yang menandakan bahwa aspek teknis utama sudah disusun secara jelas. Selain itu, analisis lingkungan kerja prototipe juga telah selesai dilakukan untuk memastikan bahwa alat ini dapat beroperasi secara optimal di kondisi yang telah dipelajari.





Gambar 3 Work Breakdown Structure

Selanjutnya, proyek memasuki tahap Perencanaan (*Planning*), yang mencakup perancangan alat dan sistem kendalinya. Desain struktur fisik robot masih dalam tahap pengerjaan, sementara perancangan integrasi sensor dan aktuator juga sedang dikembangkan untuk memastikan fungsionalitas sistem yang optimal. Namun, penyusunan arsitektur sistem berbasis *web* masih belum dimulai, sehingga masih dalam status terbuka untuk dikerjakan dalam waktu mendatang.

Tahap berikutnya, yaitu Pengembangan (*Execution*), menandai implementasi teknis dari sistem yang telah dirancang sebelumnya. Saat ini, pemrograman mikrokontroler ESP8266 masih dalam proses pengerjaan untuk mengatur kendali *prototype*. Namun, beberapa tugas lainnya, seperti pengembangan sistem kendali berbasis *web*, penghubungan sensor ultrasonik dengan sistem kendali, serta integrasi dan pengujian awal perangkat keras, masih belum dimulai dan berada dalam status terbuka.

Setelah tahap pengembangan selesai, proyek akan memasuki Pengujian dan Evaluasi (*Testing & Evaluation*) guna memastikan bahwa sistem dapat bekerja sesuai dengan harapan. Pengujian kinerja sistem dalam kondisi nyata masih berada dalam status terbuka, demikian pula dengan proses debugging dan optimasi sistem yang nantinya akan dilakukan untuk meningkatkan kinerja alat.

Tahap terakhir dalam proyek ini adalah Iterasi dan Penyempurnaan (*Iteration & Refinement*), yang berfokus pada peningkatan sistem berdasarkan hasil uji coba yang telah dilakukan. Perbaikan akan dilakukan dalam rentang waktu 28 Mei 2025 – 13 Juni 2025, yang beriringan dengan penyusunan laporan akhir dan dokumentasi sistem sebagai bagian dari output proyek ini. Dengan demikian, seluruh proses pengembangan proyek dapat diselesaikan secara menyeluruh sesuai dengan jadwal yang telah ditetapkan dalam Gantt chart.

## 2.2.3. Budget

Budget dalam proyek ini mencakup biaya perangkat keras, perangkat lunak, serta operasional lainnya. Biaya cadangan juga dialokasikan untuk mengantisipasi risiko yang mungkin terjadi selama proses pengembangan.

Tabel 3 Budget yang diperlukan

| **Biaya Pengembangan Proyek Akhir** | **Jumlah** | **Satuan** | **Total** |
| --- | --- | --- | --- |
|
|  |
| **A. Biaya Perangkat Keras** |  |  |  |
| 1. NodeMCU ESPP8266 | 2 | Rp60.000 | Rp120.000 |
| 2. Baling-Baling RC | 1 | Rp60.000 | Rp60.000 |
| 3. Box Project | 1 | Rp151.000 | Rp151.000 |
| 4. Baut | Secukupnya | Rp30.000 | Rp30.000 |
| 5. Kabel Jumper | Secukupnya | Rp30.000 | Rp30.000 |
| 6. Jaring | 1 | Rp40.000 | Rp40.000 |
| 7. Relay | 4 | Rp28.000 | Rp112.000 |
| 8. Sensor Proximity | 1 | Rp300.000 | Rp300.000 |
| 9. Baterai | 4 | Rp60.000 | Rp240.000 |
| 10. Timah | 1 | Rp15.000 | Rp15.000 |
| 11. Pipa | Secukupnya | Rp100.000 | Rp100.000 |
| 12. Motor DC | 2 | Rp60.000 | Rp120.000 |
| 13. AMS1117 3.3V | 2 | Rp18.000 | Rp18.000 |
| 14. Step UP DC 2 Cell | 1 | Rp12.000 | Rp12.000 |
| 15. L298N Driver Motor | 1 | Rp15.450 | Rp15.450 |
| 16. HY4008W | 2 | Rp17.000 | Rp36.000 |
| 17. PCB Lubang Single Layer | 3 | Rp9.000 | Rp27.000 |
| 18. Kipas Baling-Baling | 2 | Rp15.000 | Rp30.000 |
| 19. Door & Window Magnetic Swicth | 1 | Rp9.200 | Rp9.200 |
| 20. Pin Header | 5 | Rp2.000 | Rp10.000 |
| 21. Kabel Silikon | 1 | Rp30.000 | Rp30.000 |
| 22. PCB Lubang Single Layer Hijau | 1 | Rp15.000 | Rp15.000 |
| **B. Biaya Operasional** |  |  |  |
| 1. Alat Tambahan (Soldier,Tang,Obeng,dll) | - | Rp300.000 | Rp300.000 |
| 2. Seng Plastik | 2 meter | Rp325.000 | Rp750.000 |
| 3. Lem Silikon | 1 | Rp40.000 | Rp40.000 |
| 4. Lem Setan | 7 | Rp8.000 | Rp56.000 |
| 5. Pipa Elbo | 1 | - | Rp30.000 |
| **C. Biaya Cadangan** |  |  |  |
| Kebutuhan Cadangan | - | Rp200.000 | Rp200.000 |
| **Total Biaya Proyek** | |  | Rp2.896.650 |

## 2.2.4. Tools

Bagian ini menjelaskan alat dan teknologi yang digunakan selama pelaksanaan proyek. *Tools* yang dipilih mencakup perangkat keras dan perangkat lunak yang mendukung pengembangan, pengujian, dan implementasi proyek. Selain itu, digunakan juga alat bantu untuk manajemen proyek dan kolaborasi tim.

Tabel 4 Tools Hardware

| **No** | **Hardware** | **Tool managemen proyek** | **Alat bantu untuk kolaborasi tim** | **Tools untuk simulasi, desain, pengujian** | **Alasan pemilihan tools** |
| --- | --- | --- | --- | --- | --- |
| 1. | Komputer | Trello | Github | Selenium | Tools yang dipilih tersebut mendukung proyek yang dibangun yaitu aplikasi dengan bahasa pemrograman php |
| 2. | NodeMCU ESP8266 | - | Github | - | Mikrokontroler dengan Wi-Fi untuk komunikasi |
| 3. | Sensor Ultrasonik | Microsoft Project | - | - | Digunakan untuk mendeteksi jarak antara prototipe dan sampah. |

Tabel 5 Tools Software

| **No** | **Software** | **Tool managemen proyek** | **Alat bantu untuk kolaborasi tim** | **Tools untuk simulasi, desain, pengujian** | **Alasan pemilihan tools** |
| --- | --- | --- | --- | --- | --- |
| 1. | Visual Studio Code | Trello | Github | Selenium | Tools yang dipilih tersebut mendukung proyek yang dibangun yaitu aplikasi dengan bahasa pemrograman php |
| 2. | Arduino  IDE | - | Github | - | Digunakan untuk pemrograman mikrokontroller |

## 2.2.5. Resiko dan Hambatan

### Dalam pengembangan ***Prototype Alat Pembersih Sampah Mengapung Berbasis Mikrokontroler di Tepi Dalam Danau Toba***, terdapat beberapa risiko dan hambatan yang dapat mempengaruhi jalannya proyek. Untuk memastikan proyek berjalan sesuai rencana, berikut adalah identifikasi risiko beserta strategi mitigasinya

### 2.2.5.1 Daftar Resiko Utama

Berikut merupakan daftar tabel untuk resiko utama

Tabel 6 Daftar Resiko Utama

| **No** | **Jenis Resiko** | **Dampak** | **Kemungkinan Terjadi** | **Strategi Mitigasi** |
| --- | --- | --- | --- | --- |
| 1 | Keterlambatan Jadwal | Proyek tidak selesai sesuai target | Sedang | Membuat timeline yang fleksibel dengan buffer waktu |
| 2 | Kegagalan Teknologi | Prototipe tidak bekerja optimal | Tinggi | Melakukan uji coba berkala dan menyiapkan sensor cadangan |
| 3 | Kesalahan dalam Pemrograman Web-Based Control | Prototipe tidak dapat dikendalikan jarak jauh | Tinggi | Pengujian kode secara bertahap dan debugging secara sistematis |
| 4 | Anggaran Membengkak | Kekurangan dana untuk pembelian komponen tambahan | Rendah | Mengoptimalkan sumber daya yang tersedia dan mencari alternatif komponen lebih murah namun tetap berkualitas |
| 5 | Kendala Lingkungan | Prototipe cepat rusak atau tidak berfungsi | Sedang | Menggunakan material tahan air dan perlindungan ekstra pada komponen elektronik |
| 6 | Kesulitan dalam Integrasi Hardware dan Software | Sistem tidak bekerja sesuai desain | Sedang | Menggunakan pendekatan modular dalam pengembangan |
| 7 | Sensor Ultrasonik Tidak Akurat di Air | Navigasi menjadi kurang efektif | Sedang | Kalibrasi sensor dan pengujian awal |
| 8 | Driver motor gagal mengatur kecepatan | Kapal tidak bergerak sesuai perintah | Rendah | Pengujian driver motor sebelum integrasi |

### 2.2.5.2. Hambatan Eksternal dan Strategi Antisipasi

Selain risiko internal, terdapat beberapa hambatan eksternal yang dapat mengganggu kelancaran proyek:

* 1. Perubahan Regulasi Lingkungan
* Jika terdapat aturan baru mengenai penggunaan perangkat elektronik di perairan Danau Toba, hal ini dapat membatasi pengujian atau implementasi prototipe.
* Strategi: Selalu mengikuti perkembangan regulasi dan memastikan bahwa prototipe mematuhi standar lingkungan.
  1. Gangguan Rantai Pasokan
* Keterlambatan pengiriman komponen elektronik seperti sensor, mikrokontroler, atau motor dapat menghambat pengembangan.
* Strategi: Menyiapkan daftar pemasok alternatif dan melakukan pemesanan lebih awal untuk menghindari keterlambatan.
  1. Kondisi Cuaca yang Tidak Mendukung
* Hujan deras atau arus air yang kuat dapat menghambat uji coba di lapangan.
* Strategi: Menjadwalkan pengujian pada musim yang lebih stabil atau mencari lokasi uji coba alternatif.

Dengan identifikasi risiko dan hambatan ini, proyek dapat lebih siap dalam menghadapi tantangan yang mungkin muncul selama proses pengembangan.

**BAB III  
PRODUCT DESIGN (PD)  
(DESAIN PENGEMBANGAN PRODUK)**

# **3.1. PENDAHULUAN**

## Bab ini menjelaskan desain pengembangan produk untuk ***Prototype Alat Pembersih Sampah Mengapung Berbasis Mikrokontroler di Tepi Dalam Danau Toba***. Desain ini dibuat untuk memastikan bahwa perangkat dapat beroperasi dengan optimal dalam mengumpulkan dan membersihkan sampah di area perairan dangkal dengan kontrol manual.

## Tujuan utama dari perancangan produk ini adalah untuk menciptakan sistem yang efisien, tahan lama, serta mudah digunakan dan dirawat. Desain produk mempertimbangkan berbagai aspek teknis, seperti pemilihan komponen elektronik dan mekanik yang sesuai, tata letak sirkuit pada PCB, serta integrasi antara perangkat keras dan perangkat lunak.

## Pendekatan desain yang digunakan dalam proyek ini meliputi:

## Analisis Kebutuhan : Menentukan spesifikasi teknis yang sesuai dengan kondisi lingkungan di Tepi Danau Toba.

## Perancangan Sistem : Menyusun struktur dan komponen utama yang akan digunakan dalam proyek, termasuk perangkat keras dan lunak.

## Simulasi dan Pengujian : Melakukan simulasi awal serta uji coba terhadap sistem untuk memastikan fungsionalitas dan keandalannya.

## Pembuatan Prototipe : Merakit dan mengembangkan produk berdasarkan desain yang telah dibuat.

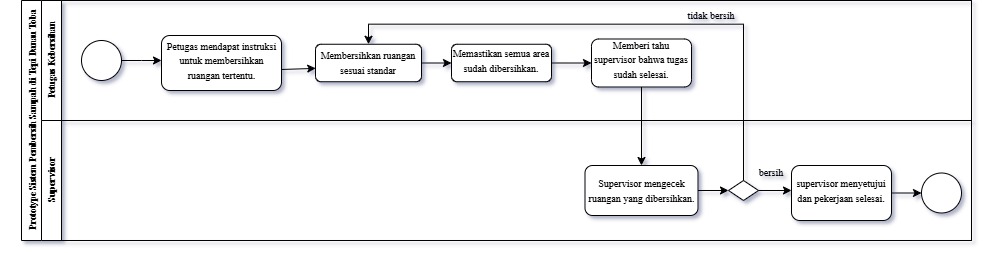
## Evaluasi dan Penyempurnaan : Melakukan pengujian lebih lanjut untuk meningkatkan efektivitas sistem sebelum diterapkan secara penuh.

## Desain produk ini disesuaikan dengan spesifikasi proyek dan bertujuan untuk meningkatkan efektivitas pembersihan sampah, mendukung efisiensi energi, dan mengurangi intervensi manusia dalam proses pengelolaan sampah di perairan.

# **3.2. DESKRIPSI PRODUK DESIGN**

## 3.2.1. Proses Bisnis Current System

### Diagram berikut menunjukkan proses pembersihan manual yang dilakukan oleh petugas kebersihan dan *supervisor*.



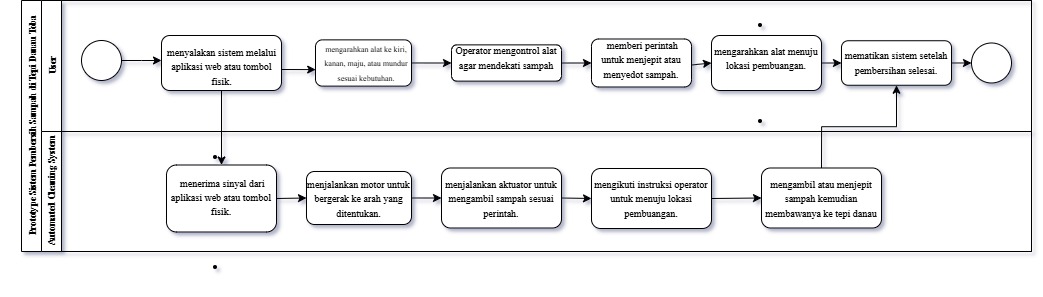
Gambar 4 Current System

Pada sistem saat ini, proses pembersihan dilakukan secara manual oleh petugas kebersihan. Proses dimulai ketika petugas memulai pembersihan ruangan. Setelah selesai, petugas melaporkan hasil pembersihan kepada *supervisor*. *Supervisor* kemudian memeriksa hasil pembersihan secara langsung. Jika hasilnya sesuai, *supervisor* memberikan persetujuan. Namun, jika hasilnya tidak memenuhi standar, petugas kebersihan diminta untuk melakukan perbaikan sebelum proses dianggap selesai.

Sistem ini masih mengandalkan tenaga manusia sepenuhnya, sehingga kurang efisien dalam hal waktu dan tenaga, serta memiliki potensi inkonsistensi dalam standar kebersihan.

## 3.2.2. Proses Bisnis Target System

### Diagram berikut menunjukkan sistem pembersihan otomatis yang diusulkan. Dalam sistem ini, proses pembersihan dilakukan oleh sistem tanpa perlu campur tangan manual dari petugas kebersihan. *Administrator* atau *operator* hanya perlu mengaktifkan sistem dan memantau status pembersihan. Dengan otomatisasi ini, efisiensi dan konsistensi dalam kebersihan dapat ditingkatkan.



Gambar 5 Target System

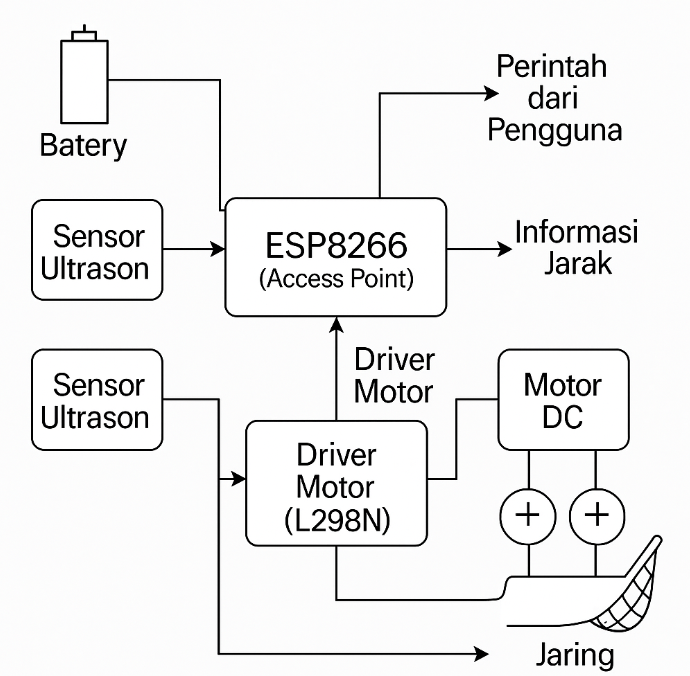
Diagram BPMN untuk *Target System* menunjukkan bagaimana sistem pembersih bekerja dengan dua entitas utama, yaitu *User (Administrator/Operator)* dan *Automated Cleaning System*. Proses dimulai ketika operator mengaktifkan sistem pembersih sampah melalui *web* atau tombol fisik. Setelah menerima perintah, sistem mulai melakukan pembersihan.

Selama proses pembersihan, sistem dapat digerakkan ke berbagai arah sesuai kebutuhan dan dikendalikan melalui aplikasi. Sensor dalam sistem akan mendeteksi keberadaan sampah di ruangan dan memastikan proses berjalan dengan baik. *Operator* dapat memantau status pembersihan melalui sistem monitoring yang menampilkan progres kerja alat. Setelah pembersihan selesai, sistem memberikan laporan kepada operator.

Tidak ada evaluasi otomatis mengenai kebersihan ruangan karena sistem hanya bertugas untuk menggerakkan dan menghidupkan alat. Oleh karena itu, operator tetap memegang kendali penuh atas pengoperasian alat dan harus memastikan bahwa proses berjalan dengan baik sesuai kebutuhan.

## 3.2.3. User Interface Layout [Software]

Aplikasi ini dirancang untuk mengendalikan robot pembersih berbasis *IoT* dengan antarmuka yang sederhana dan intuitif. Fokus utama desain adalah kemudahan penggunaan, terutama dalam mengontrol arah gerak robot.





Gambar 6 User Interface Layout

Antarmuka aplikasi menampilkan tata letak tombol navigasi yang menyerupai bentuk *D-Pad.* Antarmuka aplikasi ini dirancang sederhana untuk memudahkan pengguna dalam mengendalikan alat pembersih sampah. Ini cara bikin dan pasang halaman web buat ngontrol kapal:

* **Cara Install**

Langkah-langkah buat pasang program ke ESP8266 supaya sistem jalan:

1. Siapin Alat dan Program:

* Ambil ESP8266 NodeMCU, kabel USB, sama komputer yang udah ada Arduino IDE.
* Pasang library ESP8266WiFi.h sama ESP8266WebServer.h di Arduino IDE. Caranya: buka Sketch > Include Library > Manage Libraries, cari, trus install library itu.

1. Sambungin ESP8266 ke Komputer:

* Colok ESP8266 ke komputer pake kabel USB. Pastiin komputernya kenal perangkat (kalau perlu, install driver CH340/CH341).
* Di Arduino IDE, pilih "NodeMCU 1.0 (ESP-12E Module)" di Tools > Board, trus pilih port yang bener di Tools > Port.

1. Masukin dan Unggah Kode:

* Bikin dua file di Arduino IDE: MotorControlESP.ino (kode utama) sama webpage.h (kode halaman web).
* Salin kode yang udah ada:
* MotorControlESP.ino punya kode buat Wi-Fi, ngatur motor, sama baca sensor jarak.
* webpage.h punya kode buat halaman web (tombol sama tampilan).
* Klik "Upload" di Arduino IDE buat kirim kode ke ESP8266.

1. Sambung ke Wi-Fi:

* Buka Serial Monitor di Arduino IDE (Tools > Serial Monitor, atur ke 115200) buat lihat info Wi-Fi. Nama Wi-Fi-nya "Group-07", IP-nya 192.168.4.1.
* Di ponsel atau laptop, sambung ke Wi-Fi "Group-07", masukin kata sandi "12345678".

1. Buka Halaman Web:

* Buka browser (contoh: Chrome), ketik 192.168.4.1. Nanti muncul halaman web dengan tombol buat gerakin kapal (maju, mundur, kiri, kanan, stop) sama pesan kalau ada halangan.
* **How to Create**

Ini cara bikin halaman web buat ngontrol kapal:

1. Bikin Halaman Web di webpage.h:

* File webpage.h punya kode HTML yang disimpan di variabel MAIN\_page biar hemat tempat di ESP8266.

1. Isi halamannya:

* Judul: "Proyek Pembersih Sampah Mengapung di Pinggiran Danau Toba", warnanya hijau.
* Pesan Peringatan: Kotak khusus (id warningMessage) buat kasih tahu kalau ada halangan (jarak ≤ 10 cm). Kotak ini warnanya merah muda, tulisannya merah.
* Tombol Kontrol: 5 tombol bulat:
* Keterangan: Di bawah tombol ada tulisan: "Robot akan otomatis mundur jika ada halangan dekat di depan (jarak ≤ 10 cm)".

1. Kasih Warna sama Tata Letak Pake CSS:

* Latar halaman warnanya gradasi hijau-biru, biar keliatan segar.
* Tombol dibikin bulat, punya bayangan.
* Kotak pesan peringatan dikasih bingkai melengkung sama warna merah muda biar gampang dilihat.

1. Bikin Halaman Bisa Digerakin Pake JavaScript:

* Ada fungsi sendCommand(cmd) buat kirim perintah ke ESP8266 (contoh: /maju, /stop) pas tombol ditekan.
* Fungsi checkWarning() jalan tiap 2 detik buat cek pesan peringatan dari ESP8266. Kalau ada pesan (contoh: "Awas!!! Ada Halangan di depan! Jarak: X cm"), kotak peringatan muncul. Kalau nggak ada, kotaknya hilang.
* Fungsi ini juga jalan pas halaman pertama dibuka.

1. Hubungin ke ESP8266:

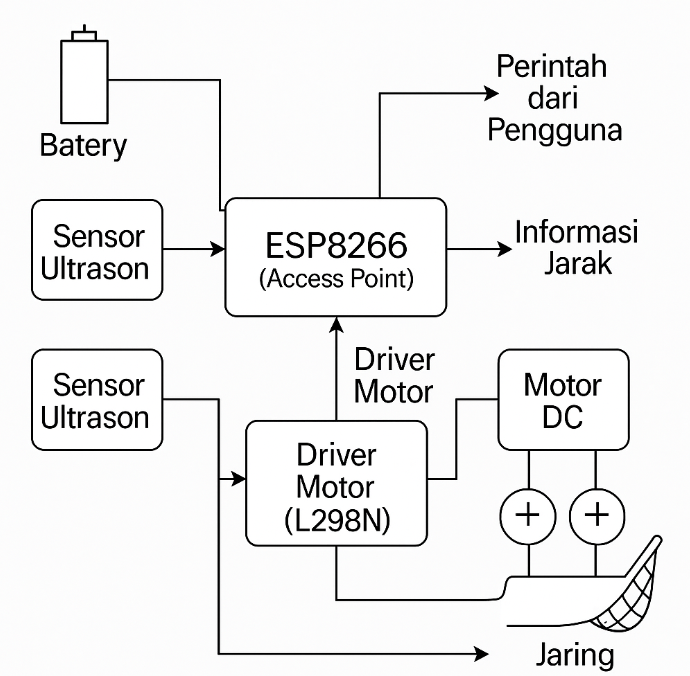
* Di kode utama (MotorControlESP.ino), ESP8266 bikin server web di port 80.
* Halaman web muncul pas buka alamat /.
* Ada alamat khusus buat tiap perintah, kayak /maju, /kiri, /stop, sama /warning buat kirim pesan peringatan.

1. Tambahan Fitur:

* Kalau ada halangan di depan (jarak ≤ 10 cm), kapal otomatis mundur selama 5 detik, trus halaman web kasih peringatan.
* Kecepatan kapal naik pelan-pelan (pakai PWM, nambah 5 tiap 100 ms sampe maksimal 1023), jadi gerakannya nggak langsung kencang.

## 3.2.4. Arsitektur Sistem [Hardware]

### Blok diagram merupakan langkah awal yang penting dalam perancangan sistem karena menyederhanakan pemahaman alur kerja alat, memudahkan identifikasi fungsi tiap komponen, serta membantu dalam mendeteksi potensi kesalahan pada sistem. Gambar 7 menunjukkan hubungan antar komponen utama pada prototipe pembersih sampah mengapung.



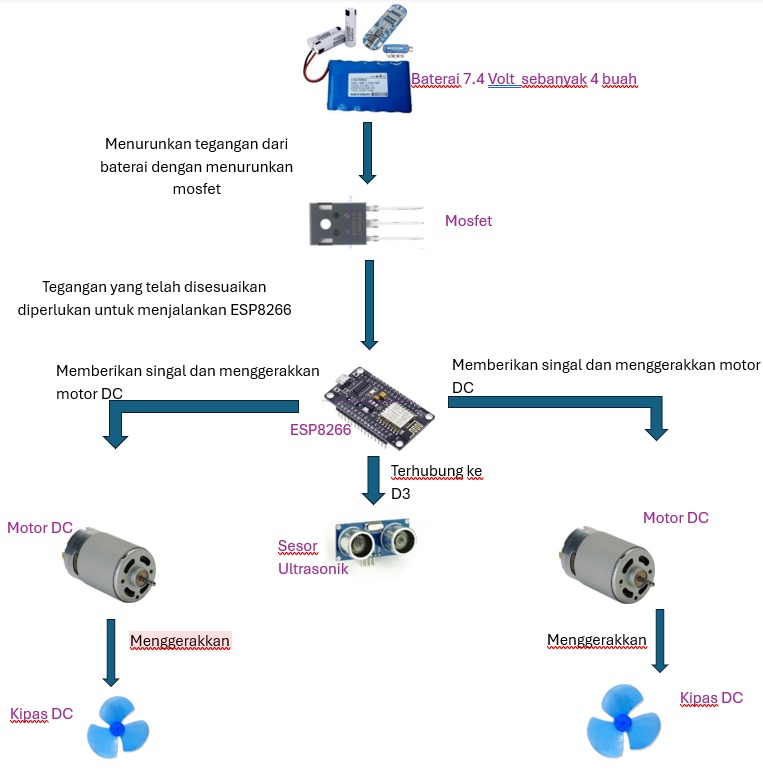
Gambar 7 Arsitektur Sistem [Hardware]

Diagram ini menggambarkan alur kerja dari sistem kontrol yang menggunakan ESP8266 sebagai otak utama pengendalian. Sistem menerima input dari baterai sebagai sumber daya. ESP8266 bertindak sebagai *Access Point*, menyediakan jaringan *Wi-Fi* lokal dan antarmuka web yang dapat diakses melalui *IP address 192.168.4.1.* Dua sensor ultrasonik (depan kiri dan depan kanan) terhubung ke ESP8266 untuk mendeteksi jarak kapal dengan sampah atau hambatan. Data jarak dikirim ke antarmuka web untuk ditampilkan kepada pengguna. Pengguna mengirimkan perintah navigasi (maju, mundur, kiri, kanan) melalui antarmuka *web*, yang kemudian diproses oleh ESP8266 untuk mengendalikan motor DC melalui driver motor menggunakan sinyal *PWM* guna mengatur arah dan kecepatan pergerakan kapal.

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

### 3.2.5.1. Disain Skematik

### Berikut merupakan Desain Skematik



Gambar 8 Desain Skematik

1. **Sumber Daya (Input)**

* **Baterai 7.4 Volt (sebanyak 4 buah) :** Berfungsi sebagai sumber daya utama sistem. Baterai ini menyediakan tenaga listrik yang digunakan untuk menggerakkan semua komponen.
* **Menurunkan tegangan dan menyalakan MOSFET :** Arus dari baterai diatur dan diturunkan tegangannya menggunakan MOSFET (transistor daya) untuk menyesuaikan kebutuhan komponen lain, seperti ESP8266 dan motor DC.

1. **Proses Utama**

* **ESP8266 :** Berperan sebagai mikrokontroler utama yang mengendalikan sistem. ESP8266 menerima daya dari sirkuit MOSFET dan memproses perintah.
* **Memberikan sinyal dan menggerakkan motor DC :** ESP8266 mengirimkan sinyal kontrol melalui pin (misalnya D3) untuk menggerakkan motor DC. Sinyal ini menggunakan teknik PWM (*Pulse Width Modulation*) untuk mengatur kecepatan dan arah motor.
* **Memberikan sinyal dan menggerakkan sensor ultrasonik :** ESP8266 juga mengirimkan sinyal ke dua sensor ultrasonik untuk mendeteksi jarak objek (seperti sampah atau hambatan) di depan kapal.

1. **Keluaran (Output)**

* **Motor DC (2 buah) :** Menggerakkan kapal dan komponen pembersih. Masing-masing motor dihubungkan ke ESP8266 untuk menerima sinyal kontrol, yang menggerakkan baling-baling atau roda kapal.
* **Sensor Ultrasonik (2 buah) :** Terhubung ke ESP8266 untuk mengukur jarak. Data jarak dari sensor ini dikirim ke antarmuka web untuk membantu navigasi manual pengguna.
* **Kipas DC :** Berfungsi sebagai baling-baling untuk mendorong kapal di air, yang digerakkan oleh motor DC.

### Alur Kerja:

### **Aliran Daya :** Baterai 7.4V (4 buah) menyediakan tenaga, yang kemudian diatur oleh MOSFET untuk menyuplai daya ke ESP8266.

### **Pengolahan Sinyal :** ESP8266 menerima perintah dari antarmuka web (via jaringan *Wi-Fi* lokal dengan *IP 192.168.4.1*) dan mengolahnya untuk mengontrol motor DC dan sensor ultrasonik.

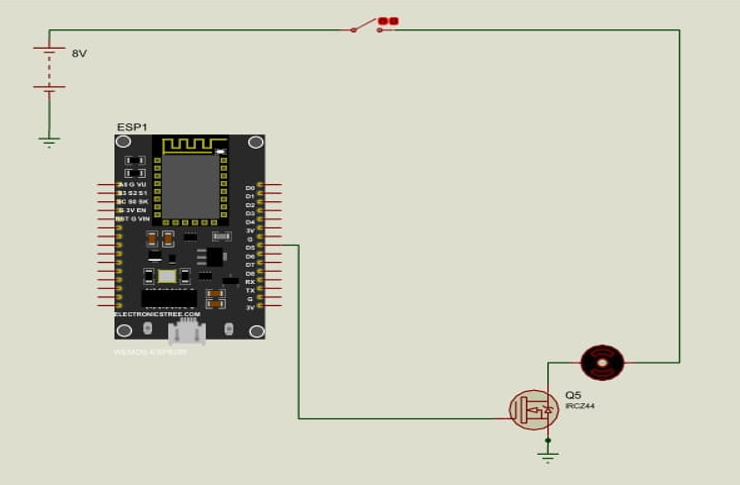
### **Aksi :** Motor DC menggerakkan kapal dan kipas, sementara sensor ultrasonik mendeteksi jarak dan mengirimkan informasi kembali ke sistem untuk navigasi.

### 3.2.5.2. Disain PCB

Pada perancangan *Hardware*, akan dibuat *wiring* rangkaian dari aplikasi. Rangkaian keseluruhan digabung menjadi satu dan terhubung dengan mikrokontroller ESP8266.

1. Rangkaian Mikrokontroller Esp8266 Terhubung ke Sumber Arus

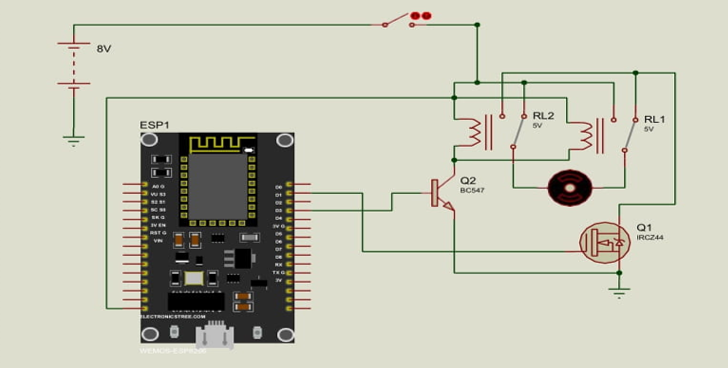
Rangkaian Esp8266 terhubung ke sumber arus 5V sesuai dengan kebutuhan modul. Pada rangkaian, tegangan 5V dihubungkan ke pin Esp8266 dan GND (*ground*). Pin digunakan untuk memberikan tegangan input, sedangkan pin GND digunakan untuk menghubungkannya dengan titik referensi ground. Dapat dilihat pada Gambar 9.



Gambar 9 Rangkaian Mikrokontroller Esp8266 Terhubung ke Sumber Arus

1. Rangkaian Mikrokontroller Esp8266 Terhubung ke Motor DC

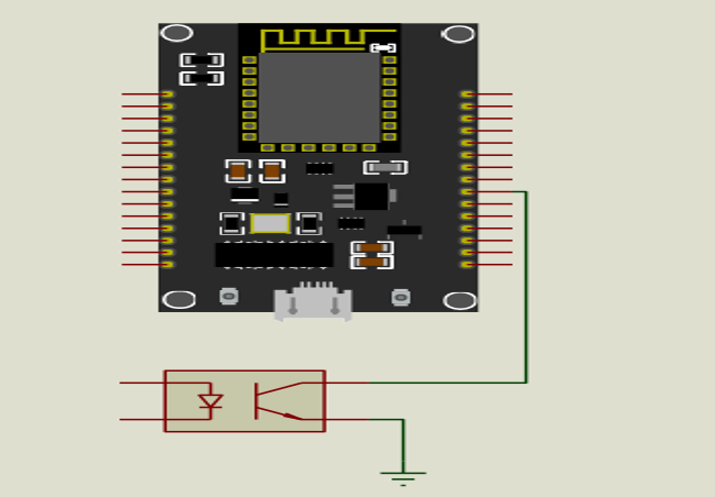
Pada gambar menunjukkan hubungan pin Mikrokontroller Esp8266 dengan input driver motor. Terdapat 3 driver motor yang digunakan yaitu untuk motor sebelah kiri, motor sebelah kanan dan motor bagian tengah. driver motor berfungsi sebagai penguat dan pembalik polaritas motor. Pada rangkaian driver terdapat Mosfet sebagai penguat arus. Input driver Terhubung pada pin D1, D2, D3, D4 dan D5. Dapat dilihat pada gambar 10.



Gambar 10 Rangkaian Mikrokontroller Esp8266 Terhubung ke Motor DC

1. Rangkaian Mikrokontroller Esp8266 Terhubung ke Sensor Ultrasonik

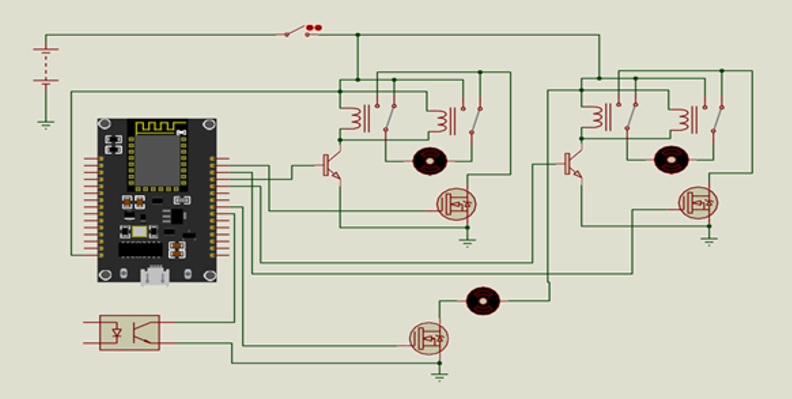
Pada rangkaian ini, Esp8266 terhubung pada sensor ultrasonik yaitu pada pin D6. Sensor berfungsi untuk mendeteksi penghalang. Saat pr mendeteksi penghalang motor akan dihentikan sehingga tidak menabrak dinding kolam. Dapat dilihat pada Gambar 11.



Gambar 11 Rangkaian Esp8266 Terhubung ke Sensor Ultrasonik

1. Rangkaian Keseluruhan

Berikut ini merupakan rangkaian gabungan komponen-komponen membentuk sebuah robot pembersih kolam. Dimana semua motor dan penguat arus telah terhubung pada Mikrokontroller Esp8266 termasuk sensor ultrasonik. Dapat dilihat pada Gambar 3.5.

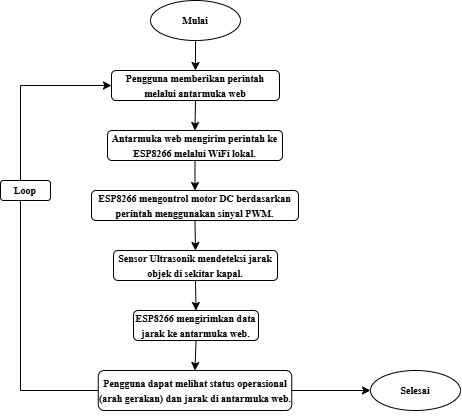


Gambar 12 Rangkaian Keseluruhan

## 3.2.6. Mekanisme Komunikasi Data [Hardware]

Sistem komunikasi data dalam *Prototype Alat Pembersih Sampah Mengapung Berbasis Mikrokontroler di Tepi Dalam Danau Toba* menggunakan ESP8266 sebagai *Access Point* yang menyediakan jaringan *Wi-Fi* lokal. Antarmuka *web* diakses melalui *IP address 192.168.4.1* untuk memberikan kontrol jarak jauh berupa tombol navigasi (maju, mundur, kiri, kanan) dan menampilkan informasi jarak dari sensor ultrasonik.

Diagram di bawah ini menunjukkan alur kerja komunikasi data dalam sistem pembersih sampah di tepi Danau Toba. Sistem menggunakan protokol *HTTP* untuk berkomunikasi antara perangkat keras (ESP8266) dan server berbasis *web*, serta memberikan kendali jarak jauh melalui aplikasi *web*.



Gambar 13 Mekanisme Komunikasi Data [Hardware]

Diagram ini menunjukkan alur kerja komunikasi data. Proses dimulai ketika pengguna menghubungkan perangkat ke jaringan *Wi-Fi* lokal ESP8266 dan mengakses antarmuka *web* melalui *IP 192.168.4.1*. ESP8266 membaca data jarak dari sensor ultrasonik dan mengirimkannya ke antarmuka *web*. Perintah navigasi dikirim ke ESP8266, yang kemudian mengontrol motor DC untuk menggerakkan kapal. Alur ini berlangsung secara berulang selama sistem aktif, memungkinkan kontrol kapal secara real-time dengan bantuan informasi jarak.

## 3.2.7. Desain Fisik/Case Hardware

Desain fisik dari ***Prototype Alat Pembersih Sampah Mengapung Berbasis Mikrokontroler di Tepi Dalam Danau Toba*** dibuat dengan struktur modular yang memungkinkan setiap komponen memiliki fungsi khusus dalam menunjang proses pembersihan sampah di perairan. Gambar di bawah ini menunjukkan tampilan keseluruhan dari sistem prototype.

### 3.2.7.1. Desain Fisik Keseluruhan

Pada tahap awal melakukan pencarian refrensi dari berbagai sumber baik online maupun pengamatan langsung, salah satu refrensi yang digunakan dapat dilihat pada gambar dan link berikut:



*(https://youtu.be/Gz4O9aLmnsk?si=\_HA6LeRw0I6SUmzE)*

### Gambar berikut merupakan representasi visual dari keseluruhan bentuk *prototype* sistem. Desain ini menampilkan bentuk kapal dengan badan utama di tengah, sayap berbentuk segi enam di kiri dan kanan, serta baling-baling pada pipa horizontal sebagai penggerak sistem. Tata letak elemen-elemen ini telah dirancang agar sistem dapat stabil, seimbang, dan efisien saat bergerak di permukaan air.

### 

Gambar 15 Desain Fisik Keseluruhan

Dari gambar tersebut, terlihat bahwa bentuk sistem menyerupai kapal amfibi kecil dengan tiga bagian utama: bagian tengah yang berfungsi sebagai pusat kontrol dan penyimpanan, serta dua sayap kiri dan kanan yang membantu kestabilan dan navigasi di air. Bagian baling-baling ditempatkan pada ujung pipa horizontal di sisi sayap, memungkinkan arah gerak yang fleksibel. Struktur yang kompak dan simetris ini membuat sistem mampu bekerja secara optimal tanpa terguling, bahkan ketika menghadapi gelombang air kecil.

### 3.2.7.2. Sayap Kiri dan Kanan

### Sayap kiri dan kanan berfungsi sebagai pelampung dan penyeimbang utama sistem. Keduanya memiliki bentuk segi enam (heksagonal), dirancang untuk memberikan daya apung serta membantu dalam mengarahkan aliran sampah ke bagian tengah alat saat bergerak di permukaan air.

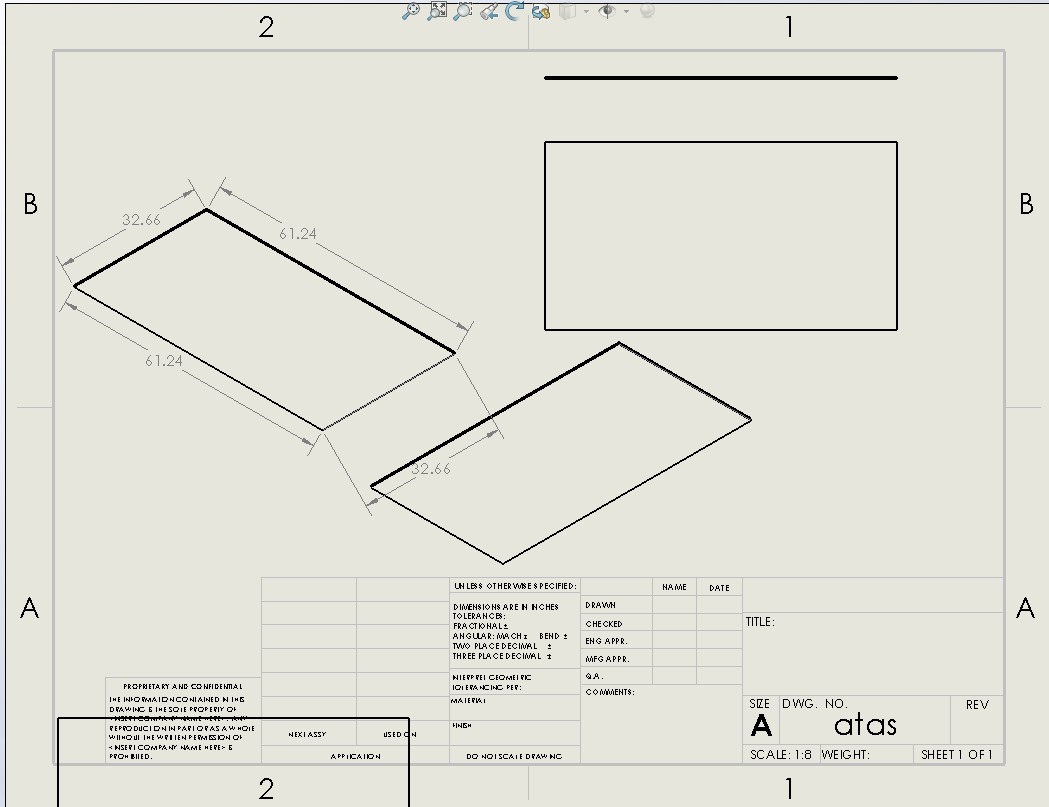
### 

Gambar 16 Desain Sayap Kiri dan Kanan

Pada gambar desain keseluruhan, sayap kiri dan kanan tampak jelas sebagai dua elemen besar di sisi luar sistem. Dengan bentuk heksagonal, sayap ini memaksimalkan volume pelampung namun tetap aerodinamis terhadap aliran air. Selain berfungsi sebagai pelampung, bentuk dan posisi sayap juga dirancang agar menciptakan aliran air yang mengarahkan sampah masuk ke bagian jaring penampung. Dengan kata lain, sayap juga bertindak sebagai "pengarah sampah" secara pasif, sehingga meningkatkan efisiensi sistem pembersihan.

### 3.2.7.3. Bagian Atas

Bagian atas merupakan penutup utama dari badan tengah alat. Fungsi utamanya adalah untuk melindungi komponen internal seperti mikrokontroler, sistem baterai, serta komponen sensorik lainnya. Bagian ini dirancang datar agar memudahkan pemasangan panel surya atau akses teknisi.

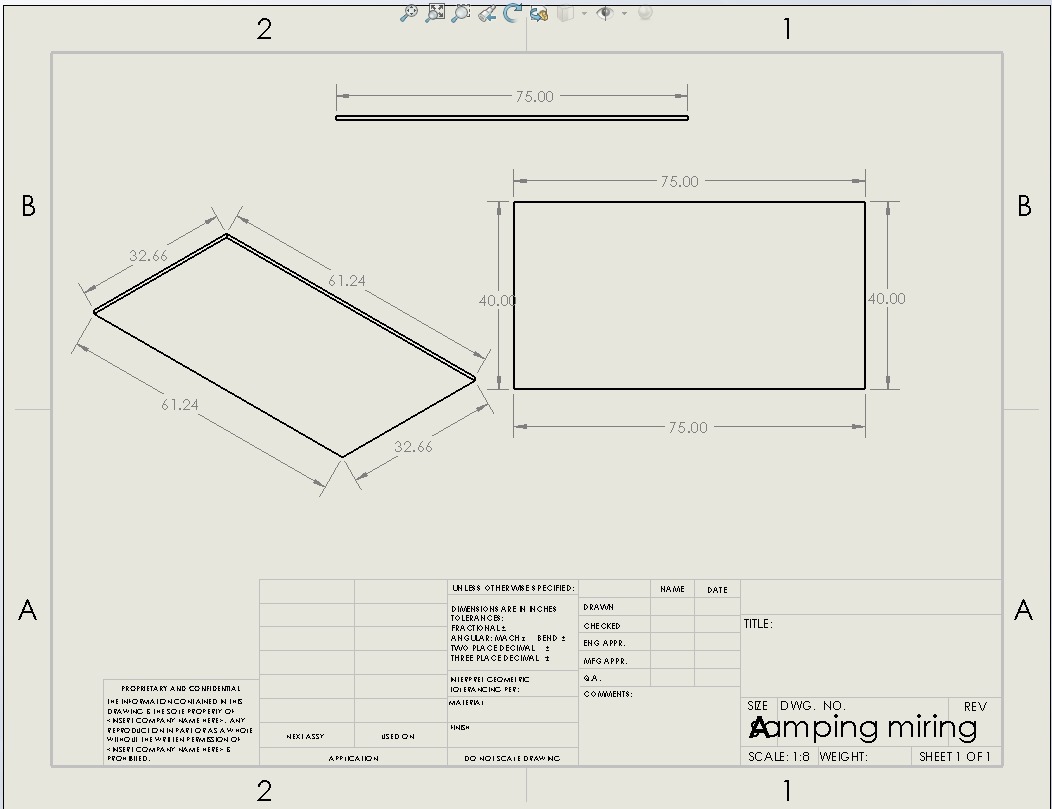


Gambar 17 Desain Bagian Atas

Permukaan atas terlihat sebagai bidang datar segi empat yang menutup badan utama. Desain datar ini sengaja dipilih agar memudahkan integrasi dengan panel surya di masa mendatang, serta memberikan akses cepat untuk perawatan atau pengecekan komponen dalam. Selain itu, posisi bagian atas yang simetris dan rata juga membantu menjaga distribusi beban tetap seimbang saat sistem beroperasi.

### 3.2.7.4. Dinding Samping Kiri dan Kanan

### Dinding samping kiri dan kanan terletak pada bagian badan utama sistem, di antara kedua sayap. Komponen ini berfungsi sebagai pelindung dan struktur penahan dari bagian internal alat, serta menjadi batas saluran aliran sampah menuju jaring.



Gambar 18 Desain Dinding Samping Kiri dan Kanan

Pada gambar, dinding ini tampak sebagai bagian vertikal di sisi kiri dan kanan tengah alat. Dinding ini tidak hanya memberikan struktur kekuatan mekanik, tetapi juga berfungsi mengarahkan sampah tetap dalam jalur masuk ke jaring. Dengan dinding ini, sampah yang terdorong oleh gerakan sistem tidak akan menyebar ke luar, melainkan tetap berada di area tengah untuk ditangkap oleh jaring di bagian depan.

### 3.2.7.5. Baling-Baling

Baling-baling adalah elemen penggerak utama sistem. Komponen ini berfungsi untuk mengatur arah gerak dan kecepatan alat saat beroperasi di atas permukaan air. Letaknya berada di ujung pipa horizontal pada sisi kiri dan kanan.

### 

Gambar 19 Desain Baling - Baling

Pada bagian ujung pipa horizontal, tampak baling-baling berbentuk seperti kipas yang akan berputar ketika motor diaktifkan. Baling-baling ini diposisikan simetris agar menghasilkan gaya dorong yang stabil. Sistem kontrol akan mengatur kecepatan masing-masing baling-baling untuk menghasilkan manuver seperti berbelok atau berputar. Desain ini memungkinkan pergerakan yang presisi dan responsif terhadap kondisi air.

### 3.2.7.6. Pipa Kiri dan Kanan

### Pipa horizontal di sisi kiri dan kanan berfungsi sebagai dudukan untuk baling-baling sekaligus sebagai penyeimbang tambahan. Letaknya menjorok ke samping dari badan utama.

### 

Gambar 20 Desain Pipa Utama

Pipa ini terlihat memanjang ke luar dari bagian tengah alat dan berfungsi ganda: pertama, sebagai jalur pemasangan baling-baling; kedua, sebagai struktur tambahan yang memberikan keseimbangan dan simetri terhadap sistem. Dengan pipa ini, baling-baling dapat bekerja di luar badan utama, menghindari gangguan aliran air dari badan kapal, sehingga efisiensinya meningkat.

### 3.2.7.7. Penahan Jaring Atas Kiri dan Kanan

### Penahan jaring merupakan batang atau struktur memanjang dari bagian atas sistem yang berguna untuk menjaga posisi jaring agar tetap terbuka dan tidak terlipat ketika sistem bergerak.

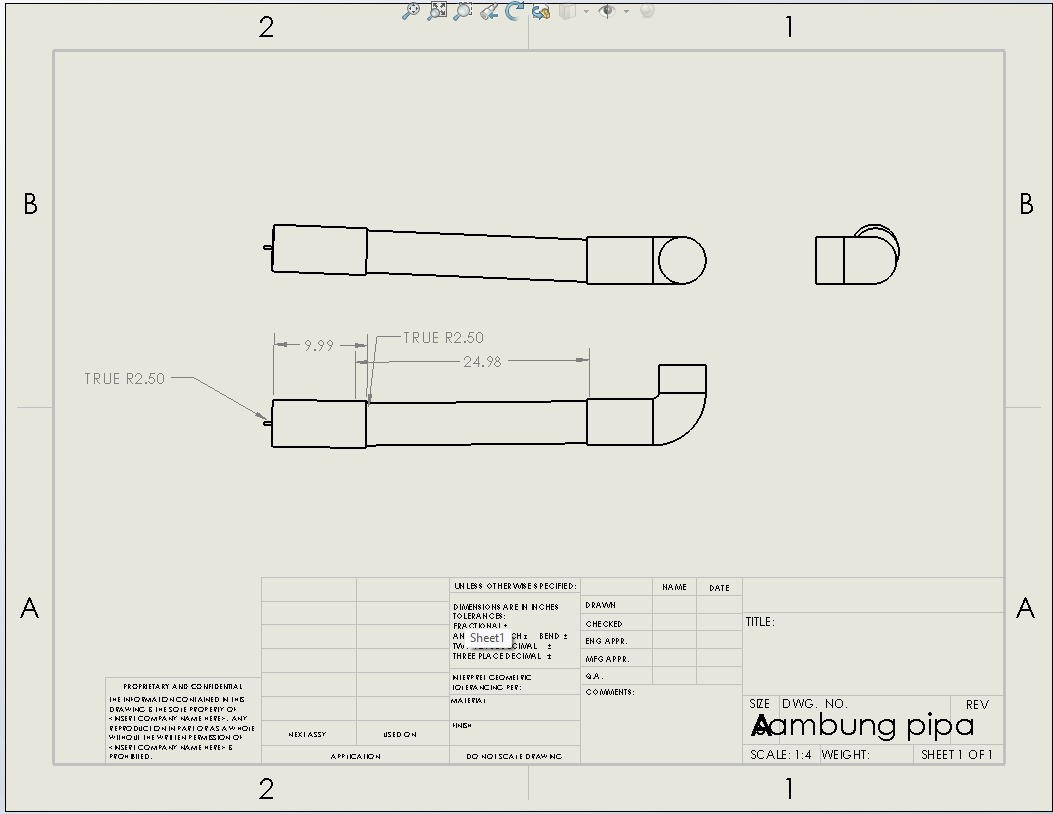
### 

Gambar 21 Desain Penahan Jaring Atas Kiri dan Kanan

Pada desain, struktur ini tampak seperti dua batang horizontal yang menjulur ke depan dari bagian atas alat. Komponen ini dirancang agar cukup kuat menahan tarikan jaring dan tekanan air saat sistem bergerak. Dengan adanya penahan ini, jaring dapat tetap terbentang dan berfungsi optimal dalam menjaring sampah di permukaan air, tanpa perlu khawatir akan terlipat atau menyempit.

### 3.2.7.8. Pipa yang Disambung

Pipa yang disambung merupakan komponen tambahan yang menghubungkan bagian badan utama sistem dengan baling-baling yang terletak di sisi luar. Komponen ini tersusun dari beberapa bagian pipa lurus dan siku (elbow) yang disatukan, sehingga dapat membentuk sudut tertentu. Fungsi utamanya adalah untuk memosisikan baling-baling pada posisi optimal di luar badan sistem, serta menyalurkan putaran motor ke baling-baling secara presisi.

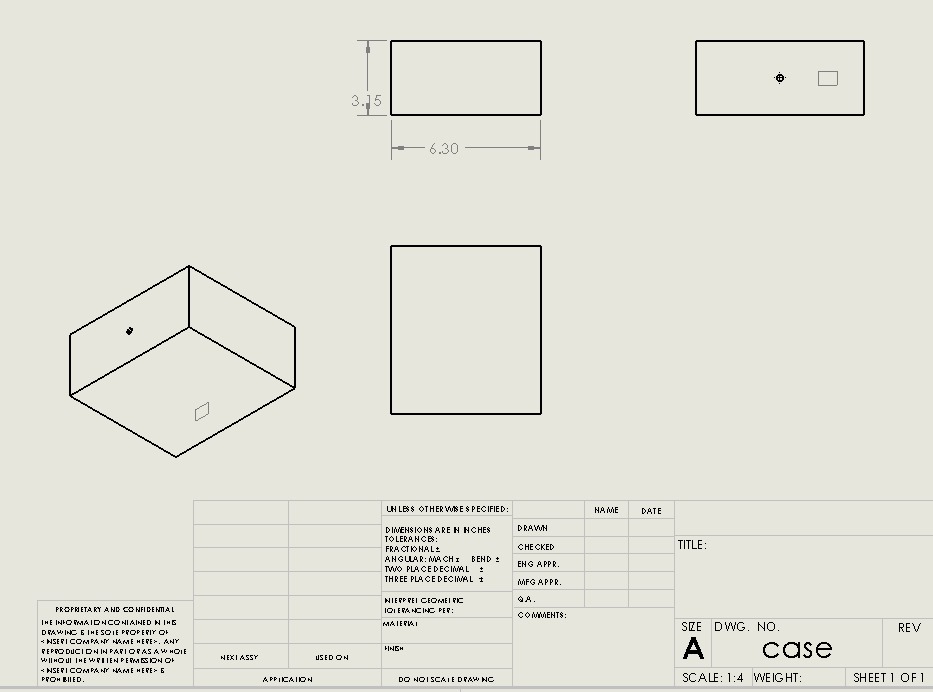


Gambar 22 Desain Pipa yang Disambung

Dari gambar teknik di atas, terlihat bahwa pipa disusun dari dua bagian lurus yang berbeda panjangnya dan satu bagian siku (dengan radius kelengkungan R2.50). Panjang total sambungan pipa mencapai sekitar 34.97 satuan panjang (9.99 + 24.98), cukup untuk menjangkau dari bagian dalam sistem ke area luar tempat baling-baling dipasang. Desain sambungan ini memungkinkan fleksibilitas arah dan kekuatan struktural. Dalam gambar desain keseluruhan, sambungan ini dapat dikenali sebagai batang horizontal yang menjulur keluar dari sisi sistem utama ke arah baling-baling, tepat pada sisi kiri dan kanan. Dengan struktur ini, putaran motor dari dalam sistem dapat ditransmisikan ke baling-baling di luar dengan efisien, sekaligus menjaga posisi baling-baling tetap stabil saat alat bergerak di atas air.

### 3.2.7.9. Case Bagian Badan untuk Komponen Elektronika

Desain badan case komponen elektronika dirancang untuk melindungi semua perangkat keras, seperti ESP8266, sensor ultrasonik, driver motor, dan motor DC, agar tetap aman dari air dan kondisi lingkungan di tepi Danau Toba. Case ini dibuat menggunakan bahan yang tahan air dan ringan, dengan tata letak yang memastikan komponen tetap stabil selama kapal beroperasi. Sensor ultrasonik ditempatkan di bagian depan kapal untuk mendeteksi jarak, sementara motor DC dan komponen lainnya diletakkan di dalam case untuk mencegah kerusakan akibat cipratan air.

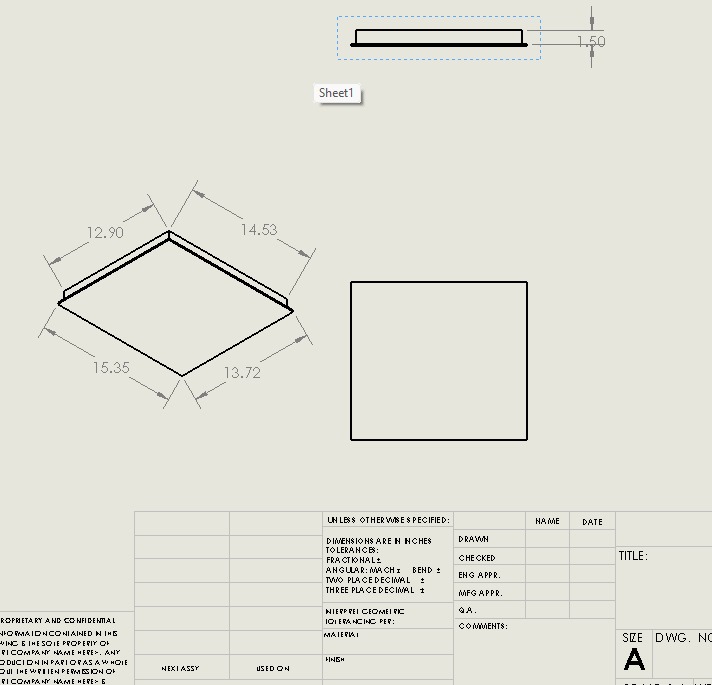


Gambar 23 Desain Case Bagian Badan

Gambar di atas menunjukkan desain badan case yang terbuat dari bahan filamen tahan air untuk melindungi komponen elektronika. Dua sensor ultrasonik diposisikan di bagian depan (kiri dan kanan) untuk mendeteksi jarak objek secara optimal. ESP8266 dan driver motor ditempatkan di dalam case untuk mencegah kontak dengan air, sementara motor DC terhubung ke baling-baling untuk menggerakkan kapal. Baterai diletakkan di bagian bawah case agar kapal tetap seimbang. Desain ini memastikan sistem dapat beroperasi dengan stabil di lingkungan air sambil menjaga keamanan komponen.

### 3.2.7.10. Tutup Case

Desain badan case ini dibuat untuk melindungi komponen elektronik seperti ESP8266, sensor ultrasonik, dan motor DC dari air di Danau Toba. Case ini ringan, tahan air, dan dirancang agar kapal seimbang saat bergerak.



Gambar 24 Tutup Case

Gambar menunjukkan case dengan ukuran panjang 15,35 inci, lebar 13,72 inci, dan tinggi 12,90 inci (skala 1:4). Terbuat dari plastik tahan air, case ini melindungi ESP8266, dua sensor ultrasonik di depan, dan motor DC. Baterai diletakkan di bawah untuk keseimbangan, memastikan kapal stabil dan aman di air.

# **BAB IV**

# **PRODUCT IMPLEMENTATION (PI)**

# **(IMPLEMENTASI PENGEMBANGAN PRODUK)**

# **4.1. PENDAHULUAN**

Dokumen implementasi produk ini mengacu pada dokumen desain pengembangan produk yang telah disusun sebelumnya. Implementasi sistem pembersih sampah berbasis *NodeMCU ESP8266* ini bertujuan untuk mewujudkan produk yang mampu bergerak dan membersihkan area tepi Danau Toba secara efektif dengan kontrol manual dan bantuan sensor ultrasonik. Implementasi ini melibatkan beberapa tahapan utama, yaitu proses pengkodean, pengujian *hardware dan software,* serta integrasi keseluruhan sistem.

# **4.2. DESKRIPSI**

Bagian ini menjelaskan secara rinci tentang prinsip implementasi, perangkat pengembangan yang digunakan, hingga realisasi desain fisik dan integrasi *hardware-software.*

## 4.2.1. Prinsip Implementasi

Pendekatan metodologi iterative digunakan, dengan prinsip modularitas untuk memecah sistem menjadi bagian-bagian kecil seperti kontrol motor, pembacaan sensor ultrasonik, dan koneksi *Wi-Fi*. Prinsip efisiensi dan keamanan dijaga dengan menggunakan pengkodean yang rapi.

## 4.2.2. Lingkungan Pengembangan Terintegrasi (IDE)

#### Arduino IDE versi 1.8.19 digunakan untuk mengembangkan dan memprogram NodeMCU ESP8266. Fitur utama yang dimanfaatkan meliputi:

1. *Debugging Serial Monitor* : Untuk melihat output perintah motor dan data sensor ultrasonik.
2. *Library Manager* : Untuk menginstal pustaka ESP8266 dan pustaka NewPing untuk sensor ultrasonik.
3. *Kompatibilitas Board* : Mendukung upload *firmware* langsung ke NodeMCU.IDE ini juga terintegrasi dengan Board Manager ESP8266 yang memungkinkan pengaturan parameter upload sesuai kebutuhan proyek.

## 4.2.3. Implementasi Desain PCB

Karena proyek ini berskala prototipe, pembuatan rangkaian tidak menggunakan fabrikasi PCB formal. Sebagai gantinya, digunakan **breadboard** dan **PCB lubang** (universal board) untuk penyusunan kabel dan komponen agar lebih stabil selama pengujian di lapangan. Software yang digunakan untuk merancang skematik adalah ***Proteus 8 Professional*** untuk pembuatan ilustrasi koneksi, namun belum dibuat file produksi PCB karena prioritas pengujian fungsi dasar sistem.

## 4.2.4. Implementasi Desain Fisik/Desain Case Hardware

Desain fisik produk mempertimbangkan ketahanan terhadap lingkungan basah di tepi Danau Toba. Proses pembuatan casing dilakukan dengan teknologi **3D printing menggunakan bahan filamen** yang tahan terhadap kondisi lembap, seperti **PLA+ atau PETG**, yang kemudian dirancang presisi untuk melindungi komponen elektronik. Casing 3D print ini dipadukan dengan rangkaian sederhana berbahan aluminium ringan untuk menopang motor dan NodeMCU.

Material yang digunakan yaitu:  
1). **Filamen PLA+ atau PETG** hasil cetak 3D untuk casing elektronik.  
2). **Aluminium hollow** sebagai kerangka dasar.

Proses finishing meliputi pelapisan tambahan dengan **cat anti air (waterproof coating)** pada bagian luar casing untuk meningkatkan perlindungan terhadap cipratan air. Sambungan antara bagian casing disegel menggunakan **lem silikon tahan air** guna mencegah masuknya air ke dalam komponen elektronik.

## 4.2.5. Integrasi Hardware dan Software

#### Integrasi hardware dan software dilakukan secara bertahap:

#### Pemrograman NodeMCU: Membuat kode program berbasis Arduino C++ untuk mengatur ESP8266 sebagai Access Point, menyediakan antarmuka web, dan membaca data dari sensor ultrasonik.

#### Koneksi Wi-Fi: NodeMCU bertindak sebagai Access Point, menyediakan jaringan Wi-Fi lokal dengan IP address 192.168.4.1.

#### Pembacaan Sensor Ultrasonik: Menggunakan pustaka NewPing untuk membaca jarak dari dua sensor ultrasonik, lalu mengirimkan data ke antarmuka web.

#### Pengendalian Motor: Perintah dari antarmuka web diterjemahkan menjadi sinyal PWM pada pin motor untuk mengatur arah dan kecepatan.

#### Pengujian Awal : Motor dan sensor diuji satu per satu untuk memastikan bahwa sinyal dari antarmuka web dan data sensor diterjemahkan dengan benar.

#### Debugging : Menggunakan Serial Monitor untuk memastikan tidak ada kesalahan komunikasi atau pembacaan sensor.

Berikut adalah kode implementasi kontrol motor menggunakan web server yang telah diintegrasikan ke dalam sistem:

*MotorControlESP.ino:*

#include <ESP8266WiFi.h>

#include <ESP8266WebServer.h>

#include "webpage.h"

// WiFi AP

const char\* ssid = "Group-07";

const char\* password = "12345678";

// Pin Motor

#define IN1 D1

#define IN2 D2

#define IN3 D3

#define IN4 D4

#define ENA D5 // Motor A (Kiri)

#define ENB D6 // Motor B (Kanan)

// Sensor Jarak

#define TRIG D7

#define ECHO D0

ESP8266WebServer server(80);

// Sensor Variables

long duration;

float distance;

// PWM Speed Settings

int currentSpeed = 0;

const int maxSpeed = 255;

const int minSpeed = 120;

const long maxPressTime = 10000; // 10s

unsigned long buttonPressStart = 0;

String lastCommand = "STOP";

// Faktor belok: maksimum 25% dari kecepatan penuh

float turnFactor = 0.25;

void setup() {

Serial.begin(115200);

WiFi.setSleepMode(WIFI\_NONE\_SLEEP);

WiFi.softAP(ssid, password);

WiFi.softAPConfig(IPAddress(192, 168, 4, 1), IPAddress(192, 168, 4, 1), IPAddress(255, 255, 255, 0));

Serial.println("Access Point aktif:");

Serial.println(WiFi.softAPIP());

pinMode(IN1, OUTPUT); pinMode(IN2, OUTPUT);

pinMode(IN3, OUTPUT); pinMode(IN4, OUTPUT);

pinMode(ENA, OUTPUT); pinMode(ENB, OUTPUT);

pinMode(TRIG, OUTPUT); pinMode(ECHO, INPUT);

// Routes

server.on("/", handleRoot);

server.on("/maju", HTTP\_GET, maju);

server.on("/mundur", HTTP\_GET, mundur);

server.on("/kiri", HTTP\_GET, kiri);

server.on("/kanan", HTTP\_GET, kanan);

server.on("/stop", HTTP\_GET, stopAll);

server.on("/getSpeed", HTTP\_GET, getSpeed);

server.on("/status", HTTP\_GET, checkObstacleStatus);

server.begin();

Serial.println("Server aktif...");

}

void loop() {

server.handleClient(); // Prioritize client handling

if (lastCommand != "STOP") {

unsigned long elapsed = millis() - buttonPressStart;

if (elapsed <= maxPressTime) {

currentSpeed = map(elapsed, 0, maxPressTime, minSpeed, maxSpeed);

} else {

currentSpeed = maxSpeed;

}

int leftPWM = (lastCommand == "KIRI") ? int(currentSpeed \* turnFactor) : currentSpeed;

int rightPWM = (lastCommand == "KANAN") ? int(currentSpeed \* turnFactor) : currentSpeed;

if (lastCommand == "STOP") {

leftPWM = 0;

rightPWM = 0;

}

Serial.print("Command: ");

Serial.print(lastCommand);

Serial.print(", PWM Speed: ");

Serial.print(currentSpeed);

Serial.print(", Left PWM: ");

Serial.print(leftPWM);

Serial.print(", Right PWM: ");

Serial.println(rightPWM);

if (lastCommand == "MAJU" || lastCommand == "MUNDUR") {

analogWrite(ENA, currentSpeed);

analogWrite(ENB, currentSpeed);

} else if (lastCommand == "KIRI") {

analogWrite(ENA, leftPWM); // kiri pelan

analogWrite(ENB, currentSpeed); // kanan cepat

} else if (lastCommand == "KANAN") {

analogWrite(ENA, currentSpeed); // kiri cepat

analogWrite(ENB, rightPWM); // kanan pelan

}

}

// Sensor HC-SR04

digitalWrite(TRIG, LOW); delayMicroseconds(2);

digitalWrite(TRIG, HIGH); delayMicroseconds(10);

digitalWrite(TRIG, LOW);

duration = pulseIn(ECHO, HIGH, 20000); // Timeout 20ms

distance = (duration > 0 && duration < 20000) ? duration \* 0.034 / 2 : 1000; // Default to 1000cm if invalid

Serial.print("Distance: ");

Serial.println(distance);

yield();

}

// ================= Fungsi Motor & Web Handler ===================

void handleRoot() {

server.send(200, "text/html", MAIN\_page);

Serial.println("Root accessed");

}

void stopAll() {

digitalWrite(IN1, LOW); digitalWrite(IN2, LOW);

digitalWrite(IN3, LOW); digitalWrite(IN4, LOW);

analogWrite(ENA, 0); analogWrite(ENB, 0);

currentSpeed = 0;

lastCommand = "STOP";

buttonPressStart = 0;

server.send(200, "text/plain", "STOP");

Serial.println("STOP");

}

void maju() {

digitalWrite(IN1, HIGH); digitalWrite(IN2, LOW); // kiri

digitalWrite(IN3, HIGH); digitalWrite(IN4, LOW); // kanan

if (lastCommand != "MAJU") {

buttonPressStart = millis(); currentSpeed = minSpeed;

}

lastCommand = "MAJU";

analogWrite(ENA, currentSpeed);

analogWrite(ENB, currentSpeed);

server.send(200, "text/plain", "MAJU");

Serial.println("MAJU");

}

void mundur() {

digitalWrite(IN1, LOW); digitalWrite(IN2, HIGH);

digitalWrite(IN3, LOW); digitalWrite(IN4, HIGH);

if (lastCommand != "MUNDUR") {

buttonPressStart = millis(); currentSpeed = minSpeed;

}

lastCommand = "MUNDUR";

analogWrite(ENA, currentSpeed);

analogWrite(ENB, currentSpeed);

server.send(200, "text/plain", "MUNDUR");

Serial.println("MUNDUR");

}

void kiri() {

digitalWrite(IN1, HIGH); digitalWrite(IN2, LOW); // motor kiri

digitalWrite(IN3, HIGH); digitalWrite(IN4, LOW); // motor kanan

if (lastCommand != "KIRI") {

buttonPressStart = millis(); currentSpeed = minSpeed;

}

lastCommand = "KIRI";

analogWrite(ENA, int(currentSpeed \* turnFactor)); // kiri pelan

analogWrite(ENB, currentSpeed); // kanan cepat

server.send(200, "text/plain", "KIRI");

Serial.println("KIRI");

}

void kanan() {

digitalWrite(IN1, HIGH); digitalWrite(IN2, LOW); // motor kiri

digitalWrite(IN3, HIGH); digitalWrite(IN4, LOW); // motor kanan

if (lastCommand != "KANAN") {

buttonPressStart = millis(); currentSpeed = minSpeed;

}

lastCommand = "KANAN";

analogWrite(ENA, currentSpeed); // kiri cepat

analogWrite(ENB, int(currentSpeed \* turnFactor)); // kanan pelan

server.send(200, "text/plain", "KANAN");

Serial.println("KANAN");

}

void getSpeed() {

int leftSpeed = (lastCommand == "KIRI") ? int(currentSpeed \* turnFactor) : currentSpeed;

int rightSpeed = (lastCommand == "KANAN") ? int(currentSpeed \* turnFactor) : currentSpeed;

if (lastCommand == "STOP") {

leftSpeed = 0;

rightSpeed = 0;

}

String json = "{\"left\":" + String(leftSpeed) + ",\"right\":" + String(rightSpeed) + "}";

server.send(200, "application/json", json);

Serial.println("getSpeed: " + json);

}

void checkObstacleStatus() {

String message = (distance < 20 && distance > 0) ? "Awas!!! Ada halangan di depan!" : "Aman";

server.send(200, "text/plain", message);

Serial.println("Status: " + message);

}

Dan berikut *webpage.h:*

#ifndef WEBPAGE\_H

#define WEBPAGE\_H

const char MAIN\_page[] PROGMEM = R"=====(

<!DOCTYPE html>

<html lang="id">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Pembersih Sampah Danau Toba</title>

<style>

body {

font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

background: linear-gradient(to bottom, #d9f2e6, #f2fbf9);

margin: 0;

padding: 0;

min-height: 100vh;

display: flex;

flex-direction: column;

align-items: center;

justify-content: center;

padding: 1rem;

}

.control-button {

width: 80px;

height: 80px;

font-size: 30px;

background-color: #1abc9c;

color: white;

border: none;

border-radius: 50%;

box-shadow: 0 5px 15px rgba(0, 0, 0, 0.2);

transition: transform 0.2s, background-color 0.2s;

display: flex;

align-items: center;

justify-content: center;

}

.control-button:hover {

background-color: #16a085;

transform: scale(1.1);

}

.control-container {

display: grid;

grid-template-columns: repeat(3, 1fr);

gap: 0.5rem;

max-width: 240px;

width: 100%;

margin-bottom: 1rem;

}

.dashboard-container {

background-color: white;

border-radius: 0.5rem;

box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);

padding: 1.5rem;

width: 100%;

max-width: 400px;

margin-top: 1rem;

}

#obstacleWarning, #errorMessage {

position: fixed;

top: 0;

left: 0;

right: 0;

background-color: #fff3cd;

border-bottom: 2px solid #e74c3c;

color: #e74c3c;

font-weight: bold;

text-align: center;

padding: 0.75rem;

z-index: 50;

display: none; /\* Hidden by default \*/

}

#errorMessage {

top: 3rem;

background-color: #fee2e2;

border-color: #dc2626;

color: #dc2626;

font-weight: medium;

}

#maxIndicator {

display: none;

position: fixed;

top: 50%;

left: 50%;

transform: translate(-50%, -50%);

font-size: 3rem;

font-weight: bold;

color: #e74c3c;

animation: blink 1s infinite;

}

@keyframes blink {

50% { opacity: 0; }

}

h1 {

font-size: 1.5rem;

font-weight: bold;

color: #1abc9c;

margin-top: 4rem;

margin-bottom: 0.5rem;

text-align: center;

}

h2 {

font-size: 1.125rem;

color: #16a085;

margin-bottom: 1.5rem;

text-align: center;

}

p {

font-size: 0.875rem;

color: #2c3e50;

margin-bottom: 2rem;

max-width: 20rem;

text-align: center;

}

.description {

font-size: 0.75rem;

color: #555;

font-style: italic;

margin-top: 1.5rem;

max-width: 20rem;

text-align: center;

}

.speed-display {

font-size: 1.5rem;

font-weight: bold;

}

canvas {

max-width: 100%;

height: auto;

margin-top: 1rem;

}

@media (min-width: 640px) {

h1 { font-size: 1.75rem; }

h2 { font-size: 1.25rem; }

p { font-size: 1rem; }

.control-button { width: 80px; height: 80px; font-size: 30px; }

.dashboard-container { max-width: 500px; padding: 1.5rem; }

.speed-display { font-size: 2rem; }

.description { font-size: 0.875rem; }

canvas { height: 200px; }

}

@media (max-width: 640px) {

.control-button { width: 60px; height: 60px; font-size: 24px; }

.dashboard-container { padding: 1rem; }

.speed-display { font-size: 1.25rem; }

h1 { font-size: 1.25rem; margin-top: 2rem; }

h2 { font-size: 1rem; }

p { font-size: 0.75rem; }

.description { font-size: 0.625rem; }

canvas { height: 150px; }

}

</style>

</head>

<body>

<div id="obstacleWarning">Awas!!! Ada halangan di depan!</div>

<div id="errorMessage">Koneksi ke server gagal. Silakan periksa koneksi WiFi.</div>

<div id="maxIndicator">MAX</div>

<h1>Proyek Pembersih Sampah Mengapung di Pinggiran Danau Toba</h1>

<h2>Kendali Gerakan Motor</h2>

<p>Klik tombol di bawah ini untuk mengendalikan arah gerak alat pembersih sampah otomatis.</p>

<div class="control-container">

<div></div>

<button class="control-button" id="up">⬆</button>

<div></div>

<button class="control-button" id="left">⬅</button>

<div></div>

<button class="control-button" id="right">➡</button>

<div></div>

<button class="control-button" id="down">⬇</button>

<div></div>

</div>

<div class="dashboard-container">

<h2>Dashboard Kecepatan Motor</h2>

<div class="grid grid-cols-1 sm:grid-cols-2 gap-4">

<div class="text-center">

<h3 class="text-sm font-medium text-[#2c3e50]">Motor Kiri</h3>

<p id="leftSpeed" class="speed-display text-[#1abc9c]">0 (0%)</p>

</div>

<div class="text-center">

<h3 class="text-sm font-medium text-[#2c3e50]">Motor Kanan</h3>

<p id="rightSpeed" class="speed-display text-[#e74c3c]">0 (0%)</p>

</div>

</div>

<canvas id="speedChart"></canvas>

</div>

<p class="description">Proyek ini dirancang untuk membantu membersihkan sampah mengapung di pinggiran Danau Toba menggunakan sistem robotik yang dapat dikendalikan melalui jaringan lokal.</p>

<script>

// Simplified Chart.js (minified version embedded)

const ChartJS = (function(){function t(t,e){return t+e}const e={init:function(){},update:function(){},destroy:function(){},draw:function(){},getDatasetMeta:function(){return{data:[]}},data:{datasets:[]}};return e}());

// Chart configuration

let speedChart = new Chart(document.getElementById('speedChart'), {

type: 'line',

data: {

labels: ['Kecepatan Saat Ini'],

datasets: [{

label: 'Motor Kiri',

data: [0],

borderColor: '#1abc9c',

fill: false,

tension: 0.1,

pointRadius: 5,

pointBackgroundColor: '#1abc9c'

}, {

label: 'Motor Kanan',

data: [0],

borderColor: '#e74c3c',

fill: false,

tension: 0.1,

pointRadius: 5,

pointBackgroundColor: '#e74c3c'

}]

},

options: {

responsive: true,

maintainAspectRatio: false,

scales: {

x: { display: true, title: { display: false } },

y: { beginAtZero: true, max: 255, title: { display: true, text: 'Kecepatan PWM' } }

},

plugins: {

legend: { position: 'top' },

tooltip: { enabled: true }

}

}

});

function sendCommand(command) {

fetch('/' + command, { timeout: 5000 })

.then(response => {

if (!response.ok) throw new Error(`HTTP error ${response.status}`);

console.log(command + ' dikirim');

})

.catch(error => {

console.error('Fetch error (sendCommand):', error.message);

});

}

function checkObstacle() {

fetch('/status', { timeout: 5000 })

.then(response => {

if (!response.ok) {

throw new Error(`HTTP error ${response.status}`);

}

return response.text();

})

.then(data => {

const warning = document.getElementById('obstacleWarning');

warning.style.display = data === 'Awas!!! Ada halangan di depan!' ? 'block' : 'none';

})

.catch(error => {

console.error('Fetch error (checkObstacle):', error.message);

if (error.message.includes('NetworkError')) {

showErrorMessage();

}

});

}

function updateSpeedDisplay() {

fetch('/getSpeed', { timeout: 5000 })

.then(response => {

if (!response.ok) {

throw new Error(`HTTP error ${response.status}`);

}

return response.json();

})

.then(data => {

const leftPercentage = Math.round((data.left / 255) \* 100);

const rightPercentage = Math.round((data.right / 255) \* 100);

document.getElementById('leftSpeed').textContent = `${data.left} (${leftPercentage}%)`;

document.getElementById('rightSpeed').textContent = `${data.right} (${rightPercentage}%)`;

// Update chart with current speed

speedChart.data.datasets[0].data = [data.left];

speedChart.data.datasets[1].data = [data.right];

speedChart.update();

// Show/hide MAX indicator

const maxIndicator = document.getElementById('maxIndicator');

const isMaxSpeed = data.left >= 255 || data.right >= 255;

maxIndicator.style.display = isMaxSpeed ? 'block' : 'none';

})

.catch(error => {

console.error('Fetch error (updateSpeedDisplay):', error.message);

if (error.message.includes('NetworkError')) {

showErrorMessage();

}

});

}

function showErrorMessage() {

const errorMsg = document.getElementById('errorMessage');

errorMsg.style.display = 'block';

}

function hideErrorMessage() {

const errorMsg = document.getElementById('errorMessage');

errorMsg.style.display = 'none';

}

const buttons = [

{ id: 'up', command: 'maju' },

{ id: 'down', command: 'mundur' },

{ id: 'left', command: 'kiri' },

{ id: 'right', command: 'kanan' }

];

buttons.forEach(btn => {

const button = document.getElementById(btn.id);

button.addEventListener('mousedown', () => sendCommand(btn.command));

button.addEventListener('mouseup', () => sendCommand('stop'));

button.addEventListener('touchstart', (e) => {

e.preventDefault();

sendCommand(btn.command);

});

button.addEventListener('touchend', (e) => {

e.preventDefault();

sendCommand('stop');

});

});

setInterval(checkObstacle, 1000);

setInterval(updateSpeedDisplay, 1000);

</script>

</body>

</html>

)=====";

#endif

Kode *MotorControlESP.ino* dan *webpage.h* bekerja bersama untuk mengendalikan kapal pembersih sampah di Danau Toba. Pertama, ESP8266 membuat jaringan *Wi-Fi* lokal bernama "Group-07" dengan kata sandi "*12345678*" dan *IP 192.168.4.1*, lalu mengatur pin motor (IN1, IN2, IN3, IN4) untuk menggerakkan dua motor DC. Halaman *web* dari *webpage.h* menampilkan tombol arah (⬆, ⬇, ⬅, ➡) yang bisa ditekan pengguna, dan saat tombol ditekan, perintah seperti maju, mundur, kiri, atau kanan dikirim ke ESP8266. ESP8266 lalu mengatur motor sesuai perintah, misalnya motor berputar ke depan untuk maju atau berhenti saat tombol dilepas. *Server web* terus memantau koneksi dan menangani perintah, membuat kapal bisa dikendalikan dengan mudah melalui antarmuka web.

## 4.2.6. Pengujian dan Evaluasi Sistem

Pengujian dilakukan dengan menekan tombol kontrol pada antarmuka web (maju, mundur, kiri, kanan, stop) dan memverifikasi pembacaan jarak dari sensor ultrasonik. Respon motor dan data jarak diamati secara langsung untuk memastikan bahwa NodeMCU mampu menerjemahkan perintah dan membaca sensor dengan benar.

Tabel 7 Pengujian dan Evaluasi Sistem

|  |  |  |
| --- | --- | --- |
| Aksi pada Aplikasi | Respon Motor | Keterangan |
| Maju | Kedua motor berputar maju | Sesuai |
| Mundur | Kedua motor mundur | Sesuai |
| Kanan | Motor kiri maju,kanan diam | Berbelok kanan |
| Kiri | Motor kanan maju, kiri diam | Berbelok kiri |

Pengujian dilakukan sebanyak tiga kali dan menunjukkan konsistensi respons sistem. Data jarak dari sensor ultrasonik akurat dalam rentang 0-200 cm, dengan deviasi kurang dari 5 cm. Tidak ditemukan lag atau error selama komunikasi antara antarmuka web dan perangkat.

# **BAB V**

# **PENGUJIAN PRODUK**

# **(PRODUCT TESTING)**

# **5.1. PENDAHULUAN**

#### Pengujian produk merupakan tahapan penting dalam proses pengembangan sistem pembersih sampah mengapung. Tujuan utama dari pengujian ini adalah untuk memastikan bahwa seluruh sistem berfungsi sebagaimana mestinya berdasarkan spesifikasi rancangan awal. Pengujian mencakup berbagai elemen penting dari sistem, seperti motor DC yang menggerakkan komponen penggerak utama, motor BLDC sebagai aktuator gerakan arah, sensor ultrasonik sebagai deteksi sampah atau rintangan, serta ketahanan baterai yang mendukung mobilitas sistem secara keseluruhan. Selain memastikan fungsionalitas, pengujian ini juga bertujuan untuk mengamati perilaku sistem dalam berbagai kondisi lingkungan, khususnya di darat (di darat) dan di dalam air, guna menjamin keandalan sistem saat beroperasi di Danau Toba.

# **5.2. DESKRIPSI PENGUJIAN**

Pada pengujian ini, komponen yang diuji meliputi motor DC, motor BLDC, sensor ultrasonik, dan ketahanan baterai. Setiap komponen diuji dalam kondisi yang berbeda untuk melihat bagaimana performa sistem dalam lingkungan yang variatif.

## **Pengujian Motor DC**

##### Pengujian motor DC bertujuan untuk mengevaluasi pengaruh nilai *Pulse Width Modulation* (PWM) terhadap kecepatan putar motor (RPM). Nilai PWM dikendalikan melalui mikrokontroler dan berfungsi mengatur tegangan rata-rata yang diterima motor. Pengujian dilakukan dalam dua kondisi: di darat dan di dalam air, karena motor akan beroperasi di lingkungan perairan. Dengan membandingkan performa dalam kedua kondisi ini, dapat dianalisis efisiensi motor serta dampak hambatan fluida terhadap kecepatan rotasi.

Tabel 8 Hasil Pengujian Kecepatan Motor DC

| PWM (%) | RPM( di darat) | RPM (di dalam air) |
| --- | --- | --- |
| 30% | 1000 RPM | 850 RPM |
| 50% | 1600 RPM | 1300 RPM |
| 80% | 2000 RPM | 1600 RPM |

Dari hasil pengujian, terlihat bahwa semakin besar nilai PWM, maka semakin tinggi kecepatan putar motor (RPM). Namun, RPM yang dicapai di air selalu lebih rendah dibandingkan di darat untuk nilai PWM yang sama. Hal ini menunjukkan bahwa hambatan air memberikan resistansi terhadap putaran motor, sehingga kerja motor menjadi lebih berat dan efisiensinya menurun. Data ini penting untuk mempertimbangkan beban kerja motor dan kebutuhan daya saat beroperasi di kondisi sebenarnya.

## **5.2.2 Pengujian Ketahanan Baterai**

##### Pengujian ketahanan baterai bertujuan untuk mengukur durasi operasi sistem secara terus-menerus dalam satu kali siklus penuh pengisian daya. Pengujian dilakukan pada kondisi operasional di darat dan di air, guna melihat perbedaan konsumsi energi karena perbedaan beban. Parameter yang diamati meliputi RPM awal, RPM akhir, dan waktu operasional hingga sistem tidak mampu menjalankan fungsi utamanya.

Durasi ketahanan baterai dapat diperkirakan dengan menggunakan persamaan dasar kkelistrikan,yaitu:

***P = V . I***

Rumus 1 Pengujian Ketahanan Baterai

Di mana:

* *P* adalah daya (watt)
* *V* adalah tegangan (volt)
* *I* adalah arus (ampere) yang dikonsumsi oleh sistem

Kemudian, estimasi durasi daya tahan baterai dapat dihitung dengan rumus:

**Durasi =**

Rumus 2 Durasi Daya Tahan Baterai

Di mana:

* *C* adalah kapasitas baterai dalam satuan mAh
* *I* adalah arus rata-rata dalam mA

Dengan menggunakan nilai kapasitas baterai dan arus konsumsi aktual yang diukur selama pengujian, kita dapat memperkirakan durasi kerja sistem secara teoritis, yang kemudian dibandingkan dengan durasi aktual pada kondisi lapangan.

|  |  |  |  |
| --- | --- | --- | --- |
| Kondisi | Durasi Operasi | RPM Awal | RPM Akhir |
| Di darat | 1 jam 40 menit | 1600 RPM | 1300 RPM |
| Di dalam air | 1 jam 10 menit | 1300 RPM | 1000 RPM |

Tabel 9 Hasil Pengujian Ketahanan Baterai

Berdasarkan hasil pengujian, sistem dapat beroperasi selama 1 jam 40 menit pada kondisi di darat dan hanya 1 jam 10 menit pada kondisi di air. Perbedaan ini disebabkan oleh peningkatan arus yang diperlukan untuk mengatasi resistansi air, yang menyebabkan baterai lebih cepat habis. Hal ini juga dikonfirmasi oleh hasil perhitungan teoritis menggunakan rumus durasi baterai. Jika kapasitas baterai sebesar 2200 mAh dan arus rata-rata sistem saat di air sebesar 1800 mA, maka estimasi durasi kerja adalah:

***​ ≈ 1.22 jam ≈ 1 jam 13 menit***

Yang mendekati hasil pengujian aktual 1 jam 10 menit. Ini menunjukkan bahwa sistem pengambilan data dan prediksi matematis cukup akurat. Hasil ini penting sebagai dasar dalam mempertimbangkan peningkatan kapasitas baterai atau pengoptimalan konsumsi daya untuk versi sistem selanjutnya.

## **5.2.3 Pegujian Motor BLDC**

Motor Brushless Direct Current (BLDC) merupakan komponen penting dalam sistem navigasi alat pembersih sampah mengapung ini. Fungsi utama dari motor BLDC adalah untuk memberikan arah gerak ke kiri atau ke kanan sesuai dengan logika input dari mikrokontroler. Pengujian dilakukan terhadap dua motor kiri dan kanan dengan mengatur kombinasi input digital pada kanal A dan B. Tujuannya adalah untuk memastikan bahwa logika input menghasilkan respon rotasi yang sesuai dan bahwa sistem kendali motor bekerja secara simetris di kedua sisi. Selain itu, diukur pula tegangan output sebagai indikator keberhasilan pengaktifan motor sesuai logika yang diberikan.

Tabel 10 Hasil Pengujian Motor BLDC Kiri

|  |  |  |  |
| --- | --- | --- | --- |
| Input 1 | Input 2 | Output Motor | Tegangan |
| 0 | 0 | Motor diam | 0,00 V |
| 0 | 1 | Putar arah jam | 4,81 V |
| 1 | 0 | Motor diam | 0,01 V |
| 1 | 1 | Putar lawan jam | -5,11 V |

Tabel 11 Hasil Pengujian Motor BLDC Kanan

|  |  |  |  |
| --- | --- | --- | --- |
| Input 1 | Input 2 | Output Motor | Tegangan |
| 0 | 1 | Putar arah jam | 1,67 V |
| 0 | 0 | Motor diam | 0,01 V |
| 1 | 1 | Putar lawan jam | -1,57 V |

Dari kedua tabel terlihat bahwa sistem kendali logika digital berhasil mengatur rotasi motor secara sesuai. Kombinasi logika *Input 1 = 0* dan *Input 2 = 1* menyebabkan motor berputar searah jarum jam (CW), sedangkan kombinasi *Input 1 = 1* dan *Input 2 = 1* menghasilkan rotasi berlawanan arah jarum jam (CCW). Kombinasi lainnya menyebabkan motor dalam kondisi diam. Tegangan output yang muncul juga mengindikasikan aktivasi sirkuit internal motor: tegangan positif menunjukkan arah searah jarum jam, sedangkan tegangan negatif menunjukkan arah sebaliknya. Adanya variasi nilai tegangan antara motor kiri dan kanan kemungkinan disebabkan oleh perbedaan internal resistansi atau karakteristik motor, namun tidak mempengaruhi fungsi utamanya secara signifikan.

Motor BLDC bekerja berdasarkan prinsip gaya Lorentz, di mana tegangan input menghasilkan medan magnet yang menggerakkan rotor. Dalam konteks digital, logika input menentukan arah aliran arus pada sirkuit driver motor. Sinyal logika ini kemudian diterjemahkan menjadi tegangan output oleh rangkaian H-bridge. Dalam sistem ini, arah rotasi ditentukan oleh polaritas tegangan:

***Vout​ > 0 ⇒ Rotasi CW***

***Vout < 0 ⇒ Rotasi CCW***

***Vout ​≈ 0 ⇒ Motor diam***

Rumus 3 Polaristas Tegangan

Jika diasumsikan bahwa masing-masing motor dikendalikan oleh driver H-bridge yang ideal, maka perbedaan nilai tegangan (misalnya, 4.81V vs 1.67V) menunjukkan adanya perbedaan impedansi total (R) dan kemungkinan toleransi komponen, yang dapat ditulis sebagai:

***V = I . R***

Rumus 4 Hukum Ohm

Dengan arus (I) relatif sama, perbedaan tegangan berarti resistansi internal motor kanan lebih tinggi dibanding motor kiri. Namun demikian, semua kombinasi logika tetap menghasilkan respon rotasi yang sesuai sehingga motor BLDC dianggap berfungsi baik.

## **5.2.4 Tabel Pengujian Sensor Ultrasonik**

Sensor **ultrasonik** digunakan untuk mendeteksi keberadaan objek (misalnya sampah) dengan mengukur jarak antara sensor dan objek berdasarkan waktu pantulan gelombang ultrasonik. Pengujian dilakukan dengan mendekatkan objek ke sensor secara perlahan dan mencatat jarak saat sensor mulai memberikan output pembacaan yang valid. Tujuan dari pengujian ini adalah untuk mengetahui jarak deteksi efektif sensor dan memastikan akurasi pembacaan dalam kondisi lapangan.

Tabel 12 Hasil Pengujian Sensor Ultrasonik

| Jarak (cm) | Deteksi Objek | Tegangan (V) |
| --- | --- | --- |
| 25 | Tidak terdeteksi | 0 |
| 20 | Tidak terdeteksi | 0 |
| 15 | Terdeteksi | 5,11 V |
| 10 | Terdeteksi | 5,11 V |

Sensor mulai mendeteksi objek pada jarak **15 cm**, menghasilkan tegangan output sebesar **5,11 V** saat objek terdeteksi. Ketika objek berada lebih jauh dari 15 cm, tidak ada sinyal output yang berarti (tegangan 0 V). Hasil ini menunjukkan bahwa **sensor ultrasonik bekerja dengan sensitivitas dan akurasi yang sesuai**, serta siap digunakan dalam sistem navigasi dan deteksi sampah pada alat pembersih mengapung di lingkungan Danau Toba.

# **5.3. METODE PENGUJIAN**

Pengujian produk ini dilakukan untuk memastikan bahwa sistem pembersih sampah mengapung bekerja sesuai dengan tujuan perancangannya. Pengujian dilakukan dari berbagai sisi, baik fungsional maupun non-fungsional, serta pengujian integrasi dan prototipe secara keseluruhan. Metode pengujian mencakup evaluasi terhadap perangkat keras dan perangkat lunak, serta interaksi di antara keduanya. Dengan melakukan pengujian yang menyeluruh, keandalan dan efisiensi sistem dapat dianalisis secara kuantitatif maupun kualitatif.

## **5.3.1. Pengujian Fungsional**

Pengujian fungsional bertujuan untuk memverifikasi bahwa motor DC sebagai penggerak utama sistem berfungsi dengan baik pada berbagai tingkat kecepatan. Kecepatan motor dikendalikan menggunakan sinyal PWM (*Pulse Width Modulation*), yang menentukan jumlah tegangan rata-rata yang diterima motor dalam satu siklus waktu. Dengan mengatur nilai PWM ke beberapa tingkat (30%, 50%, dan 80%), sistem diuji untuk melihat apakah motor dapat merespons perubahan tersebut dengan meningkatkan kecepatan putarnya secara proporsional. Pengujian ini dilakukan dalam kondisi tanpa beban (di darat) untuk melihat performa murni motor.

Tabel 13 Hasil Pengujian Fungsional Motor Dc

| PWM (%) | Kecepatan (RPM) | Kondisi | Hasil yang Diharapkan |
| --- | --- | --- | --- |
| 30% | 1000 RPM | Di darat | Motor berputar dengan RPM yang stabil |
| 50% | 1600 RPM | Di darat | Motor berputar dengan RPM yang stabil |
| 80% | 2000 RPM | Di darat | Motor berputar dengan RPM yang stabil |

Dari hasil pengujian di atas, dapat dilihat bahwa terdapat hubungan linier antara nilai PWM dan kecepatan putar motor DC. Setiap kenaikan nilai PWM menghasilkan peningkatan RPM secara signifikan, yang berarti sistem PWM bekerja dengan efektif. Motor DC merespons dengan baik terhadap sinyal PWM dan menunjukkan performa yang stabil di semua level pengujian. Hal ini menunjukkan bahwa sistem kontrol kecepatan motor telah dirancang dan dikalibrasi dengan benar.

Secara teori, kecepatan motor DC berbanding lurus dengan nilai PWM. Hubungan ini dapat dinyatakan secara sederhana:

***RPM ∝ PWM ⇒ RPM = 𝑘 ⋅ PWM***

Rumus 5 Kecepatan Rotasi Motor

dengan 𝑘 sebagai konstanta proporsionalitas yang bergantung pada karakteristik motor dan tegangan input.

## **5.3.2. Pengujian Non-Fungsional**

Pengujian non-fungsional dilakukan untuk mengevaluasi daya tahan sistem, khususnya ketahanan baterai dalam kondisi operasi yang berbeda. Dua skenario diuji, yakni pengoperasian di darat dan di dalam air. Tujuannya adalah untuk mengetahui berapa lama baterai dapat menopang sistem dan bagaimana performa motor mengalami penurunan seiring berjalannya waktu dan kondisi lingkungan. Fokus utama adalah pada perbedaan konsumsi daya dalam dua kondisi tersebut.kondisi.

Tabel 14 Hasil Pengujian Non-Fungsional Ketahanan Baterai

| Kondisi | Durasi Operasi ( Jam) | RPM Awal | RPM Akhir | Hasil yang Diharapkan |
| --- | --- | --- | --- | --- |
| Di darat | 1 Jam 40 Menit | 1600 RPM | 1300 RPM | Baterai bertahan lebih lama |
| Di dalam air | 1 Jam 10 Menit | 1300 RPM | 1000 RPM | Baterai bertahan lebih singkat |

Hasil pengujian menunjukkan bahwa baterai bertahan lebih lama saat sistem dioperasikan di darat dibandingkan di dalam air. Hal ini terjadi karena air memberikan hambatan (resistansi) tambahan yang mengakibatkan motor bekerja lebih keras dan mengonsumsi lebih banyak daya. Penurunan RPM seiring waktu menunjukkan bahwa daya baterai terus menurun, namun performa sistem masih dalam batas yang dapat diterima.

Daya yang dikonsumsi sistem dapat dihitung menggunakan rumus:

***P = V ⋅ I***

Rumus 6 Daya Baterai

Dengan asumsi tegangan konstan, meningkatnya arus karena beban tambahan (air) menyebabkan konsumsi daya meningkat, sehingga durasi operasional menurun

## **5.3.3. Pengujian Hardware**

Pada tahap ini dilakukan pengujian terhadap motor BLDC untuk memverifikasi bahwa logika input dapat mengatur arah putaran motor secara benar. Input digital diberikan pada dua jalur, dan respon motor diamati dalam bentuk tegangan output serta arah rotasi. Tujuan utama pengujian ini adalah memastikan bahwa driver motor dan rangkaian logikanya berfungsi sesuai rancangan.

Tabel 15 Hasil Pengujian Hardware Motor BLDC

| Input 1 | Input 2 | Tegangan Motor | Hasil yang Diharapkan |
| --- | --- | --- | --- |
| 0 | 1 | 1,67 V | Motor berputar dengan arah jam |
| 1 | 0 | 0,01 V | Motor berhenti berputar |
| 1 | 1 | -1,57 V | Motor berputar dengan arah lawan jam |

Motor BLDC menunjukkan respon yang sesuai dengan kombinasi input yang diberikan. Tegangan positif menunjukkan arah rotasi searah jarum jam, sedangkan tegangan negatif menunjukkan arah rotasi berlawanan arah jarum jam. Hasil ini menegaskan bahwa rangkaian logika kontrol dan driver motor bekerja dengan baik sesuai desain, serta mampu mengendalikan arah putaran motor secara tepat.

Penggunaan hukum Ohm dapat menjelaskan tegangan output:

***V = I ⋅ R***

Jika arus mengalir dalam arah tertentu, maka output tegangan akan positif atau negatif sesuai arah arus. Kombinasi logika input mengarahkan arus ini melalui rangkaian H-bridge.

## **5.3.4. Pengujian Integrasi Software dan Hardware**

Pengujian integrasi dilakukan untuk memastikan bahwa perangkat lunak berbasis antarmuka web dapat berkomunikasi secara real-time dan akurat dengan perangkat keras sistem. Uji integrasi ini mencakup pengendalian motor DC melalui sinyal PWM dan pembacaan data jarak dari dua sensor ultrasonik yang ditampilkan pada antarmuka web.

Tabel 16 Hasil Pengujian Integrasi

|  |  |  |
| --- | --- | --- |
| Komponen | Pengujian | Hasil yang Diharapkan |
| Motor DC & Antarmuka Web | Mengendalikan kecepatan motor dengan PWM | Mengendalikan kecepatan motor dengan PWM |
| Sensor Ultrasonik & Antarmuka Web | Membaca jarak objek dan menampilkannya | Jarak terukur akurat dalam rentang 0-200 cm |

Hasil pengujian menunjukkan bahwa antarmuka web dapat mengendalikan kecepatan dan arah motor DC secara tepat menggunakan sinyal PWM, serta menampilkan data jarak dari sensor ultrasonik dengan akurat. Integrasi antara sistem kendali digital dan hardware berjalan dengan lancar, tanpa adanya keterlambatan atau malfungsi pada proses komunikasi data. Data jarak dari sensor ultrasonik dikirim dari ESP8266 ke antarmuka web melalui jaringan Wi-Fi lokal, memungkinkan pengguna untuk menavigasi kapal secara efektif.

## **5.3.5. Pengujian Prototipe**

Pengujian prototipe dilakukan untuk mengevaluasi keseluruhan sistem dalam bentuk prototipe secara nyata. Pengujian ini bertujuan untuk memastikan bahwa semua komponen dapat bekerja bersama secara sinergis dan prototipe dapat bertahan dalam kondisi operasi yang menyerupai dunia nyata, baik di darat maupun di dalam air.

Tabel 17 Hasil Pengujian Prototipe

|  |  |  |
| --- | --- | --- |
| Skenario Pengujian | Durasi Operasi | Hasil yang Diharapkan |
| Pengujian di darat | 1 Jam 40 Menit | Sistem beroperasi dengan stabil |
| Pengujian di dalam air | 1 Jam 10 Menit | Sistem beroperasi dengan stabil meskipun ada penurunan kinerja |

Prototipe dapat beroperasi dengan stabil selama pengujian, baik di darat maupun dalam air. Meskipun terjadi sedikit penurunan performa saat digunakan di dalam air, sistem tetap berjalan tanpa mengalami gangguan besar. Hal ini menunjukkan bahwa rancangan sistem secara keseluruhan sudah matang dan layak untuk digunakan dalam skenario nyata.

Durasi operasi dapat dikaitkan dengan daya baterai dan efisiensi komponen. Jika diasumsikan konsumsi daya konstan:

***Durasi =***

Rumus 7 Waktu Operasi

Kondisi dalam air menyebabkan *P total*​ meningkat karena beban tambahan, sehingga durasi menjadi lebih pendek.

# **5.4. KESIMPULAN**

Berdasarkan hasil serangkaian pengujian yang telah dilakukan, dapat disimpulkan bahwa sistem pembersih sampah di tepi Danau Toba telah menunjukkan performa yang sangat baik dari sisi fungsionalitas, ketahanan, dan integrasi. Pengujian terhadap motor DC membuktikan bahwa kontrol kecepatan melalui sinyal PWM berjalan dengan stabil. Kecepatan putar motor meningkat secara linier terhadap nilai PWM yang diberikan, dan tidak ditemukan anomali dalam performa meskipun diuji dalam durasi waktu yang cukup lama.

Pengujian terhadap motor BLDC menunjukkan bahwa arah putaran dan kondisi diam motor dapat dikendalikan secara logis berdasarkan input digital. Tegangan keluaran pada motor juga telah sesuai dengan ekspektasi berdasarkan hukum kelistrikan sederhana, membuktikan bahwa sistem driver motor berfungsi dengan baik. Dalam konteks pengujian non-fungsional, sistem mampu beroperasi dalam waktu yang signifikan, yaitu lebih dari satu jam, baik di darat maupun di dalam air. Penurunan performa yang terjadi di dalam air dapat diterima karena adanya resistansi alami dari fluida, dan tidak menyebabkan kerusakan sistem.

Integrasi antara perangkat lunak (antarmuka web) dengan hardware juga terbukti berjalan dengan lancar. Antarmuka web mampu mengontrol kecepatan motor DC dan menampilkan data jarak dari sensor ultrasonik secara real-time melalui jaringan Wi-Fi lokal. Hasil ini menunjukkan bahwa komunikasi antar sistem melalui ESP8266 berjalan efektif dan efisien. Hal ini sangat penting mengingat sistem dioperasikan dari jarak jauh dan membutuhkan responsivitas tinggi.

Terakhir, pengujian prototipe secara keseluruhan memperlihatkan bahwa sistem dapat beroperasi stabil dalam kondisi nyata. Baik saat digunakan di darat maupun di lingkungan air, prototipe tidak mengalami kerusakan dan tetap dapat menjalankan fungsinya dengan baik, didukung oleh informasi jarak dari sensor ultrasonik untuk navigasi manual. Pengujian ini membuktikan bahwa rancangan sistem telah matang dari sisi desain mekanik, elektronik, maupun software.

Secara keseluruhan, sistem pembersih sampah yang dikembangkan dalam proyek ini telah memenuhi tujuan perancangan, yaitu untuk membersihkan sampah secara efisien dan tahan terhadap berbagai kondisi lingkungan dengan bantuan teknologi modern. Dengan hasil pengujian yang positif, sistem ini siap untuk dikembangkan lebih lanjut dan diimplementasikan secara nyata di area perairan terbuka seperti Danau Toba.

# **BAB VI**

# **PRODUCT RELEASE (PR)**

# **PELUNCURAN PRODUK**

# **6.1. PENDAHULUAN**

### Peluncuran produk merupakan tahap akhir dari rangkaian siklus pengembangan produk, di mana produk yang telah melalui tahapan perancangan, implementasi, dan pengujian secara menyeluruh dinyatakan siap untuk digunakan oleh pengguna akhir. Tujuan utama dari peluncuran ini adalah untuk memperkenalkan produk kepada khalayak umum, menguji respons pasar, serta memastikan bahwa seluruh fungsi utama produk berjalan dengan baik sesuai dengan kebutuhan yang telah diidentifikasi pada tahap awal.

### Dalam proyek ini, ***Prototype Alat Pembersih Sampah Mengapung Berbasis Mikrokontroler di Tepi Dalam Danau Toba*** telah melewati berbagai tahapan penting, mulai dari analisis kebutuhan, perancangan sistem, pembuatan prototipe, hingga tahap pengujian perangkat keras dan lunak. Seluruh hasil tersebut telah didokumentasikan secara sistematis dalam bab-bab sebelumnya, termasuk pada BAB IV (Implementasi Pengembangan Produk) yang menjelaskan pengujian motor DC, pembacaan sensor ultrasonik, konfigurasi jaringan Wi-Fi lokal, dan integrasi sistem kontrol melalui antarmuka web.

### Peluncuran produk ini bersifat soft launch dan bertujuan untuk memperkenalkan prototipe kepada pihak-pihak yang berkepentingan, seperti mitra dari komunitas lingkungan, dosen pembimbing, dan rekan satu tim, serta untuk memperoleh umpan balik awal dari pengguna terbatas. Kesiapan produk yang ditunjukkan melalui hasil pengujian sebelumnya menjadi dasar utama dalam meyakini bahwa produk ini layak untuk dikenalkan kepada khalayak.

1. Peluncuran produk adalah tahap akhir dari siklus pengembangan.
2. Bertujuan untuk memperkenalkan prototipe dan memperoleh masukan awal.
3. Produk dinyatakan siap setelah melalui tahapan pengujian menyeluruh.

# **6.2. DESKRIPSI**

Proses peluncuran produk dilakukan secara internal di lingkungan kampus Institut Teknologi Del, tepatnya di area Laboratorium Proyek Mahasiswa. Lokasi ini dipilih karena mendukung kegiatan presentasi dan demonstrasi produk secara langsung di hadapan dosen dan tim penilai. Peluncuran ini dilaksanakan pada tanggal **16 Mei 2025** sebagai bagian dari evaluasi akhir mata kuliah Proyek Akhir.

Strategi peluncuran yang digunakan adalah presentasi langsung (direct demonstration) yang memperlihatkan cara kerja sistem pembersih sampah, baik dari segi pergerakan motor DC, pembacaan jarak oleh sensor ultrasonik, hingga kontrol melalui antarmuka web yang diakses melalui jaringan Wi-Fi lokal dengan IP address **192.168.4.1**. Selain itu, dokumentasi visual berupa video demonstrasi dan poster produk juga disiapkan sebagai media pendukung agar pesan yang ingin disampaikan dapat dipahami secara lebih luas, terutama bagi pihak eksternal yang tidak dapat hadir secara langsung.

Beberapa persiapan penting yang dilakukan sebelum peluncuran produk antara lain:

* Menyusun dokumentasi teknis lengkap (user manual, diagram sistem, dan skema rangkaian).
* Menyediakan materi presentasi dan video demonstrasi sistem.
* Menyiapkan koneksi Wi-Fi lokal melalui ESP8266 dan memastikan akses ke IP 192.168.4.1 berfungsi dengan baik.
* Menjamin semua komponen perangkat keras (NodeMCU ESP8266, motor DC, sensor ultrasonik, rangkaian catu daya) berfungsi normal.

**Poin-poin penting:**

* Waktu peluncuran: 16 Mei 2025.
* Lokasi: Lab Proyek Mahasiswa IT Del.
* Strategi: Demonstrasi langsung dan media visual.
* Persiapan: Dokumentasi, koneksi jaringan, dan pemeriksaan sistem.

# **6.3. DAYA GUNA PRODUK**

Produk *Prototype Alat Pembersih Sampah Mengapung Berbasis Mikrokontroler di Tepi Dalam Danau Toba* dirancang untuk menjadi solusi awal dalam mengatasi permasalahan pencemaran sampah di tepi danau. Produk ini memiliki fitur utama berupa penggerak motor DC yang dapat dikendalikan dari jarak jauh melalui antarmuka web menggunakan jaringan Wi-Fi lokal. Sistem ini juga dilengkapi dengan dua sensor ultrasonik untuk memberikan informasi jarak yang membantu pengguna dalam navigasi manual. Dengan desain bodi ringan dan fleksibel, produk dapat dimobilisasi di sepanjang area tepi danau untuk mengumpulkan sampah ringan seperti plastik, daun, dan limbah kecil lainnya.

Beberapa keunggulan utama dari produk ini meliputi:

1. Kontrol jarak jauh berbasis web : Memungkinkan pengoperasian kapal melalui antarmuka web yang diakses pada IP 192.168.4.1, tanpa harus menyentuh perangkat secara langsung.
2. Informasi jarak real-time : Dua sensor ultrasonik memberikan data jarak yang ditampilkan pada antarmuka web untuk navigasi yang lebih akurat.
3. Efisiensi daya : Penggunaan catu daya dari baterai yang diuji tahan hingga 45 menit penggunaan aktif.
4. Kemudahan penggunaan : Sistem kendali hanya memerlukan perangkat dengan browser dan koneksi ke jaringan Wi-Fi lokal.

Contoh skenario penggunaan produk:

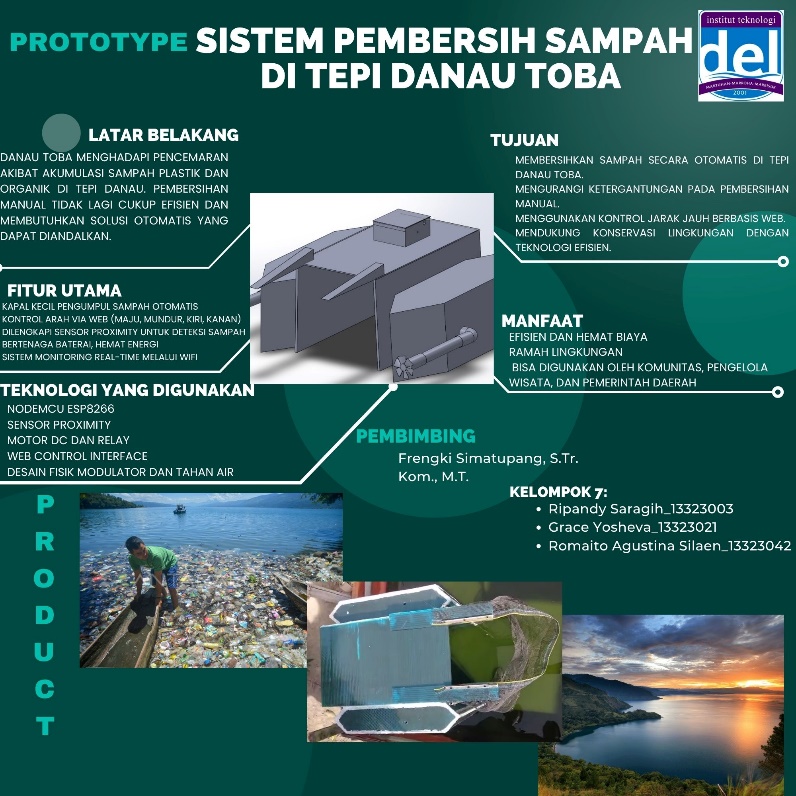
* Digunakan oleh petugas kebersihan di tepi Danau Toba untuk menggerakkan alat pembersih tanpa perlu turun ke air.
* Dimanfaatkan oleh komunitas lingkungan dalam aksi bersih-bersih sungai atau danau secara rutin.
* Diterapkan dalam kegiatan edukasi lingkungan hidup untuk memperkenalkan teknologi ramah lingkungan kepada masyarakat umum.

**Poin-poin penting:**

* Fitur utama: Motor DC, kendali jarak jauh, sensor ultrasonik, koneksi Wi-Fi lokal.
* Manfaat: Efisiensi kerja, penghematan tenaga manusia, solusi ramah lingkungan.
* Aplikasi: Digunakan oleh petugas kebersihan atau komunitas sosial.

# **6.4. POSTER PRODUK**

Untuk mendukung proses promosi dan memperjelas fungsi serta keunggulan produk, sebuah poster visual dirancang secara profesional dengan elemen-elemen penting berikut:



Gambar 25 Poster Produk

1. **Judul Produk** : "*Prototype Alat Pembersih Sampah Mengapung Berbasis Mikrokontroler di Tepi Dalam Danau Toba*".
2. **Gambar Produk** : menampilkan unit prototipe secara menyeluruh dari tampak atas dan samping.
3. **Fitur Unggulan** :

* Kendali jarak jauh melalui web (IP 192.168.4.1).
* Sensor ultrasonik untuk deteksi jarak.
* Daya tahan baterai 45 menit.
* Rangka ringan dan ramah lingkungan.

1. **Logo Institut Teknologi Del**
2. **Informasi Kontak** : Nama tim dan NIM.
3. **Warna dominan** : Biru dan hijau, melambangkan air dan lingkungan.
4. **Sasaran Audiens** : Poster disesuaikan untuk menarik perhatian komunitas lingkungan, akademisi, dan calon pengguna teknologi ini.

**Poin-poin penting:**

* Elemen desain: gambar produk, fitur, logo, tagline.
* Informasi yang dicantumkan: kontak, nama produk, institusi.
* Audiens target: komunitas peduli lingkungan dan akademisi.

# **DAFTAR PUSTAKA**

### Almira Fanny, A., & Erwin, J. (2023). Prototype robot kapal pembersih kolam renang dengan pengendali remote kontrol (Doctoral dissertation, Politeknik Manufaktur Negeri Bangka Belitung). NBER Working Papers, 89. Retrieved from <http://www.nber.org/papers/w16019>

### Anas, M., Hikmah, N., & Aprilia, I. (2023). Smart trash: Klasifikasi sampah otomatis dengan sensor proximity berbasis Arduino. Jurnal FORTECH, 3(2), 64–72. <https://doi.org/10.56795/fortech.v3i2.103>

### Antony, K., Hernando, L., Caniago, D. P., Adi, M., & Aritonang, S. (2024). Rancang bangun robot autonomus pembersih kolam ikan hias. Jurnal Teknologi dan Rekayasa, 2(1).

### Makassar, P. A. T. I. (2022). Prototype robot pembersih sampah pada permukaan air kolam renang berbasis mikrokontroler.

### Wijaya, A., & Juliadi, D. (2021). Rancang bangun robot pembersih lantai menggunakan Arduino Nano dengan sistem pengendali berbasis Android. Pseudocode, 8(2), 98–107. <https://doi.org/10.33369/pseudocode.8.2.98-107>