

## **TUGAS AKHIR**

# **IMPLEMENTASI SISTEM PENYEMPROTAN PESTISIDA OTOMATIS BERBASIS *INTERNET OF THINGS* (IOT)**

*Sebagai salah satu syarat untuk menyelesaikan  
Program Studi Diploma III Jurusan Teknik Elektro*



**Disusun Oleh:**

**INDAH TRIA ALFINA**

**3103211297**

**PROGRAM STUDI D-III TEKNIK ELEKTRONIKA  
JURUSAN TEKNIK ELEKTRO  
POLITEKNIK NEGERI BENGKALIS  
TAHUN 2024**

## LEMBAR PENGESAHAN

### IMPLEMENTASI SISTEM PENYEMPROTAN PESTISIDA OTOMATIS BERBASIS *INTERNET OF THINGS* (IOT)

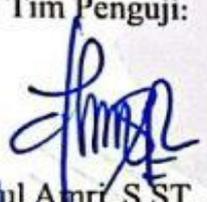
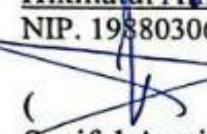
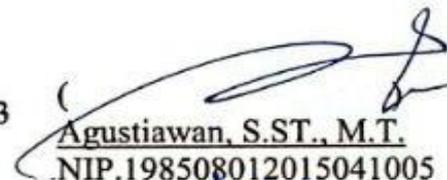
*Sebagai salah satu syarat untuk menyelesaikan  
Program Studi Diploma III Jurusan Teknik Elektro*

Oleh:

Indah Tria Alfina  
3103211297

Disetujui oleh Tim Penguji:

Tanggal Ujian : 16 Agustus 2024  
Periode Wisuda : XXI

- 1 (  ) (Pembimbing)  
Hikmatul Amri, S.ST., M.T.  
NIP. 198803062018031001
- 2 (  ) (Penguji 1)  
Syaiful Amri, S.ST., MT  
NIP. 198308302021211005
- 3 (  ) (Penguji 2)  
Agustiawan, S.ST., M.T.  
NIP.198508012015041005
- 4 (  ) (Penguji 3)  
Abdul Hadi, ST., M.T.  
NIP. 199001182019031017

Bengkalis, 23 Agustus 2024  
Ketua Program Studi Teknik Elektronika  
Politeknik Negeri Bengkalis

  
Abdul Hadi, ST., M.T.  
NIP:199001182019031017

## HALAMAN PENGESAHAN

Kami dengan sebenarnya menyatakan bahwa, kami telah membaca keseluruhan dari Tugas Akhir ini, dan kami berpendapat bahwa Tugas Akhir ini layak dan memenuhi syarat untuk memperoleh gelar Ahli Madya.

Tanda tangan :   
Penguji 1 : Syaiful Amri, S.ST., MT  
Tanggal Pengujian : 16 Agustus 2024

Tanda tangan :   
Penguji 2 : Agustiawan, S.ST., M.T.  
Tanggal Pengujian : 16 Agustus 2024

Tanda tangan :   
Penguji 3 : Abdul Hadi, ST., M.T.  
Tanggal Pengujian : 16 Agustus 2024

## PERNYATAAN KEASLIAN TUGAS AKHIR

Saya menyatakan dengan sesungguhnya bahwa Tugas Akhir dengan judul Implementasi Sistem Penyemprotan Pestisida Otomatis Berbasis *Internet of Things* (IoT) adalah hasil karya yang pernah dilakukan untuk memperoleh gelar ahli madya diperguruan tinggi dan sepanjang pengetahuan saya juga tidak terdapat atau pendapat yang pernah ditulis atau dipublikasikan oleh orang lain. Kecuali secara tertulis disebutkan sumbernya dalam naskah dan dalam daftar pustaka.

Bengkalis, 16 Agustus 2024



Indah Tria Alfina  
NIM: 3103211297

## KATA PENGANTAR

Assalamualikum warahmatullahi wabarakatuh

Alhamdulillah rabbil 'aalamiin penulis ucapkan Puji Syukur kehadiran Allah SWT yang telah memberikan Rahmat dan Hidayah-Nya, sehingga penulis dapat menyelesaikan sekaligus menyusun laporan Tugas Akhir sebagai salah satu syarat penulis dalam menyelesaikan Program Studi D-III Teknik Elektronika Jurusan Teknik Elektro Politeknik Negeri Bengkalis dengan judul Implementasi Sistem Penyemprotan Pestisida Otomatis Berbasis *Internet of Things* (IoT).

Penyusunan laporan Tugas Akhir ini dapat terselesaikan dengan baik tidak terlepas dari bantuan serta dukungan dari semua pihak. Oleh karena itu, penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Ayahanda tercinta dan Almarhumah Ibunda yang senantiasa memberikan kasih sayang, semangat, serta do'a yang selalu beliau panjatkan tanpa henti untuk penulis.
2. Bapak Johny Custer, S.T.,M.T., selaku Direktur Politeknik Negeri Bengkalis.
3. Bapak Hikmatul Amri, S.ST.,M.T., selaku Dosen Pembimbing Tugas Akhir serta selaku Kordinator Tugas Akhir.
4. Bapak Abdul Hadi, ST., M.T., selaku ketua program studi D-III Teknik Elektronika.
5. Bapak Aris Hidayat yang telah memberikan dukungan serta nasehat dan juga telah membantu dalam menyelesaikan Tugas Akhir ini.
6. Seluruh staf pengajar dan staf administrasi Jurusan Teknik Elektro.
7. Kepada teman-teman seperjuangan khususnya program studi Teknik Elektronika B yang memberi semangat serta dukungan dalam menyelesaikan Tugas Akhir.

Dengan segala kerendahan hati penulis menyadari masih banyak kekurangan dan kesalahan dalam pembuatan Tugas Akhir, sehingga penulis mengharapkan adanya kritik dan saran yang membangun dalam upaya

menyempurnakan laporan Tugas Akhir ini dan perbaikan di kemudian hari. Akhir kata, penulis berharap agar laporan ini dapat bermanfaat bagi semuanya.

Wassalamualaikum warahmatullahi wabarakatuh.

Bengkalis, 14 Agustus 2024

Indah Tria Alfina  
3103211297

# IMPLEMENTASI SISTEM PENYEMPROTAN PESTISIDA OTOMATIS BERBASIS *INTERNET OF THINGS* (IOT)

Nama : Indah Tria Afina  
NIM : 3103211297  
Dosen pembimbing : Hikmatul Amri, S.ST., M.T.

## ABSTRAK

Penyemprotan pestisida masih dilakukan secara manual oleh sebagian petani di Indonesia dengan memanfaatkan tangki semprot yang dioperasikan dengan cara menggondong tangki tersebut, seringkali penyempotan yang dilakukan tidak merata dan penggunaan pestisida yang boros. Masalah tersebut dapat diselesaikan dengan cara membuat alat penyemprotan pestisida otomatis berbasis *internet of things* (IoT). Sistem ini terdiri dari mikrokontroler NodeMCU ESP8266 sebagai kontroler dan sensor PZEM-004T yang digunakan untuk mendeteksi tegangan dan arus pompa air, RTC digunakan sebagai pengaturan waktu. Alat ini dapat dikontrol melalui jarak jauh di manapun dan kapan saja. Sehingga tanaman terawat dengan optimal dan dapat menghasilkan hasil panen yang berkualitas. Hasil pengujian menunjukkan bahwa alat ini dapat melakukan penyemprotan secara merata dan berhasil melakukan penyemprotan otomatis. Pompa air berkerja dengan tegangan 239 V dan arus 1,6 A dengan waktu siram selama 1 menit dengan nilai rata-rata tegangan 238,8 V dan arus rata-rata 1,57 A, alat ini berkerja dengan baik dengan tingkat keberhasilan 100 % berhasil.

**Kata kunci:** Pestisida, Penyemprotan Otomatis, IoT, NodeMCU ESP8266

# **IMPLEMENTATION OF AUTOMATIC PESTICIDE SPRAYING SYSTEM BASED ON INTERNET OF THINGS (IOT)**

*Name : Indah Tria Afina  
NIM : 3103211297  
Supervisor : Hikmatul Amri, S.ST., M.T.*

## **ABSTRACT**

*Spraying pesticides is still done manually by some farmers in Indonesia by utilizing a spray tank that is operated by carrying the tank, often the spraying is uneven and the use of pesticides is wasteful. The problem can be solved by making an automatic pesticide spraying tool based on the internet of things (IoT). This system consists of a NodeMCU ESP8266 microcontroller as a controller and a PZEM-004T sensor used to detect the voltage and current of the water pump, RTC is used as a time setting. This tool can be controlled remotely anywhere and anytime. So that plants are optimally maintained and can produce quality crops. The test results show that this tool can spray evenly and successfully perform automatic spraying. The water pump works with a voltage of 239 V and a current of 1.6 A with a flush time of 1 minute with an average voltage of 238.8 V and an average current of 1.57 A, this tool works well with a success rate of 100% success.*

**Keywords:** *Pesticide, Automatic Spraying, IoT, NodeMCU ESP8266*

## DAFTAR ISI

|   |           |
|---|-----------|
| HALAMAN DEPAN .....                     | i         |
| LEMBAR PENGESAHAN .....                 | ii        |
| HALAMAN PENGESAHAN .....                | iii       |
| PERNYATAAN KEASLIAN TUGAS AKHIR.....    | iv        |
| KATA PENGANTAR.....                     | v         |
| ABSTRAK .....                           | vii       |
| <i>ABSTRACT</i> .....                   | viii      |
| DAFTAR ISI.....                         | ix        |
| DAFTAR GAMBAR.....                      | xii       |
| DAFTAR TABEL .....                      | xiii      |
| <b>BAB I PENDAHULUAN.....</b>           | <b>1</b>  |
| <b>1.1 Latar Belakang .....</b>         | <b>1</b>  |
| <b>1.2 Rumusan Masalah.....</b>         | <b>2</b>  |
| <b>1.3 Batasan Masalah .....</b>        | <b>3</b>  |
| <b>1.4 Tujuan Penelitian .....</b>      | <b>3</b>  |
| <b>1.5 Manfaat Penelitian .....</b>     | <b>3</b>  |
| <b>1.6 Sistematika Penulisan .....</b>  | <b>4</b>  |
| <b>BAB II TINJAUAN PUSTAKA.....</b>     | <b>5</b>  |
| <b>2.1 Kajian Terdahulu .....</b>       | <b>5</b>  |
| <b>2.2 Landasan Teori.....</b>          | <b>8</b>  |
| 2.2.1 Cabai.....                        | 8         |
| 2.2.2 <i>Internet of Things</i> .....   | 9         |
| 2.2.3 Arduino IDE .....                 | 9         |
| 2.2.4 Blynk .....                       | 12        |
| <b>2.3 Komponen yang Digunakan.....</b> | <b>12</b> |
| 2.3.1 NodeMCU ESP8266 .....             | 12        |
| 2.3.2 Modul <i>Relay</i> .....          | 14        |
| 2.3.3 RTC .....                         | 15        |

|  |   |           |
|--|---|-----------|
| 2.3.4  | Sensor PZEM-004T.....   | 15        |
| 2.3.5  | Kabel <i>Jumper</i> .....   | 16        |
| 2.3.6  | <i>Power Supply</i> .....   | 17        |
| 2.3.7  | <i>Sprayer</i> .....  | 17        |
| 2.3.8  | Pompa Air.....  | 18        |
| 2.3.9  | <i>Push Button</i> .....  | 18        |
| <b>BAB III METODOLOGI PENELITIAN .....</b>       |   | <b>20</b> |
| 3.1  | <b>Tinjauan Umum .....</b>  | <b>20</b> |
| 3.2  | <b>Blok Diagram.....</b>  | <b>20</b> |
| 3.3  | <b><i>Flowchart</i>.....</b>                                      | <b>22</b> |
| 3.4  | <b>Rancangan <i>Hardware</i> .....</b>                            | <b>27</b> |
| 3.5  | <b>Rancangan <i>Software</i> .....</b>                            | <b>28</b> |
| 3.5.1  | Rancangan Program untuk RTC.....                                  | 28        |
| 3.5.2  | Rancangan Program Untuk Sensor PZEM-004T .....                    | 28        |
| 3.5.3  | Rancangan Program Untuk <i>Relay</i> dan <i>Push Button</i> ..... | 29        |
| 3.6  | <b>Rancangan <i>Prototype</i> Alat Keseluruhan .....</b>          | <b>29</b> |
| <b>BAB IV HASIL PENELITIAN DAN ANALISA .....</b> |   | <b>31</b> |
| 4.1.   | <b>Hasil Perancangan Alat .....</b>                               | <b>31</b> |
| 4.2.   | <b>Hasil Pengujian Alat .....</b>                                 | <b>33</b> |
| 4.2.1.   | <b>Pengujian RTC DS3231 .....</b>                                 | <b>33</b> |
| 4.2.2.   | <b>Pengujian Modul <i>Relay</i> .....</b>                         | <b>34</b> |
| 4.2.3.   | <b>Pengujian <i>Push Button</i> .....</b>                         | <b>35</b> |
| 4.2.4.   | <b>Pengujian Sensor PZEM-004T .....</b>                           | <b>35</b> |
| 4.2.5.   | <b>Pengujian Aplikasi Blynk .....</b>                             | <b>36</b> |
| 4.2.6.   | <b>Pengujian <i>Power Supply</i> 5 V .....</b>                    | <b>36</b> |
| 4.3.   | <b>Pengujian Alat Keseluruhan .....</b>                           | <b>37</b> |
| <b>BAB V KESIMPULAN DAN SARAN .....</b>          |   | <b>39</b> |
| 5.1.   | <b>Kesimpulan .....</b>   | <b>39</b> |
| 5.2.   | <b>Saran.....</b>   | <b>39</b> |
| <b>DAFTAR PUSTAKA .....</b>                      |   | <b>41</b> |
| <b>LAMPIRAN.....</b>                             |   | <b>43</b> |

**Lampiran 1 Program Alat Keseluruhan**

**Lampiran 2 Perbaikan Dan Saran**

**Lampiran 3 Lembar Asestensi**

## DAFTAR GAMBAR

|  |    |
|--|----|
| Gambar 2.1 Cabai Rawit .....   | 9  |
| Gambar 2.2 Arduino IDE .....   | 10 |
| Gambar 2.3 Aplikasi Blynk.....   | 12 |
| Gambar 2.4 NodeMCU ESP8266 .....   | 13 |
| Gambar 2.5 Modul <i>Relay</i> .....  | 14 |
| Gambar 2.6 <i>Real Time Clock</i> (RTC).....                                 | 15 |
| Gambar 2.7 Sensor PZEM-004T.....   | 16 |
| Gambar 2.8 Kabel <i>Jumper</i> .....   | 16 |
| Gambar 2.9 <i>Power Supply</i> .....   | 17 |
| Gambar 2.10 <i>Sprayer</i> .....   | 17 |
| Gambar 2.11 Pompa Air .....  | 18 |
| Gambar 2.12 <i>Push Button</i> .....   | 19 |
| Gambar 3.1 Blok Diagram .....  | 21 |
| Gambar 3.3 <i>Flowchart</i> Sistem Kerja Alat Keseluruhan .....              | 22 |
| Gambar 3.4 <i>Flowchart</i> Sistem Kerja Alat Otomatis.....                  | 24 |
| Gambar 3.5 <i>Flowchart</i> Sistem Kerja Secara Manual.....                  | 26 |
| Gambar 3.6 Rancangan <i>Hardware</i> .....                                   | 27 |
| Gambar 3.7 Rancangan Program Untuk RTC.....                                  | 28 |
| Gambar 3.8 Rancangan Program Untuk Sensor PZEM-004T.....                     | 29 |
| Gambar 3.9 Rancangan Program Untuk <i>Relay</i> Dan <i>Push Button</i> ..... | 29 |
| Gambar 3.10 Rancangan Tampak Depan.....                                      | 30 |
| Gambar 3.11 Rancangan Tampak Atas.....                                       | 30 |
| Gambar 3.12 Rancangan <i>Prototype</i> .....                                 | 30 |
| Gambar 4.1 Tampak Dalam .....  | 31 |
| Gambar 4.2 Tampak Samping .....  | 31 |
| Gambar 4.3 <i>Sprayer</i> dan Tanaman Cabai .....                            | 32 |
| Gambar 4.4 Pompa Air Dan Ember .....   | 32 |

## DAFTAR TABEL

|  |    |
|--|----|
| Tabel 2.1 Spesifikasi NodeMCU ESP8266 .....      | 13 |
| Tabel 2.2 Spesifikasi Modul <i>Relay</i> .....   | 14 |
| Tabel 2.3 Spesifikasi Sensor PZEM-004T.....      | 16 |
| Tabel 2.4 Spesifikasi Pompa Air.....             | 18 |
| Tabel 4.1 Pengujian RTC DS3231 .....             | 33 |
| Tabel 4.2 Pengujian Modul <i>Relay</i> .....     | 34 |
| Tabel 4.3 Pengujian <i>Push Button</i> .....     | 35 |
| Tabel 4.4 pengujian Sesor PZEM-004T.....         | 35 |
| Tabel 4.5 Pengujian Aplikasi Blynk .....         | 36 |
| Tabel 4.6 Pengujian <i>Power Supply</i> .....    | 36 |
| Tabel 4.7 Pengujian Alat Secara Keseluruhan..... | 37 |

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Pengaruh globalisasi dan perkembangan teknologi pertanian menyebabkan dunia pertanian Indonesia harus segera memberikan tanggapan untuk terus bersaing, salah satu caranya adalah dengan menaikkan mutu dan hasil produktivitas hasil pertanian. Salah satu sayuran hasil holtikultural ini yang terus menaikkan hasil produksi yaitu cabai. Cabai merupakan bahan dasar dalam membuat sebuah masakan dan memiliki perubahan harga jual yang bervariasi. Adanya harga jual cabai di pasaran yang bervariasi disebabkan oleh rendahnya hasil panen akibat hama dan penyakit pada tanaman cabai. Penyakit dan hama ini membuat produktivitas tanaman cabe berkurang, di antara hama dan penyakit yang dapat merusak tanaman cabai yaitu lalat buah, kutu kebul, thrips, kutu daun, kutu daun persik, tungau, layu fusarium, penyakit layu bakteri *Ralstonia*, penyakit virus kuning, penyakit bercak daun, dan penyakit busuk buah. Dalam mengatasi hama dan mencegah rendahnya hasil panen maka petani perlu melakukan penyemprotan tanaman cabai dalam rentang waktu 3 hari sekali untuk mencegah rusaknya daun muda maupun buah cabai, pestisida berperan penting dalam mencegah hama dan penyakit yang tak terhindarkan. Pestisida merupakan jenis zat kimia yang digunakan dalam mengatasi hama dan penyakit terhadap tanaman khususnya pada tanaman cabai. Penyemprotan pestisida dilakukan agar mendapatkan hasil panen yang bagus yang dapat dijual di pasaran dengan harga tertentu. Pestisida yang beredar di pasaran saat ini sangat beragam dan jumlahnya sangat banyak. Kondisi seperti ini menyebabkan petani mengalami kesulitan untuk menentukan pilihan ketika hendak memilih. Terdapat sebanyak 3207 pestisida yang diizinkan oleh kementerian pertanian Indonesia 2016. Hal ini merupakan bentuk dukungan pemerintah terhadap petani sebagai penekanan bahkan pencegahan terhadap kegagalan panen dan meningkatkan hasil panen petani (Budiono, 2021).

Penyemprotan pestisida yang dilakukan sebagian besar petani di Indonesia masih menggunakan penyemprotan manual dengan memanfaatkan tangki semprot yang dioperasikan dengan cara menggendong tangki tersebut. Perlakuan tersebut menuntut petani menggunakan tenaga yang lebih besar. Dampak dari penyemprotan secara langsung tersebut bagi petani adalah terhirupnya cairan pestisida yang mengandung bahan kimia sehingga bisa menyebabkan gangguan pada tubuh di antaranya adalah merasakan mual, pusing dan mata berair (Dwiningtias, 2023).

Masuknya era revolusi industri 4.0 di mana keberadaan *internet of things* (IoT) menjadi salah satu ciri utamanya telah membuat perubahan yang sangat signifikan di berbagai sektor kehidupan. IoT dipandang sebagai sebuah solusi cerdas yang menjadikan manusia dan berbagai benda, objek ataupun perangkat yang ada di alam nyata bisa saling terhubung dan saling berkomunikasi dalam sebuah sistem yang terintegrasi dengan menggunakan jaringan internet sebagai penghubungnya. Hal ini memiliki tujuan agar penggunaanya bisa mengambil informasi semua benda, objek atau perangkat tersebut kapan pun dan di mana pun, untuk kemudian bisa mengambil keputusan untuk melakukan suatu tindakan yang tepat Berdasarkan informasi tersebut (Utomo, 2019).

Berdasarkan dari latar belakang yang ada, penulis memiliki gagasan merancang dan membuat alat yang dapat melakukan kontrol dan *monitoring* melalui *smartphone* menggunakan *internet of things*. Dengan alat tersebut diharapkan bisa mempermudah pekerjaan para petani dalam melakukan penyemprotan pestisida.

## **1.2 Rumusan Masalah**

Berdasarkan latar belakang di atas maka peneliti merumuskan masalah sebagai berikut:

1. Bagaimana merancang sistem penyemprotan pestisida otomatis pada tanaman cabai berbasis *internet of things* (IoT)?
2. Bagaimana cara kerja penyemprotan pestisida otomatis pada tanaman cabai?

3. Bagaimana alat ini dapat bekerja dengan baik dalam melakukan penyemprotan pestisida?

### **1.3 Batasan Masalah**

Agar pembahasan ini teratur dan terarah dengan baik, perlu dibuat batasan masalah supaya pembahasan bisa berjalan sesuai yang direncanakan. Adapun batasan masalah dalam penelitian ini yaitu:

1. Berfokus pada sistem penyemprotan pestisida otomatis.
2. Jenis tanaman yang digunakan yaitu tanaman cabai.
3. Jenis pestisida yang digunakan pada penelitian ini yaitu Agrimec dan Pegasus.
4. Menggunakan *real time clock* (RTC).
5. Perancangan dan pembuatan alat ini berbasis *internet of things* (IoT).
6. Menggunakan NodeMCU ESP8266.

### **1.4 Tujuan Penelitian**

Tujuan penelitian yang akan dilakukan adalah sebagai berikut:

1. Untuk mengurangi risiko penyakit dan hama tanaman melalui pengaturan penyemprotan yang tepat.
2. Untuk menghasilkan penyemprotan secara terukur dan hasil semprotan sampai ke seluruh tanaman.

### **1.5 Manfaat Penelitian**

Adapun manfaat dari penelitian ini yaitu:

1. Memanfaatkan IoT sebagai pusat pengendali dari sistem penyemprotan pestisida otomatis.
2. Memudahkan petani dalam melakukan penyemprotan.
3. Mencegah terjadinya kontaminasi zat kimia terhadap petani akibat penyemprotan secara langsung.

## 1.6 Sistematika Penulisan

Memberikan gambaran secara garis besar, dalam hal ini dijelaskan dari masing-masing bab dari tugas akhir ini. Sistematika penulisan dalam pembuatan laporan ini sebagai berikut:

1. Bagian pendahuluan

Berisi tentang latar belakang mengapa penulis mengambil judul Implementasi Sistem Penyemprotan Pestisida Otomatis Berbasis *Internet of Things* (IoT), rumusan masalah, batasan masalah, tujuan, manfaat dan sistematika penulisan.

2. Bagian tinjauan pustaka

Berisi tentang kajian terdahulu, landasan teori dan penjelasan komponen yang digunakan.

3. Bagian metodologi penelitian

Berisi mengenai tujuan umum, blok diagram, *flowchart*, rancangan *hardware*, rancangan *software*, rancangan *prototype* secara keseluruhan.

4. Bagian hasil dan pembahasan

Berisi tentang hasil perancangan, pengujian alat dan pengambilan data dari alat yang diujikan.

5. Bagian penutup

Membahas tentang kesimpulan hasil dan pengujian yang telah dilakukan, serta saran bagi penulis untuk memperbaiki kesalahan terhadap perancangan yang telah dilakukan.

## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Kajian Terdahulu

Penyusunan tugas akhir ini mengambil beberapa referensi penelitian sebelumnya yang berhubungan dengan tugas akhir ini. Berikut merupakan referensi penelitian yang sudah dilakukan di antaranya:

Menurut penelitian yang dilakukan Inzhagi dan kawan-kawan yang berjudul *Monitoring Sistem Pestisida Otomatis Pada Pertanian Cabai Berbasis Internet of Things (IoT)*. Penelitian ini bertujuan mengembangkan alat penyemprot hama otomatis yang dapat mengukur intensitas cahaya dan suhu berbasis IoT. Sistem otomasi penyemprotan pestisida tanaman cabai dilengkapi dengan mikrokontroler ESP8266, sensor *soil moisture* untuk mendeteksi kelembapan tanah, sensor DHT11 untuk mendeteksi suhu dan kelembapan sekitar tanaman, *relay* sebagai saklar listrik untuk menghidupkan atau mematikan pompa air dan pompa pestisida, LCD 16x2 yang berfungsi untuk menampilkan data yang sudah diperoleh sensor, selang dan *sprayer* sebagai pengaliran cairan pestisida dan air untuk melakukan penyemprotan dan penyiraman. Alat dan sistem *smart farming* yang digunakan dapat bekerja dengan baik untuk *monitoring* kelembapan tanah, suhu udara, dan kelembapan udara pada tanaman cabai. Hasil pengujian *throughput*, *packet loss*, dan *delay* mendapatkan indeks yang bagus dengan nilai rata-rata *throughput* 24,9 KBPS, *packet loss* 0,21 %, dan *delay* 66,593403 ms, nilai rata-rata dari pengujian *soil moisture* sensor adalah 39,93 % angka ini dapat diklasifikasikan sebagai standar dari kelembapan tanah ideal untuk tanaman cabai dan pada pengujian DHT11 didapatkan rata-rata suhu udara adalah 27,64 % dan kelembapan udara adalah 60,05 %, Berdasarkan rata-rata tersebut dapat dikategorikan suhu dan kelembapan udara ide (Inzhagi, Muayyadi, & Fardan, 2023).

Penelitian yang dilakukan oleh Dwiningtias yang berjudul *Prototype Sistem Monitoring Semprot Pestisida Otomatis Berbasis IoT*. Pada *monitorin* alat ini telah menggunakan *real time clock* sehingga petani dapat mengatur kapan pestisida

disemprotkan. Alat ini juga menggunakan sensor pH dan sensor TDS digunakan untuk *monitoring* cairan pestisida. Pada tangki air juga terdapat sensor *water level* sebagai sensor pendeteksi debit cairan pada bak, dengan *buzzer* yang berbunyi apabila cairan telah habis. Alat ini dapat bekerja secara otomatis dengan waktu yang telah ditentukan sehingga petani tidak mengeluarkan banyak tenaga dan bersusah payah mengelilingi petak sawah untuk menyemprotkan pestisida. Hasil dari pengujian ke tanaman telah berjalan dengan semestinya dengan penyemprotan sesuai jam yang sudah ditentukan. *Monitoring* pestisida menggunakan sensor pH didapat dengan membandingkan sensor pH digital dan sensor pH pada alat sebesar 0,34 % dan *monitoring* pestisida menggunakan sensor TDS berjalan dengan baik. Perbandingan hasil yang diperoleh dari sensor TDS digital dengan sensor TDS pada alat memiliki *error* yang tidak begitu besar yaitu 10,92 %. Aplikasi Blynk pada penelitian ini terhubung dengan alat dan bekerja secara baik serta *real time clock* (RTC) berfungsi dengan baik dan berjalan secara *real time* (Dwiningtias, 2023).

Pada penelitian yang dilakukan oleh Sulfiani dan Firmawati yang berjudul Rancang Bangun Sistem Penyemprot Tanaman Otomatis Berdasarkan Waktu Dengan *Real Time Clock* (RTC) Dan Sensor Ultrasonik Serta Notifikasi Via SMS. Pada sistem ini terdiri dari RTC DS3231 sebagai penghitung waktu, satu buah *relay* sebagai saklar, pompa air mini untuk menyemprot tanaman, sensor ultrasonik HC-SR04 untuk mendeteksi ketinggian air, GSM SIM 800L untuk mengirim SMS sebagai peringatan kepada pengguna dan LCD untuk menampilkan jarak dan waktu. Sistem ini bekerja Berdasarkan waktu sedemikian rupa sehingga pompa air hidup pada saat pukul 08.00 WIB dan pukul 17.00 WIB. Pada waktu yang sama ketinggian air dalam wadah dideteksi oleh sensor ultrasonik. Apabila jarak air dengan sensor 25 cm maka peringatan kepada pengguna berupa SMS “air hampir habis” dikirim oleh GSM. Hasil pengujian sistem jarak GSM yang digunakan dengan lokasi penerima SMS tidak berpengaruh, rata-rata waktu yang didapatkan untuk lima variasi jarak pengiriman yaitu 7,876 s, 8,646 s, 8,824 s, dan 8,676 s. Sistem penyiram tanaman otomatis ini dikontrol dengan modul Arduino Uno (Sulfiani & Firmawati, 2019).

Penelitian yang dilakukan oleh Oka dan kawan-kawan yang berjudul Model IoT Berbasis *Fuzzy Tsukamoto* Untuk Penyemprotan Pestisida Otomatis Pada Tanaman Sayur Kubis. Penelitian ini menggunakan sebuah pompa diafragma DC menyedot cairan pestisida dari tangki penampungan pestisida. Pompa terhubung dengan *relay* sebagai saklar pengendali pompa otomatis yang dikendalikan oleh modul ESP8266. Sensor suhu, cahaya dan *raindrop* merupakan variabel *input* bagi metode *Fuzzy Tsukamoto* dalam memprediksi keadaan cuaca yang tepat untuk melakukan penyemprotan. Berdasarkan hasil percobaan yang dilakukan sebanyak 5 kali terdapat selisih nilai rata-rata 0,06 % dari perbandingan *fuzzy NodeMCU* dan *fuzzy manual* (Oka, Nirmala, & Putra, 2022).

Menurut penelitian Samsinar dan Setiawan yang berjudul Perancangan Alat Implementasi *Internet of Thing* (IoT) Untuk Penyiraman Pestisida Dengan Metode Otomasi Berbasis Aplikasi Blynk Pada Tanaman. Sistem ini bisa bekerja apabila semua *input* seperti *NodeMCU*, *flow sensor*, dan aplikasi Blynk terhubung agar bisa mengirimkan sinyal notifikasi pada *handphone user* kemudian *flow sensor* membaca jumlah pestisida yang tersedia. Apabila volume pestisida yang terdapat di dalam bak penampungan cukup dan sesuai dengan jumlah yang sudah ditetapkan maka pompa bekerja menyiramkan pestisida pada tanaman. Alat ini bisa dikendalikan dari jarak jauh selama *handphone* pengguna terkoneksi dengan internet maka Wi-Fi *NodeMCU* selalu terhubung. Aplikasi Blynk juga mengirimkan sinyal notifikasi ketika alat pestisida otomatis sedang bekerja. Alat ini bekerja apabila kelembapan tanah di atas 80 %, alat ini bekerja dalam 2 waktu yaitu pagi dan sore. Alat ini beroperasi di tanggal 1 dan 15, pengujian alat ini dilakukan pada tanggal genap. Hasil pengujian alat pada tanggal genap selama 5 hari yaitu hasil respons alat sebesar 4 % dan hasil *error* dari *flow sensor* yaitu 7 %, adanya respons *error* alat tersebut disebabkan sinyal dari *provider* itu sendiri (Samsinar & Setiawan, 2023).

Menurut penelitian yang dilakukan oleh Thoriq dan kawan-kawan yang berjudul Rancang Bangun Alat Penyiram Dan Pembasmi Hama Otomatis pada Tanaman Bayam dengan *Monitoring* Berbasis *Website*. Untuk mempermudah dalam memantau keadaan tanaman bayam dibutuhkan sistem *monitoring*

menggunakan *website*, yang dapat diakses di mana pun selama 24 jam. *Website* menampilkan pembacaan data sensor kelembapan tanah dan sensor PIR yang dikirim melalui mikrokontroler NodeMCU ESP8266 ke *database* berupa persentase tanah, status tanah, dan keadaan tanaman. Berdasarkan hasil pengujian yang dilakukan bahwa data kelembapan tanah dan pergerakan hama yang dibaca oleh mikrokontroler NodeMCU ESP8266 setelah berhasil tersimpan ke *database* MySQL berhasil ditampilkan pada halaman *website* dalam bentuk tabel, hasil pengujian *monitoring* sensor pada alat penyiraman dan pembasmi hama secara otomatis pada tanaman bayam dapat ditampilkan pada *website* secara *real time* dengan pengiriman setiap 10-15 detik (Thooriq, Sujatmika, & Umami, 2023).

## 2.2 Landasan Teori

Beberapa teori yang dapat digunakan dalam menyelesaikan perancangan sistem penyemprotan pestisida otomatis berbasis *internet of things* (IoT) adalah:

### 2.2.1 Cabai

Cabai merupakan salah satu tanaman hortikultura dari famili *Solanaceae* yang memiliki nilai ekonomi tinggi. Tanama cabai ini berasal dari Amerika Serikat, cabai mengandung senyawa kimia yang dinamakan *capsaicin*. Selain itu terkandung juga berbagai senyawa yang mirip dengan *capsaicin*, yang dinamakan *capsaicinoids*. Kandungan vitamin C pada cabai cukup tinggi dapat mencegah kekurangan vitamin C seperti penyakit sariawan, meskipun memiliki banyak manfaat tetapi harus dikonsumsi secukupnya saja untuk mencegah nyeri lambung. Cabai dibutuhkan setiap keluarga, restoran, industri dan lainnya sebagainya yang berfungsi sebagai bahan pencampur makanan, bumbu dan lain-lain.

Jumlah spesies tanaman cabai yaitu sekitar 20 spesies, namun spesies tanaman cabai yang paling banyak dibudidayakan yaitu cabai rawit (*Capsicum Frutescens L.*), cabai besar (*Capsicum annum var. Grossum*), paprika (*Capsicum Longum L. Sendt*), dan cabai keriting (*Capsicum annum var. Longum*). Cabai kaya akan karbohidrat, protein, lemak, vitamin (vitamin B, vitamin C, dan vitamin E), *flavonoid*, *capsaicin*, mineral, air, dan serat. Cabai juga mengandung senyawa

antioksidan antara lain vitamin C, vitamin E, vitamin K, *fitosterol*, beta *karoten* dan beta *cryptoxanchin* (Anggraeni & Fadlil, 2013).



Gambar 2.1 Cabai Rawit  
Sumber: (Anggraeni & Fadlil, 2013)

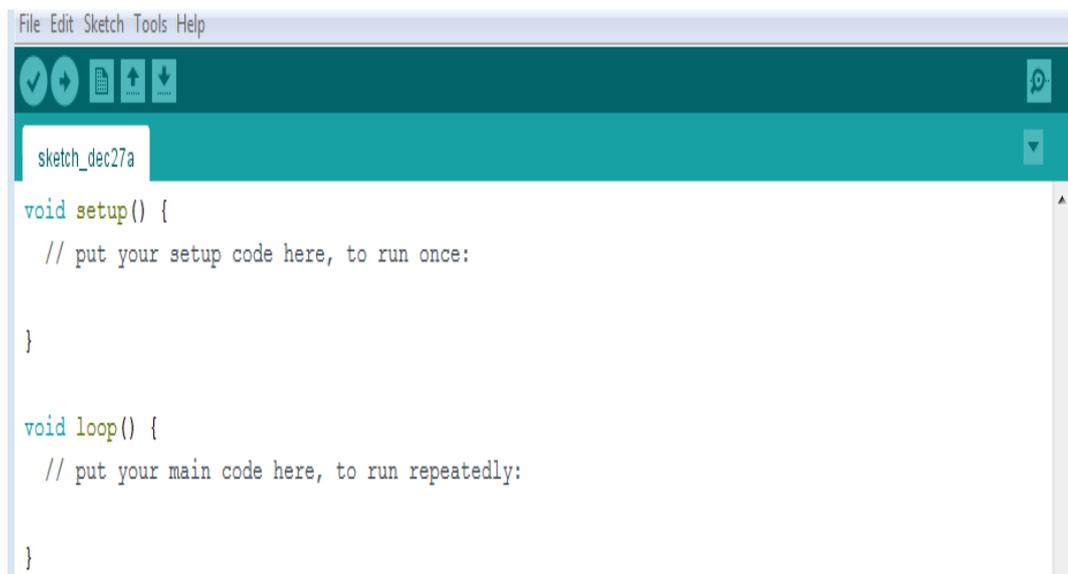
### 2.2.2 *Internet of Things*

*Internet of things* ataupun kerap disebut dengan IoT merupakan suatu gagasan di mana seluruh barang yang ada di dunia bisa berkomunikasi antara satu dan yang lain selaku bagian dari perpaduan satu kesatuan sistem yang memakai jaringan internet sebagai penghubung. Konsep IoT itu sendiri sebenarnya lumayan sederhana dengan metode kerja beracuan kepada 3 elemen pokok dalam arsitektur IoT, yaitu benda fisik yang diberikan modul IoT, alat penghubung ke internet semacam modem serta *router wireless* yang ada di rumah, serta *cloud* pusat data sebagai tempat untuk menempatkan aplikasi dan *database*. IoT bekerja dengan cara memanfaatkan suatu argumentasi pemrograman, setiap perintah argumen akan menghasilkan suatu interaksi yang terjadi antara mesin dengan mesin dan terhubung otomatis tanpa campur tangan seseorang dan tidak dibatasi oleh jarak. (Sandi & Fatma, 2023).

### 2.2.3 **Arduino IDE**

Arduino diciptakan untuk para pemula bahkan yang tidak memiliki dasar bahasa pemrograman sama sekali karena menggunakan bahasa C++ yang telah dipermudah melalui *library. Software* Arduino ini dapat di-*install* diberbagai operating sistem (OS) seperti: LINUX, Mac OS, Windows. Arduino tidak hanya sekedar sebuah alat pengembangan, tetapi kombinasi dari *hardware*, bahasa

pemrograman dan *integrated development environment* (IDE) yang canggih. IDE adalah sebuah *software* yang sangat berperan untuk menulis program, meng-*compile* menjadi kode biner dan meng-*upload* ke dalam memori mikrokontroler (Arifin, Zulita, & Hermawansyah, 2016).



Gambar 2.2 Arduino IDE  
Sumber: (Novaria, 2017)

Pada *software* Arduino IDE, terdapat semacam *message box* berwarna hitam yang berfungsi menampilkan status, seperti pesan *error*, *compile* dan *upload* program. Di bagian bawah paling kanan *software* Arduino IDE, menunjukkan *board* yang terkonfigurasi beserta *COM ports* yang digunakan, salah satu kebutuhan untuk melakukan tindakan yang sama beberapa kali dalam sebuah program (Novaria, 2017).

Berikut ini adalah bagian-bagian dari Arduino IDE:

1. *Verify*

Berfungsi untuk melakukan *checking* kode yang telah dibuat apakah sudah sesuai dengan kaidah pemrograman yang ada atau belum.

2. *Upload*

Berfungsi untuk melakukan kompilasi program atau kode yang telah dibuat menjadi bahasa yang dapat dipahami oleh Arduino.

3. *New*

Berfungsi untuk membuat *sketch* baru.

4. *Open*

Berfungsi untuk membuka *sketch* yang pernah dibuat dan membuka kembali untuk dilakukan *editing* atau sekedar *upload* ulang ke Arduino.

5. *Save*

Berfungsi untuk menyimpan *sketch* yang telah dibuat.

6. Serial monitor

Berfungsi untuk membuka serial monitor. Serial monitor di sini merupakan jendela yang menampilkan data apa saja yang dikirimkan atau dipertukarkan antara Arduino dengan *sketch* pada *port* serialnya. Serial monitor ini sangat berguna ketika ingin membuat program atau melakukan *debugging* tanpa menggunakan LCD pada Arduino. Serial monitor ini dapat digunakan untuk menampilkan nilai proses, nilai pembacaan, bahkan pesan *error*.

7. Keterangan aplikasi

Pesan-pesan yang dilakukan aplikasi muncul di sini, misal *Compiling* dan *done uploading* ketika meng-*compile* dan meng-*upload sketch* ke *board* Arduino.

8. Konsol log

Pesan-pesan yang dikerjakan aplikasi dan pesan-pesan tentang *sketch* muncul pada bagian ini. Misal, ketika aplikasi meng-*compile* atau ketika ada kesalahan pada *sketch*, maka informasi *error* dan baris diinformasikan di bagian ini.

9. Baris *sketch*

Bagian ini menunjukkan posisi baris kursor yang sedang aktif pada *sketch*.

10. Informasi *board* dan *port*

Bagian ini menginformasikan *port* yang dipakai oleh *board* Arduino. Pada Arduino bahasa pemrograman yang digunakan adalah bahasa C/C++. Program pada Arduino terbagi menjadi tiga bagian utama yaitu *structure*, *values* (berisi variabel dan konstanta) dan yang terakhir *function*.

#### 2.2.4 Blynk

Blynk adalah aplikasi *platform* IoT untuk Android dan IOS yang banyak digunakan untuk pengendalian modul Arduino maupun sejenisnya. Mempunyai media penghubung dari perangkat mikrokontroler dengan koneksi jaringan, membuat Blynk ini mudah digunakan. Memiliki 3 inti komponen utama dalam *platform* Blynk yaitu, Blynk App, Blynk Server, dan Blynk Library. Aplikasi Blynk ini memiliki beragam fitur seperti, penyimpanan data dan dapat mengendalikan objek yang terhubung dengan koneksi jaringan dari jarak yang jauh, membuat aplikasi ini memudahkan pengguna untuk memakainya (Inzhagi, Muayyadi, & Fardan, 2023).



Gambar 2.3 Aplikasi Blynk  
Sumber: (Naldi, 2022)

Aplikasi ini adalah tempat kreatif untuk membuat GUI untuk proyek yang hanya akan dilakukan dengan metode *drag and drop widget*. Blynk sangat mudah digunakan untuk mengatur semuanya dan dapat dilakukan dalam waktu kurang dari 5 menit. Blynk tidak terikat pada *tag* atau modul tertentu. *Platform* aplikasi ini dapat mengontrol semuanya dari jarak jauh, kapan saja, di mana saja (Naldi, 2022).

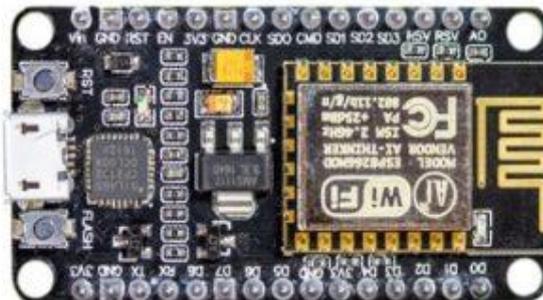
### 2.3 Komponen yang Digunakan

Beberapa komponen yang digunakan dalam menyelesaikan perancangan ini:

#### 2.3.1 NodeMCU ESP8266

ESP8266 adalah sebuah *chip* yang sudah lengkap di mana di dalamnya sudah termasuk *processor*, memori dan juga akses ke GPIO. Hal ini menyebabkan

ESP8266 dapat secara langsung menggantikan Arduino dan ditambah lagi dengan kemampuannya untuk *men-support* koneksi Wi-Fi secara langsung. *Internet of things* (IoT) semakin berkembang seiring dengan perkembangan mikrokontroler, modul yang berbasis *ethernet* maupun Wi-Fi semakin banyak dan beragam dimulai dari *wiznet*, *ethernet shield* hingga yang terbaru adalah Wi-Fi modul yang dikenal dengan ESP8266. Ada beberapa jenis ESP8266 yang dapat ditemui di pasaran, namun yang paling mudah didapatkan di Indonesia adalah tipe ESP-01, 07, dan 12 dengan fungsi yang sama perbedaannya terletak pada GPIO pin yang disediakan. Tegangan kerja ESP8266 adalah sebesar 3,3 V, sehingga untuk penggunaan mikrokontroler tambahannya dapat menggunakan *board* Arduino yang memiliki fasilitas tegangan sumber 3,3 V, akan tetapi akan lebih baik jika membuat secara terpisah *level shifter* untuk komunikasi dan sumber tegangan untuk Wi-Fi modul ini. Karena Wi-Fi modul ini dilengkapi dengan mikrokontroler dan GPIO sehingga banyak orang yang mengembangkan *firmware* untuk dapat menggunakan modul ini tanpa perangkat mikrokontroler tambahan. *Firmware* yang digunakan agar Wi-Fi modul ini dapat bekerja *standalone* (Manullang, Saragih, & Hidayat, 2021). Tabel spesifikasi NodeMCU ESP8266 dapat dilihat pada Tabel 2.1.



Gambar 2.4 NodeMCU ESP8266  
 Sumber: (Manullang, Saragih, & Hidayat, 2021)

Tabel 2.1 Spesifikasi NodeMCU ESP8266

| Spesifikasi           | Keterangan       |
|-----------------------|------------------|
| Tegangan <i>Input</i> | 3,4~ 5V          |
| GPIO                  | 17 Pin           |
| <i>Flash Memory</i>   | 16 MB            |
| RAM                   | 32 KB + 80 KB    |
| Konsumsi Daya         | 10 uA ~ 170 mA   |
| USB <i>Port</i>       | <i>Micro</i> USB |

Sumber: (Manullang, Saragih, & Hidayat, 2021)

### 2.3.2 Modul Relay

*Relay* adalah saklar (*switch*) yang dioperasikan secara listrik dan merupakan komponen *electromechanical* (elektromekanikal) yang terdiri dari 2 bagian utama yakni elektromagnet (*coil*) dan mekanikal (seperangkat kontak saklar/*switch*). *Relay* menggunakan prinsip elektromagnetik untuk menggerakkan kontak saklar sehingga dengan arus listrik yang kecil (*low power*) dapat menghantarkan listrik yang bertegangan lebih tinggi. Cara kerja *relay* adalah apabila diberi tegangan pada kaki 1 dan kaki *ground* pada kaki 2 *relay* maka secara otomatis posisi kaki *change over* (CO) pada *relay* akan berpindah dari kaki *normally close* (NC) ke kaki *normally open* (NO). *Relay* juga dapat disebut komponen elektronika berupa saklar elektronik yang digerakkan oleh arus listrik. Secara prinsip *relay* merupakan tuas saklar dengan lilitan kawat pada batang besi di dekatnya, ketika solenoid dialiri arus listrik tuas akan tertarik karena adanya gaya magnet yang terjadi pada *solenoid* sehingga kontak saklar akan menutup. Pada saat arus dihentikan gaya magnet akan hilang tuas akan kembali ke posisi semula dan kontak saklar kembali terbuka. *Relay* biasanya digunakan untuk menggerakkan arus/tegangan yang besar (misalnya peralatan listrik 4 A 220 VAC) dengan memakai arus/tegangan yang kecil (misalnya 0,1 A 12 VDC) (Santosa & Nugroho, 2021). Tabel spesifikasi NodeMCU ESP8266 dapat dilihat pada Tabel 2.2.



Gambar 2.5 Modul Relay  
Sumber: (Santosa & Nugroho, 2021)

Tabel 2.2 Spesifikasi Modul Relay

| No | Spesifikasi                | Keterangan               |
|----|----------------------------|--------------------------|
| 1  | <i>Voltage Relay</i>       | 5 V                      |
| 2  | <i>Max Voltage/Current</i> | 10 A 125VAC/ 10 A 28 VDC |
| 3  | Dimensi                    | 7 x 5,3 x 1,8 cm         |

Sumber: (Santosa & Nugroho, 2021)

### 2.3.3 RTC

*Real time clock* (RTC) adalah jam elektronik berupa *chip* yang dapat menghitung waktu (mulai detik hingga tahun) dengan akurat dan menjaga atau menyimpan data waktu tersebut secara *real time*. Jam tersebut bekerja *real time*, *output* datanya langsung disimpan atau dikirim ke *device* lain melalui sistem antarmuka setelah proses hitung waktu dilakukan. *Chip* RTC sering dijumpai pada *motherboard* PC (biasanya terletak dekat *chip* BIOS). Semua komputer menggunakan RTC karena berfungsi menyimpan informasi jam terkini dari komputer yang bersangkutan. RTC dilengkapi dengan baterai sebagai pemasok daya pada *chip*, sehingga jam akan tetap *up-to-date* walaupun komputer dimatikan. RTC dinilai cukup akurat sebagai perwaktu (*timer*) karena menggunakan osilator kristal. Banyak contoh *chip* RTC yang ada di pasaran seperti DS12C887, DS1307, DS1302, DS3234 (Iqbar & Kartika, 2020).



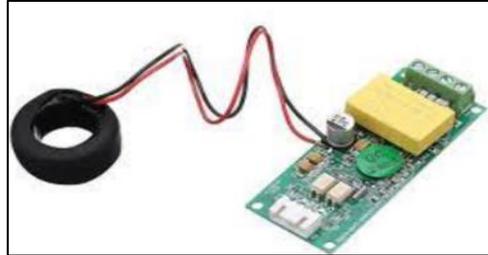
Gambar 2.6 *Real Time Clock* (RTC)  
Sumber: (Iqbar & Kartika, 2020)

### 2.3.4 Sensor PZEM-004T

PZEM-004T merupakan sensor yang memiliki fungsi ganda. Sensor ini dapat membaca nilai tegangan RMS, arus RMS dan daya aktif yang dapat diintegrasikan melalui NodeMCU ESP32 atau *platform open source* lainnya. Modul PZEM-004T sendiri dibundel dengan kumparan trafo arus diameter 3 mm yang dapat dipakai untuk mengukur arus maksimal 100 A (Pratikto, Hendrawan, & Agustini, 2022).

Modul ini tersedia tanpa fungsi tampilan, data dibaca melalui *interface* TTL. *Interface* TTL dari modul ini adalah *interface* pasif, membutuhkan catu daya eksternal 5 V, yang berarti ketika berkomunikasi, keempat *port* harus terhubung (5

V, RX, TX, GND) jika tidak modul ini tidak dapat berkomunikasi (Rini, 2022).  
 Sensor PZEM-004T dapat dilihat pada Gambar 2.5.



Gambar 2.7 Sensor PZEM-004T  
 Sumber: (Pratikto, Hendrawan, & Agustini, 2022)

Tabel 2.3 Spesifikasi Sensor PZEM-004T

| No | Spesifikasi  | Keterangan     |
|----|--------------|----------------|
| 1  | Tegangan     | 80 ~ 260 V     |
| 2  | Arus         | 0 ~ 100 A      |
| 3  | Daya aktif   | 0 ~ 23 kW      |
| 4  | Power faktor | 0,00 ~ 1,00    |
| 5  | Frekuensi    | 0,00 ~ 1,00 Hz |
| 6  | Interface    | UART TTL       |

Sumber: (Rini, 2022)

### 2.3.5 Kabel Jumper

Dalam merancang sebuah desain peralatan elektronik tentunya sangat dibutuhkan sebuah kabel untuk menghubungkan komponen elektronik yang satu dengan komponen elektronik yang lainnya. Maka dari itu kabel *jumper breadboard male to male* merupakan salah satu jenis kabel *jumper* untuk *breadboard* yang dapat digunakan untuk menghubungkan komponen-komponen elektronik yang satu dengan yang lainnya. Kabel *jumper* ini dapat digunakan untuk menyambungkan komponen elektronik yang satu dengan yang lainnya pada saat membuat *prototype* dengan menggunakan *breadboard* selain itu kabel *jumper* ini berfungsi untuk menghubungkan beberapa *breadboard*, menghubungkan antar titik pada PCB *single slide* dan juga dapat digunakan untuk menghubungkan jalur rangkaian yang terputus dengan cara men-*jumper*-nya (Iqbar & Kartika, 2020).



Gambar 2.8 Kabel Jumper  
 Sumber: (Iqbar & Kartika, 2020)

### 2.3.6 Power Supply

*Power supply* adalah salah satu *hardware* dalam perangkat komputer yang berperan untuk memberikan suplai daya. Biasanya komponen *power supply* ini bisa ditemukan pada *casing* komputer dan berbentuk persegi. Pada dasarnya *power supply* membutuhkan sumber listrik yang kemudian diubah menjadi energi yang menggerakkan perangkat elektronik. Sistem kerjanya cukup sederhana yakni dengan mengubah daya 220 V ke dalam bentuk aliran dengan daya yang sesuai kebutuhan komponen tersebut. Sesuai dengan pengertian *power supply* pada komputer, maka fungsi utamanya adalah untuk mengubah arus AC menjadi arus DC yang kemudian diubah menjadi daya atau energi yang dibutuhkan komponen-komponen pada komputer seperti *motherboard*, *CD Room*, *Hardisk*, dan komponen lainnya (Santosa & Nugroho, 2021).



Gambar 2.9 *Power Supply*  
Sumber: (Santosa & Nugroho, 2021)

### 2.3.7 Sprayer

*Sprayer* adalah salah satu alat pertanian yang digunakan untuk membantu pekerjaan petani, sebagai sarana membantu memberi tanaman pestisida atau pupuk daun alami untuk memberantas hama penyakit yang ada pada pertanian. *Sprayer* sendiri memiliki cara kerja sebagai alat pembantu untuk mengontrol hama tanaman dengan mencampurkan pestisida yang digunakan dengan air ataupun bahan aktif lainnya, dari bahan aktif pestisida yang telah dilarutkan dengan air digunakan untuk menyemprot tanaman pertanian (Inzhagi, Muayyadi, & Fardan, 2023).



Gambar 2.10 *Sprayer*  
Sumber: (Inzhagi, Muayyadi, & Fardan, 2023)

### 2.3.8 Pompa Air

Pompa air adalah suatu alat yang digunakan untuk memindahkan zat cair/air dari suatu tempat ke tempat lain yang biasanya menggunakan energi listrik sebagai tenaga untuk mendorong air dengan cara menaikkan tekanan zat cair untuk mengatasi hambatan aliran. Prinsip pengoperasian pompa air mengubah energi motor menjadi energi aliran fluida (Naldi, 2022).



Gambar 2.11 Pompa Air  
Sumber: (Naldi, 2022)

Tabel 2.4 Spesifikasi Pompa Air

| No | Spesifikasi         | Keterangan        |
|----|---------------------|-------------------|
| 1  | Output              | 200 W             |
| 2  | Input               | 0,4 Kw            |
| 3  | Arus                | 2,2 A             |
| 4  | Daya hisap maksimal | 9 meter           |
| 5  | Pipa hisap          | 1 inci            |
| 6  | Pipa dorong         | 1 inci            |
| 7  | Kapasitas           | 32 liter/21 menit |

Sumber: (Naldi, 2022)

### 2.3.9 Push Button

*Push button switch* (saklar tombol tekan) adalah perangkat/saklar sederhana yang berfungsi untuk menghubungkan atau memutuskan aliran arus listrik dengan sistem kerja tekan *unlock* (tidak mengunci). Sistem kerja *unlock* di sini berarti saklar akan bekerja sebagai *device* penghubung atau pemutus aliran arus listrik saat tombol ditekan, dan saat tombol tidak ditekan (dilepas), maka saklar akan kembali pada kondisi normal sebagai *device* penghubung atau pemutus. *Push button switch* hanya memiliki 2 kondisi, yaitu *on* dan *off* (1 dan 0). Istilah *on* dan *off* ini menjadi sangat penting karena semua perangkat listrik yang memerlukan sumber energi listrik pasti membutuhkan kondisi *on* dan *off*. Karena sistem kerjanya yang *unlock* dan langsung berhubungan dengan operator, *push button switch* menjadi *device* paling utama yang biasa digunakan untuk memulai dan mengakhiri kerja mesin di

industri. Secanggih apa pun sebuah mesin bisa dipastikan sistem kerjanya tidak terlepas dari keberadaan sebuah saklar seperti *push button switch* atau perangkat lain yang sejenis yang bekerja mengatur pengkondisian *on* dan *off* (Munandar, Veronika, Abdullah, & Sahputra, 2023).



Gambar 2.12 *Push Button*

Sumber: (Munandar, Veronika, Abdullah, & Sahputra, 2023)

## **BAB III**

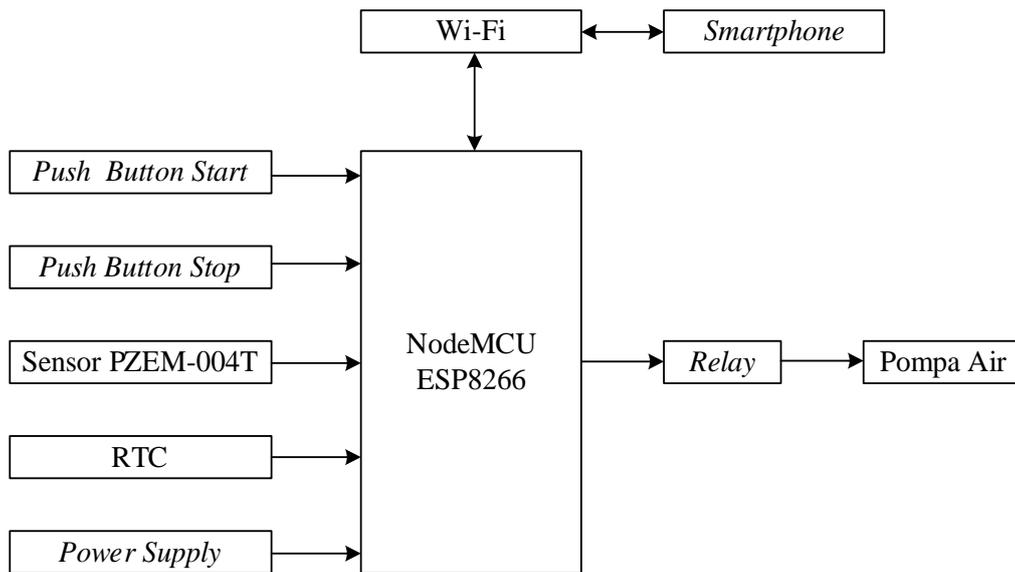
### **METODOLOGI PENELITIAN**

#### **3.1 Tinjauan Umum**

Metode yang digunakan pada penelitian yang akan dilakukan pada Implementasi Sistem Penyemprotan Pestisida Otomatis Berbasis *Internet of Things* (IoT) yaitu dirangkai dengan menggunakan mikrokontroler NodeMCU ESP8266 dan aplikasi Blynk sebagai pengaturan utama. Dalam pemrograman ini dibutuhkan *software* Arduino IDE yang digunakan untuk memasukkan program agar terhubung pada NodeMCU ESP8266. Selain itu komponen yang digunakan adalah sensor PZEM berfungsi untuk mengetahui pemakaian arus dan tegangan yang digunakan. Kemudian pompa air dihubungkan langsung ke sumber listrik dan pestisida secara otomatis disemprotkan ke tanaman dengan kontrol waktu yang ditentukan oleh RTC yang sudah diperintahkan dan terhubung ke ESP8266, *relay* digunakan untuk pengontrolan pompa air. Selain itu sistem alat ini dapat bekerja dengan kontrol manual dan otomatis. Blynk mengirim data perintah yang dikendalikan menuju NodeMCU ESP8266 yang telah dikoneksikan melalui token dari aplikasi Blynk untuk sistem kendali dan informasi dari jarak jauh. Sistem ini adalah sistem yang mudah karena sudah bisa diatur hanya menggunakan perangkat yang terhubung ke Wi-Fi.

#### **3.2 Blok Diagram**

Blok diagram merupakan gambaran urutan keseluruhan kerja secara umum dari suatu sistem. Tujuannya yaitu untuk memudahkan dalam melihat proses yang berlangsung dalam sistem yang dibuat. Blok diagram memiliki arti yang khusus dengan memberikan keterangan di dalamnya. Untuk setiap blok dihubungkan dengan satu garis yang menunjukkan arah kerja dari setiap blok yang bersangkutan. Blok diagram dari Implementasi Sistem Penyemprotan Pestisida Otomatis Berbasis *Internet of Things* (IoT) dapat dilihat pada Gambar 3.1.



Gambar 3.1 Blok Diagram  
Sumber: (Dokumentasi, 2024)

Fungsi komponen yang ada pada Gambar 3.1 blok diagram perancangan dapat dijelaskan sebagai berikut:

1. NodeMCU ESP8266  
Sebagai mikrokontroler untuk mengendalikan RTC dan sensor PZEM.
2. Pompa air  
Sebagai pengalir cairan pestisida ke pipa.
3. RTC  
Sebagai pengaturan waktu kerja alat.
4. Sensor PZEM-004T  
Sebagai sensor yang mendeteksi tegangan dan arus.
5. *Power supply*  
Memberikan daya kepada mikrokontroler NodeMCU ESP8266 dan memberikan *supply* pada pompa air.
6. Jaringan internet.  
Sebagai penghubung antara mikrokontroler NodeMCU ESP8266 dan Blynk.
7. Blynk  
Aplikasi yang digunakan untuk melakukan kontrol dan *monitoring* di *smartphone*.

8. *Smartphone*

Operator melakukan kontrol dan *monitoring*.

9. *Relay*

Sebagai pengontrol aliran listrik ke pompa air.

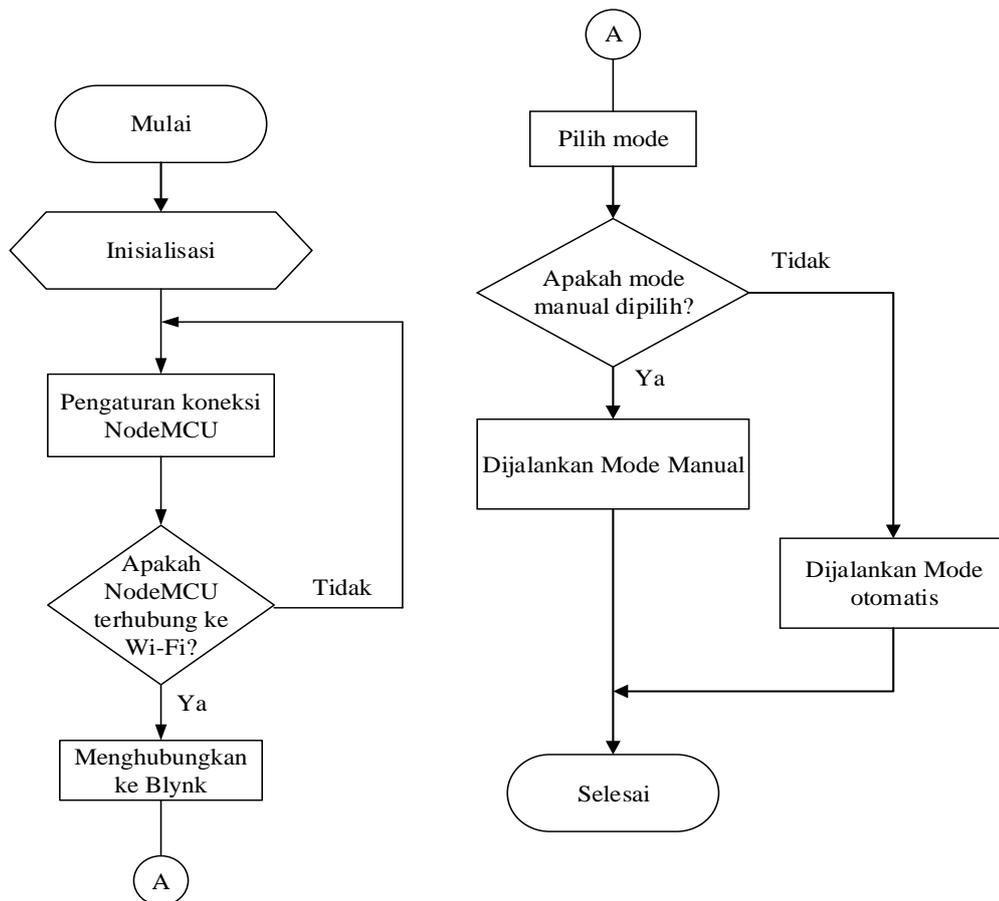
10. *Push button*

*Push button on* dan *push button off* digunakan sebagai kontrol manual untuk menghidupkan dan mematikan pompa air.

3.3 *Flowchart*

Alur kerja sistem mudah dipahami jika dibuat dalam bentuk diagram alir.

*Flowchart* sistem kerja alat ini ditunjukkan pada Gambar 3.2.



Gambar 3.2 *Flowchart* Sistem Kerja Alat Keseluruhan  
Sumber: (Dokumentasi, 2024)

Gambar 3.2 merupakan sistem kerja alat secara keseluruhan yang meliputi dua pemilihan mode yaitu sistem kerja alat secara otomatis dan secara manual.

Berdasarkan Gambar 3.2 dapat diperjelas uraian *flowchart* sistem kerja alat sebagai berikut:

1. Mulai

Mulai merupakan tahapan awal permulaan dari sistem kerja alat, dalam tahap ini semua peralatan sudah siap dibuat untuk tahap berikutnya yaitu pengujian alat.

2. Inisialisasi

Inisialisasi adalah bagian untuk pengecekan semua perangkat, sehingga dapat dipastikan bahwa alat yang digunakan semua dalam keadaan baik dan bisa dijalankan sesuai yang diharapkan.

3. Pengaturan koneksi agar NodeMCU terhubung ke Wi-Fi.

4. Jika NodeMCU terhubung ke Wi-Fi maka dilakukan proses menghubungkan ke Blynk. Jika belum terhubung ke Wi-Fi maka dilakukan pengaturan kembali.

5. Jika suda terhubung ke Wi-Fi maka lakukan pemilihan mode.

6. Lakukan pemilihan mode sesuai keinginan pengguna terdapat dua pemilihan mode yaitu mode manual dan mode otomatis.

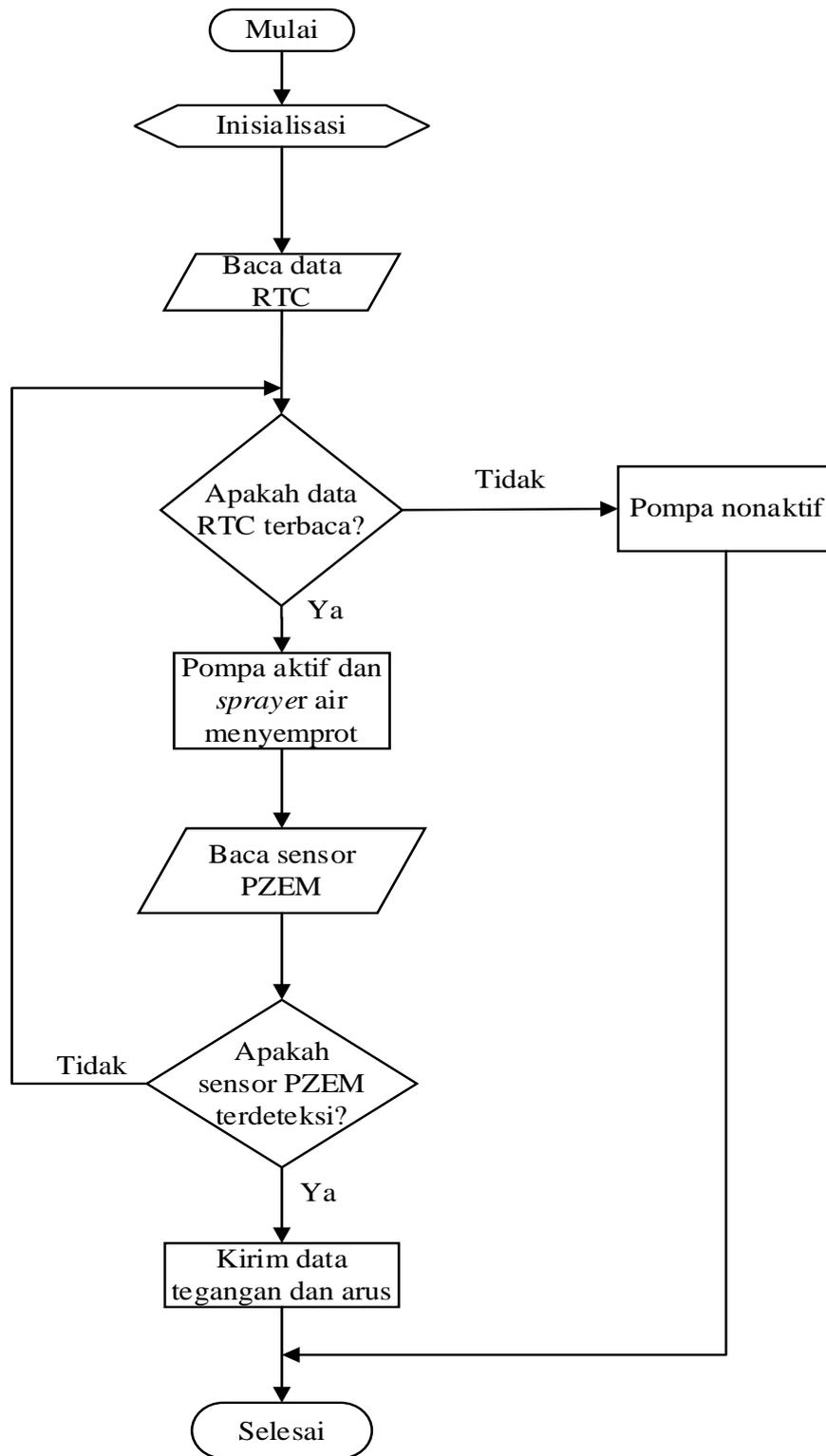
7. Jika pengguna memilih mode manual maka sistem kerja alat dijalankan secara manual.

8. Jika pengguna memilih mode otomatis maka sistem kerja alat dijalankan secara otomatis.

9. Selesai

Pada tahap ini merupakan tahap akhir yaitu mengambil kesimpulan dari hasil pengujian alat yang telah dilakukan. Kesimpulan berisi penjelasan persentase keberhasilan sistem yang dibuat dengan metode awal, penjelasan metode apa yang sesuai untuk pemecahan masalah-masalah yang timbul saat melakukan pengujian alat.

Gambar 3.3 merupakan *flowchart* sistem kerja secara otomatis dengan penggunaan RTC (*real time clock*) sebagai pengaturan waktu yang sudah pengguna atur sesuai jadwal penyemprotan pestisida serta menggunakan sensor PZEM-004T untuk mendeteksi tegangan dan arus yang dipakai pada saat penyemprotan.



Gambar 3.3 Flowchart Sistem Kerja Alat Otomatis  
 Sumber: (Dokumentasi, 2024)

Berdasarkan Gambar 3.3 dapat diperjelas uraian *flowchart* sistem kerja alat sebagai berikut:

1. Mulai

Mulai merupakan tahapan awal permulaan dari sistem kerja alat, dalam tahap ini semua peralatan sudah siap dibuat untuk tahap berikutnya yaitu pengujian alat.

2. Inisialisasi

Inisialisasi adalah bagian untuk pengecekan semua perangkat, sehingga dapat dipastikan bahwa alat yang digunakan semua dalam keadaan baik dan bisa dijalankan sesuai yang diharapkan.

3. Baca data RTC

Sistem membaca data RTC yang telah diperintahkan NodeMCU ESP8266, data RTC ini dibuat dengan jadwal dan jam yang sudah ditentukan.

4. Jika data RTC terbaca maka pompa air aktif dan menyemprotkan pestisida.

5. Jika data RTC tidak terbaca maka pompa air nonaktif dan tidak menyemprotkan pestisida.

6. Baca sensor PZEM-004T

Selanjutnya baca sensor PZEM-004T yang telah diperintahkan NodeMCU ESP8266.

7. Deteksi tegangan dan arus

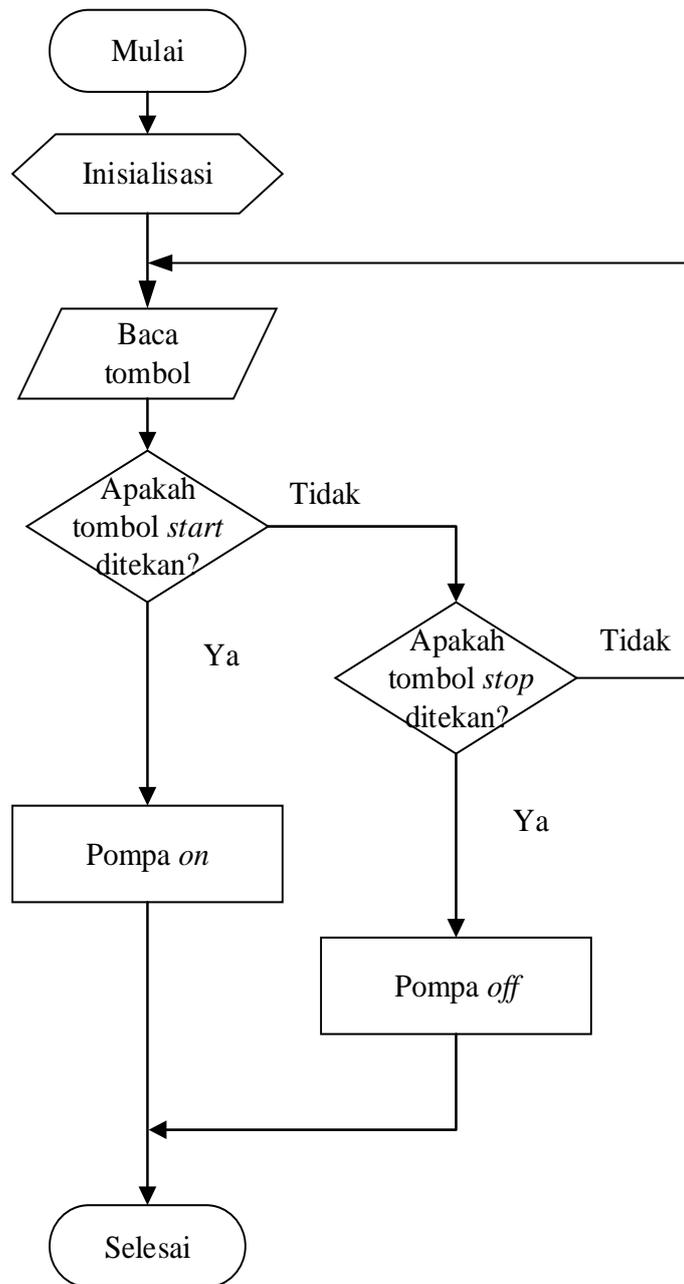
Jika sensor PZEM-004T terbaca maka dilakukan proses deteksi tegangan dan arus pompa air.

8. Kirim tegangan dan arus

Selanjutnya data pembacaan sensor PZEM-004T dikirim dan dilakukan *monitoring* tegangan dan arus melalui aplikasi Blynk.

9. Selesai

Pada tahap ini merupakan tahap akhir yaitu mengambil kesimpulan dari hasil pengujian alat yang telah dilakukan. Kesimpulan berisi penjelasan persentase keberhasilan sistem yang dibuat dengan metode awal, penjelasan metode apa yang sesuai untuk pemecahan masalah-masalah yang timbul saat melakukan pengujian alat.



Gambar 3.4 *Flowchart* Sistem Kerja Secara Manual  
 Sumber: (Dokumentasi, 2024)

Berdasarkan Gambar 3.4 dapat diperjelas uraian *flowchart* sistem kerja alat sebagai berikut:

1. Mulai

Mulai merupakan tahapan awal permulaan dari sistem kerja alat, dalam tahap ini semua peralatan sudah siap dibuat untuk tahap berikutnya yaitu pengujian alat.

## 2. Inisialisasi

Inisialisasi adalah bagian untuk pengecekan semua perangkat, sehingga dapat dipastikan bahwa alat yang digunakan semua dalam keadaan baik dan bisa dijalankan sesuai yang diharapkan.

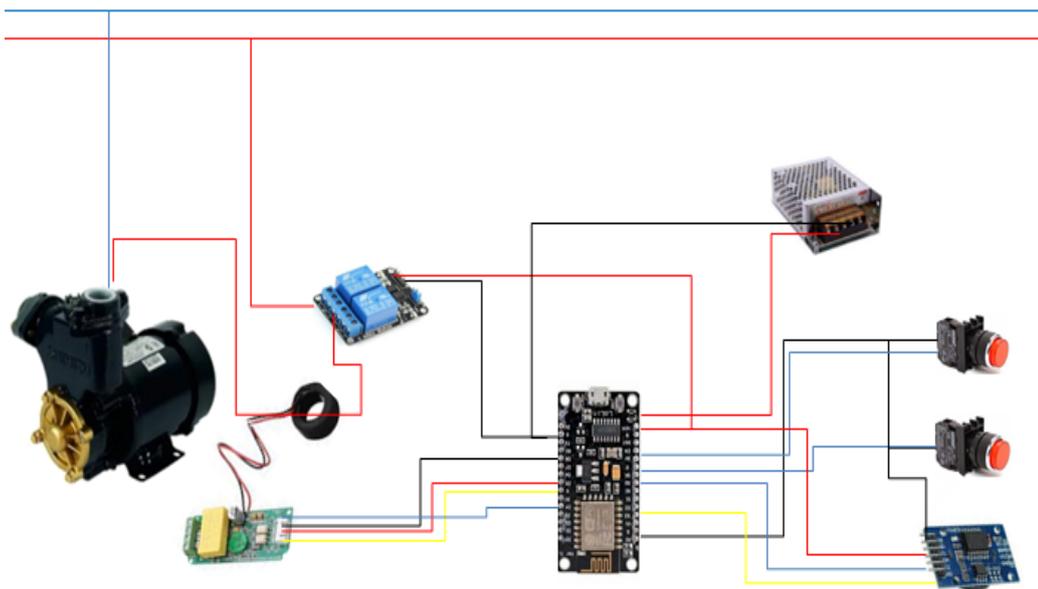
## 3. Selanjutnya dilakukan pembacaan tombol, jika tombol *start* maka pompa on dan jika tombol *stop* yang terbaca maka pompa air *off*.

## 4. Selesai

Pada tahap ini merupakan tahap akhir yaitu mengambil kesimpulan dari hasil pengujian alat yang telah dilakukan. Kesimpulan berisi penjelasan persentase keberhasilan sistem yang dibuat dengan metode awal, penjelasan metode apa yang sesuai untuk pemecahan masalah-masalah yang timbul saat melakukan pengujian alat.

### 3.4 Rancangan *Hardware*

Dalam hal perancangan *hardware* dimulai dengan merancang blok diagram dan prinsip kerja alat, dilanjutkan dengan merancang rangkaian alat dengan menggabungkan beberapa perangkat menjadi suatu sistem. Rancangan ini dapat dilihat pada Gambar 3.5.



Gambar 3.5 Rancangan *Hardware*  
Sumber: (Dokumentasi, 2024)

### 3.5 Rancangan Software

Rancangan *software* yang dimaksud adalah dengan memprogram NodeMCU ESP8266 sebagai otak dari komponen yang digunakan dalam *prototype* seperti *relay*, sensor PZEM-004T, dan *real time clock* (RTC). Pemrograman NodeMCU ESP8266 diprogram menggunakan bahasa C.

#### 3.5.1 Rancangan Program untuk RTC

Program ini digunakan sebagai pengaturan waktu dalam perancangan alat yang dibuat, dengan menggunakan bahasa program seperti pada Gambar 3.6.

```
#include <RTClib.h>
RTC_DS1307 rtc;
int jam,menit,detik,tanggal,bulan,tahun,hari;
char temp[60];
char nama_hari[7][7] = {
  "Minggu",
  "Senin",
  "Selasa",
  "Rabu",
  "Kamis",
  "Jumat",
  "Sabtu",
};

void setup() {
  Serial.begin(115200);
  delay(1000);
  rtc.begin();
  rtc.adjust(DateTime(F(__DATE__), F(__TIME__))); //ngambil tgl waktu dri laptop
}

void loop() {
  DateTime now = rtc.now();
  jam = now.hour();
  menit = now.minute();
  detik = now.second();
  tanggal = now.day();
  bulan = now.month();
  tahun = now.year();
  hari = now.dayOfTheWeek();
  sprintf(temp,"JAM %02d:%02d:%02d WIB",jam,menit,detik);
  Serial.print(temp);

  sprintf(temp,"%s,%02d-%02d-%d",nama_hari[hari],tanggal,bulan,tahun);
  Serial.print(temp);
}
```

Gambar 3.6 Rancangan Program Untuk RTC  
Sumber: (Dokumentasi, 2024)

#### 3.5.2 Rancangan Program Untuk Sensor PZEM-004T

Program ini digunakan sebagai pendeteksi tegangan dan arus dalam perancangan alat yang dibuat, dengan menggunakan bahasa program seperti pada Gambar 3.7.

```

#include <PZEM004Tv30.h>

PZEM004Tv30 pzemR(D3, D0); // (RX,TX) connect to D3_TX,D0_RX of PZEM
PZEM004Tv30 pzemS(D4, D5); // (RX,TX) connect to D4_TX,D5_RX of PZEM
PZEM004Tv30 pzemT(D6, D7); // (RX,TX) connect to D6_TX,D7_RX of PZEM

float VR = 0;
float IR = 0;

void setup(){
  Serial.begin(115200);
  delay(1000);
}

void loop(){
  float voltageR = pzemR.voltage();
  if(isnan(voltageR)){
    Serial.println("Error reading voltage"); VR=0; voltageR=0;
    Serial.print("VoltageR: ");    Serial.print(voltageR);    Serial.println("V");
  }
  else {
    VR=voltageR;Serial.print("VoltageR: ");    Serial.print(voltageR);    Serial.println("V");
  }

  float currentR = pzemR.current();
  if(isnan(currentR)){
    Serial.println("Error reading current"); IR=0; currentR=0;
    Serial.print("CurrentR: ");    Serial.print(currentR);    Serial.println("A");
  }
  else {
    IR=currentR;Serial.print("CurrentR: ");    Serial.print(currentR);    Serial.println("A");
  }
}

```

Gambar 3.7 Program Rancangan Untuk Sensor PZEM-004T  
 Sumber: (Dokumentasi, 2024)

### 3.5.3 Rancangan Program Untuk Relay dan Push Button

Gambar 3.8 merupakan Bahasa pemrograman pada *software* Arduino IDE untuk *relay* dan *push button*.

```

#define relay D4
#define start D5
#define stop D6
void setup() {
  pinMode(relay,OUTPUT);
  pinMode(start,INPUT_PULLUP);
  pinMode(stop,INPUT_PULLUP);
  digitalWrite(relay,HIGH);
  Serial.begin(9600);
}

void loop() {
  if(digitalRead(start)==LOW){
    digitalWrite(relay,LOW);
  } else if(digitalRead(stop)==LOW){
    digitalWrite(relay,HIGH);
  }
}

```

Gambar 3.8 Rancangan Program Untuk Relay Dan Push Button  
 Sumber: (Dokumentasi, 2024)

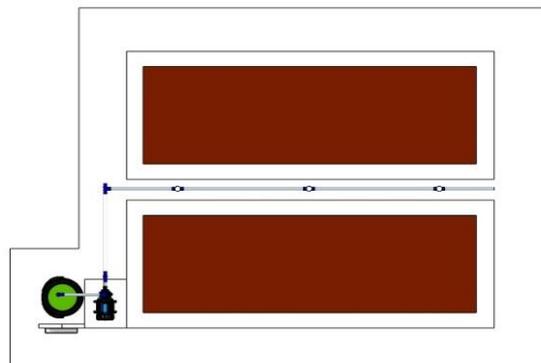
### 3.6 Rancangan *Prototype* Alat Keseluruhan

Rancangan *prototype* merupakan proses pembuatan model sederhana menggunakan perangkat lunak yang bertujuan untuk memiliki gambaran dasar

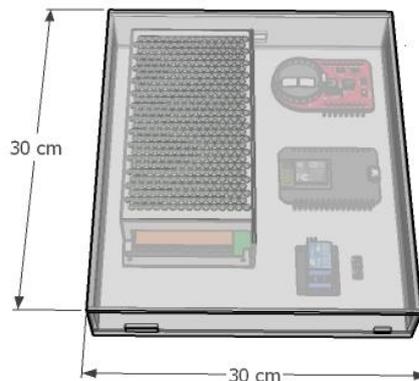
tentang sistem yang dibuat, sehingga dapat dengan mudah memodelkan gambaran alat yang dibuat. Perancangan *prototype* yang dibuat dalam penelitian ini dapat dilihat pada Gambar 3.9 sampai Gambar 3.11.



Gambar 3.9 Rancangan Tampak Depan  
Sumber: (Dokumentasi, 2024)



Gambar 3.10 Rancangan Tampak Atas  
Sumber: (Dokumentasi, 2024)



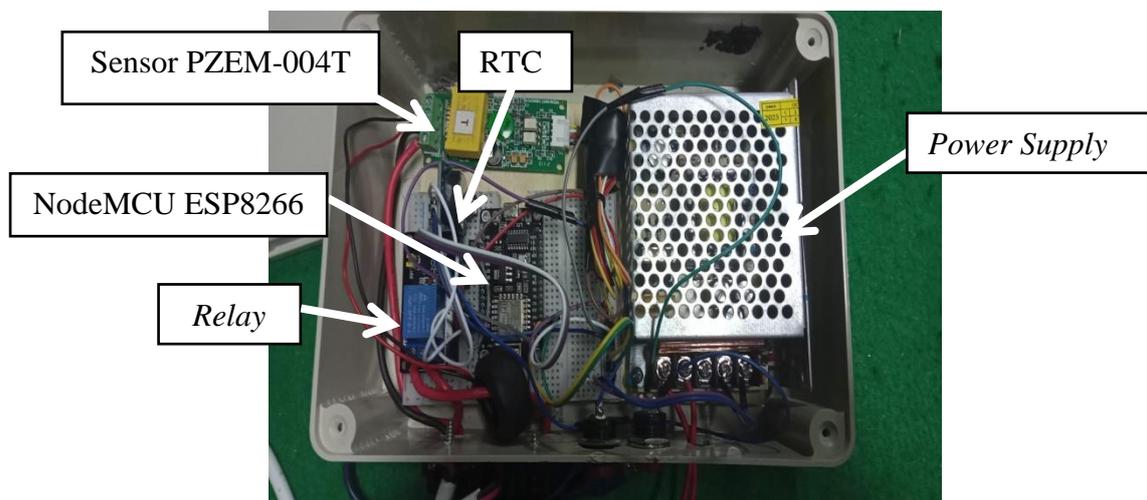
Gambar 3.11 Rancangan *Prototype*  
Sumber: (Dokumentasi, 2024)

## BAB IV

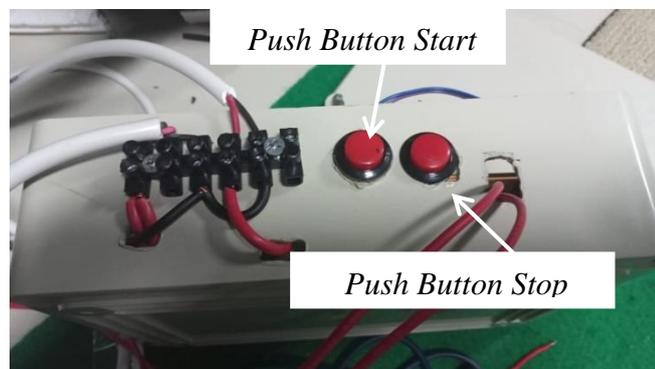
### HASIL PENELITIAN DAN ANALISA

#### 4.1. Hasil Perancangan Alat

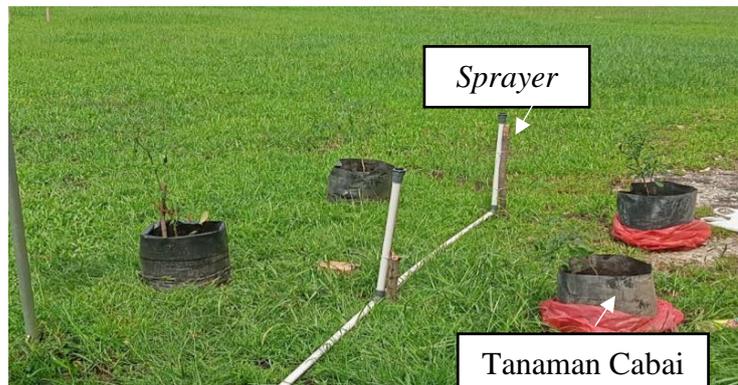
Pada tahapan perancangan sistem penyemprotan pestisida otomatis ini perlu diperhatikan pada sistem kerja dan rangkaian yang digunakan. Pemasangan komponen yang baik dan benar sangat diperlukan agar tidak terjadi kerusakan yang tidak diharapkan, di antara komponen yang digunakan yaitu: NodeMCU ESP8266, *relay*, *push button*, sensor PZEM-004T, *real time clock* (RTC), *power supply* 5 V. Bentuk fisik alat perancangan dapat dilihat pada Gambar 4.1 sampai Gambar 4.4.



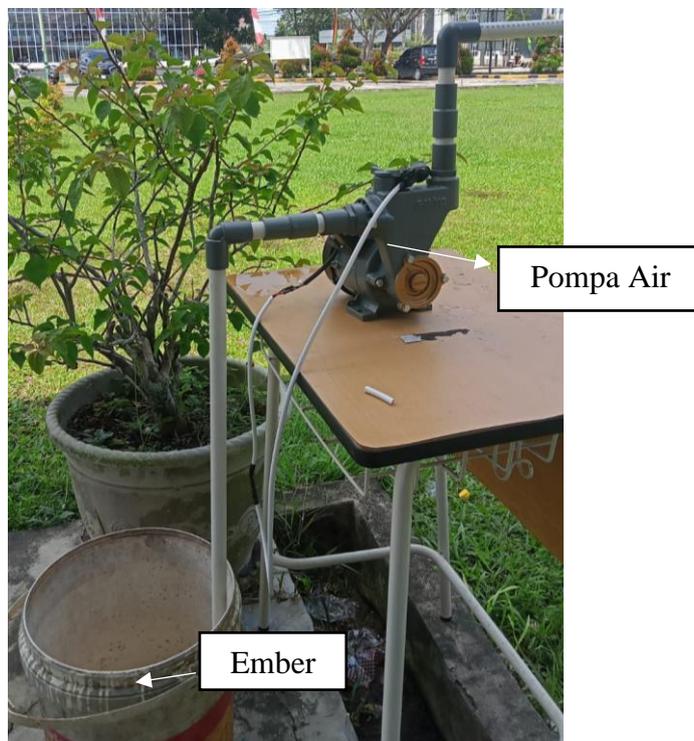
Gambar 4.1 Tampak Dalam  
Sumber: (Dokumentasi, 2024)



Gambar 4.2 Tampak Samping  
Sumber: (Dokumentasi, 2024)



Gambar 4.3 *Sprayer* dan Tanaman Cabai  
 Sumber: (Dokumentasi, 2024)



Gambar 4.4 Pompa Air Dan Ember  
 Sumber: (Dokumentasi, 2024)

Pada Gambar 4.1 merupakan tampilan alat utama yang digunakan, terdapat NodeMCU ESP8266, *relay*, *push button*, sensor PZEM-004T, *real time clock* (RTC), *power supply* 5 V. *Push button* digunakan sebagai mengaktifkan dan menonaktifkan pompa air, dan pada Gambar 4.3 dan Gambar 4.4 merupakan rancangan keseluruhan alat yang digunakan dalam penelitian ini, ember digunakan sebagai tempat pestisida dan pompa air digunakan untuk menarik air dari ember dan dialirkan melalui pipa paralon lalu disemprotkan dengan *sprayer* sehingga pestisida tersemprot secara merata.

## 4.2. Hasil Pengujian Alat

Dari hasil pengujian dan perancangan alat maka selanjutnya adalah pengujian dan menganalisa data dari alat yang telah dibuat. Hal ini bertujuan untuk mengetahui apakah pengoperasian penyiraman tanaman dengan android berbasis IoT sudah sesuai dengan perancangan alat, adapun pengujian yang dilakukan baik pengujian *hardware* maupun *software*. Pengujian yang dilakukan pada tiap-tiap komponen dari setiap sistem sehingga diketahui kinerja dari masing-masing komponen. Pengujian komponen yang dilakukan yaitu:

1. Pengujian RTC
2. Pengujian Modul *Relay*
3. Pengujian *Power Supply*
4. Pengujian Sensor PZEM-004T
5. Pengujian *Push Button*
6. Pengujian Aplikasi Blynk

### 4.2.1. Pengujian RTC DS3231

Pengujian RTC diperlukan untuk memastikan bahwa pembacaan dari modul RTC ini sesuai dengan waktu yang pengguna atur. Pengujian waktu RTC ini membandingkan waktu pada RTC dengan membandingkan waktu sekarang yang ada pada *handphone*. Pengujian RTC dapat dilihat pada Tabel 4.1.

Tabel 4.1 Pengujian RTC DS3231

| Percobaan ke- | Waktu Sekarang | Waktu pada RTC | Selisih |
|---------------|----------------|----------------|---------|
| 1             | 10:10:30       | 10:10:33       | 3 detik |
| 2             | 11:30:48       | 11:30:51       | 3 detik |
| 3             | 12:56:10       | 12:56:13       | 3 detik |
| 4             | 13:01:28       | 13:01:31       | 3 detik |
| 5             | 14:28:10       | 14:28:13       | 3 detik |

Sumber: (Data Olahan, 2024)

Perhitungan selisih pada RTC DS3231 menggunakan rumus sebagai berikut:

Selisih = Waktu pada RTC – Waktu sekarang

1. Percobaan ke-1 = Waktu pada RTC – Waktu sekarang  
= 10: 10: 33–10: 10: 30  
= 3 detik
2. Percobaan ke – 2 = Waktu pada RTC – Waktu sekarang

$$= 11:30:51 - 12:56:13$$

$$= 3 \text{ detik}$$

3. Percobaan ke – 3 = Waktu pada RTC – Waktu sekarang

$$= 12:56:10 - 12:56:13$$

$$= 3 \text{ detik}$$

4. Percobaan ke – 4 = Waktu pada RTC – Waktu sekarang

$$= 10:10:33 - 10:10:30$$

$$= 3 \text{ detik}$$

5. Percobaan ke – 5 = Waktu pada RTC – Waktu sekarang

$$= 14:28:13 - 14:28:13$$

$$= 3 \text{ detik}$$

Berdasarkan data pada Tabel 4.1 menunjukkan hasil pengujian dari modul RTC DS3231, waktu yang dibandingkan dengan waktu yang ada pada modul RTC DS3231 dengan perbandingan waktu pada *handphone* dengan selisih waktu selama 3 detik.

#### 4.2.2. Pengujian Modul Relay

*Relay* adalah salah satu saklar otomatis yang memanfaatkan elektromagnet untuk mengaktifkan maupun menonaktifkan suatu komponen. Pengujian ini dilakukan untuk mengetahui kondisi pada *relay* saat aktif maupun nonaktif sehingga modul ini dapat difungsikan dengan semestinya sebagai pengaturan untuk menonaktifkan serta mengaktifkan pompa air yang digunakan dalam penelitian ini. Pengujian modul *relay* dapat dilihat pada Tabel 4.2.

Tabel 4.2 Pengujian Modul Relay

| Relay  | IN | Kondisi Anak (NO) | Tegangan keluaran Anak Kontak (NO) |
|--------|----|-------------------|------------------------------------|
| Pin D4 | 0  | Terputus          | 0                                  |
|        | 1  | Terhubung         | 230 V                              |

Sumber: (Data Olahan, 2024)

Dari pengujian *relay* saat kondisi anak kontak NO terputus maka tegangan keluaran anak kontak *normally open* (NO) 0 V dan pada saat kondisi anak kontak NO terhubung maka tegangan keluaran anak kontak *normally open* (NO) yang terhubung dengan pompa air sebesar 230 V AC.

### 4.2.3. Pengujian *Push Button*

*Push button* digunakan untuk mengaktifkan pompa air 220 VAC. Pengujian ini bertujuan untuk mengetahui berapa tegangan pada *push button* 1 dan *push button* 2 pada saat ditekan dan tidak ditekan. Hasil pengujian yang telah dilakukan dapat dilihat pada Tabel 4.3.

Tabel 4.3 Pengujian Push Button

| No | Push Button | Kondisi       | Tegangan (V) |
|----|-------------|---------------|--------------|
| 1  | PB 1        | Ditekan       | 0            |
|    |             | Tidak Ditekan | 3,14         |
| 2  | PB 2        | Ditekan       | 0            |
|    |             | Tidak Ditekan | 3,14         |

Sumber: (Data Olahan, 2024)

Berdasarkan Tabel 4.3 dapat dilihat bahwa tegangan keluaran yang dihasilkan ketika *push button* 1 ditekan yaitu 0 V dan pada saat tidak ditekan maka tegangan yang dihasilkan yaitu 3,14 V, sedangkan pada saat *push button* 2 ditekan maka tegangan yang dihasilkan yaitu 0 V dan pada saat tidak ditekan maka tegangan keluaran yang dihasilkan 3,14 V. Dengan kondisi *push button* menggunakan aktif *low*.

### 4.2.4. Pengujian Sensor PZEM-004T

Pengujian sensor dilakukan untuk mengetahui bahwa sensor ini dapat difungsikan dengan baik. Hal ini bertujuan untuk mengetahui bahwa sensor PZEM-004T ini dapat mendeteksi arus dan tegangan. Penggunaan sensor ini digunakan untuk mendeteksi tegangan dan arus pompa air. Pengujian ini dilakukan dengan membandingkan tegangan sensor dengan tegangan alat ukur. Pengujian sensor PZEM-004T ini dapat dilihat pada Tabel 4.4.

Tabel 4.4 Pengujian Sesor PZEM-004T

| No | Tegangan Input (V) | Tegangan Alat Ukur (V) | Tegangan Sensor (V) |
|----|--------------------|------------------------|---------------------|
| 1  | 70                 | 70                     | 0                   |
| 2  | 80                 | 80                     | 0                   |
| 3  | 90                 | 91                     | 90,8                |
| 4  | 100                | 100                    | 100,6               |
| 5  | 110                | 110                    | 108,1               |

Sumber: (Data Olahan, 2024)

Berdasarkan pegujian yang telah dilakukan pada saat sensor diberi tegangan *input* 70 V sampai 80 V maka tegangan keluaran yang terbaca pada sensor yaitu 0 V dan pada alat ukur terbaca sama dengan tegangan *input* yang diberikan. Tegangan

keluaran yang terbaca dari sensor apabila nilai tegangan *input* bernilai 90 V maka nilai yang terbaca dari sensor sebesar 90,8 V. Pada saat tegangan *input* dinaikkan menjadi 100 V maka tegangan keluaran dari sensor yang terbaca 100,6 V dan begitu seterusnya jika tegangan *input* dinaikkan.

#### 4.2.5. Pengujian Aplikasi Blynk

Pengujian aplikasi Blynk dilakukan untuk mengetahui apakah sudah terhubung dan untuk mengetahui apakah aplikasi Blynk berkerja sesuai dengan fungsinya. Hasil pengujian aplikasi Blynk dapat dilihat pada Tabel 4.5.

Tabel 4.5 Pengujian Aplikasi Blynk

| Percobaan | Tegangan (V) | Arus (A) | Waktu kirim |
|-----------|--------------|----------|-------------|
| 1         | 241,2        | 1,6      | 1 detik     |
| 2         | 241,7        | 1,6      | 1 detik     |
| 3         | 241,2        | 1,6      | 1 detik     |
| 4         | 241,1        | 1,7      | 1 detik     |
| 5         | 241,3        | 1,6      | 1 detik     |

Sumber: (Data Olahan, 2024)

Data hasil pengujian aplikasi Blynk dengan menggunakan beban pompa air 220 V. Sumber tegangan didapat langsung dari sumber listrik yang kemudian dihubungkan untuk menjadikan sumber pada alat dan saat pada pengujian dilakukan. Pada saat pengujian terdapat *delay* pada PZEM-004T dan Blynk selama 1 detik pada saat pengiriman data.

#### 4.2.6. Pengujian Power Supply 5 V

Pengujian ini dilakukan untuk mengetahui *power supply* yang digunakan bekerja atau tidak. Berikut hasil pengujian yang telah dilakukan dapat dilihat pada Tabel 4.6.

Tabel 4.6 Pengujian Power Supply

| No | Tegangan <i>Input</i> AC (V) | Tegangan <i>Output</i> DC (V) |
|----|------------------------------|-------------------------------|
| 1  | 0                            | 0                             |
| 2  | 90                           | 5,04                          |
| 3  | 220                          | 5,04                          |

Sumber: (Data Olahan, 2024)

Berdasarkan Tabel 4.6 dapat diketahui bahwa *power supply* tidak berkerja ketika tegangan *input* 0 V, jika diberi tegangan *input* 90 V maka tegangan *output* sebesar 5,04 V dan jika diberikan tegangan *input* 220 V maka tegangan keluran sebesar 5,04 V.

### 4.3. Pengujian Alat Keseluruhan

Pengujian ini dilakukan untuk mengetahui berhasil atau tidaknya alat yang dibuat, setelah melalui pengujian tiap-tiap komponen dan proses perakitan maka dilakukan proses pengujian alat secara keseluruhan. Sebelum menjalankan sistem komponen-komponen tersebut diberi tegangan sesuai dengan ketentuannya. Modul NodeMCU ESP8266 digunakan untuk mengatur komponen-komponen seperti *relay*, sensor dan modul *real time clock* (RTC). Pengaturan mode manual dan otomatis pada alat ini atur melalui aplikasi Blynk. Pengujian alat ini dilakukan sebanyak 15 kali percobaan, pengujian alat keseluruhan ini dapat dilihat pada Tabel 4.7.

Tabel 4.7 Pengujian Alat Secara Keseluruhan

| Mode      | PB Start | PB Stop | Waktu    | Kondisi Motor        | Tegangan (V) | Arus (A) | Keterangan |
|-----------|----------|---------|----------|----------------------|--------------|----------|------------|
| Manual    | 1        | 0       | 08:08:10 | Aktif                | 240,1        | 1,64     | Sesuai     |
|           | 0        | 1       | 08:10:15 | Nonaktif             | 240          | 0        | Sesuai     |
| Manual    | 1        | 0       | 10:02:10 | Aktif                | 240,4        | 1,63     | Sesuai     |
|           | 0        | 1       | 10:03:20 | Nonaktif             | 240          | 0        | Sesuai     |
| Manual    | 1        | 0       | 15:12:20 | Aktif                | 240,3        | 1,6      | Sesuai     |
|           | 0        | 1       | 15:12:30 | Nonaktif             | 240          | 0        | Sesuai     |
| Manual    | 1        | 0       | 16:15:20 | Aktif                | 240          | 1,64     | Sesuai     |
|           | 0        | 1       | 16:16:30 | Nonaktif             | 240          | 0        | Sesuai     |
| Manual    | 1        | 0       | 17:20:20 | Aktif                | 239          | 1,6      | Sesuai     |
|           | 0        | 1       | 17:21:10 | Nonaktif             | 240          | 0        | Sesuai     |
| Otomatis  | -        | -       | 08:09:10 | Aktif selama 1 menit | 239          | 1,6      | Sesuai     |
| Otomatis  | -        | -       | 10:10:27 | Aktif selama 1 menit | 239          | 1,6      | Sesuai     |
| Otomatis  | -        | -       | 13:09:10 | Aktif selama 1 menit | 239          | 1,6      | Sesuai     |
| Otomatis  | -        | -       | 15:12:26 | Aktif selama 1 menit | 239          | 1,6      | Sesuai     |
| Otomatis  | -        | -       | 15:09:10 | Aktif selama 1 menit | 239          | 1,6      | Sesuai     |
| Manual    | 1        | 0       | 16:25:05 | Aktif                | 237          | 1,6      | Sesuai     |
|           | 0        | 1       | 16:25:15 | Nonaktif             | 239,5        | 0        | Sesuai     |
| Manual    | 1        | 0       | 17:00:05 | Aktif                | 237,4        | 1,16     | Sesuai     |
|           | 0        | 1       | 17:00:15 | Nonaktif             | 240          | 0,2      | Sesuai     |
| Manual    | 1        | 0       | 19:00:37 | Aktif                | 237,8        | 1,59     | Sesuai     |
|           | 0        | 1       | 19:00:47 | Nonaktif             | 237          | 0        | Sesuai     |
| Manual    | 1        | 0       | 19:00:37 | Aktif                | 237,3        | 1,61     | Sesuai     |
|           | 0        | 1       | 19:00:47 | Nonaktif             | 239,7        | 0        | Sesuai     |
| Manual    | 1        | 0       | 20:00:12 | Aktif                | 239,8        | 1,6      | Sesuai     |
|           | 0        | 1       | 20:00:24 | Nonaktif             | 239,1        | 0        | Sesuai     |
| Rata-rata |          |         |          |                      | 238,8        | 1,57     |            |

Sumber: (Data Olahan, 2024)

Dari hasil pengujian yang dilakukan alat berkerja sesuai sistem yang telah dirancang pengujian alat ini dengan menggunakan kontrol manual dengan *push button*, kontrol otomatis dengan pengaturan RTC dan kontrol manual dengan menggunakan aplikasi Bynk pada *smartphone*.

Pada saat pengujian manual pukul 08:08:10 WIB maka tegangan pada saat motor aktif dari pembacaan sensor 240,1 V dan arus 1,64 A dan pada kondisi nonaktif arus menjadi 0 V. Pada saat pengujian otomatis pukul 08:09:10 pompa air aktif selama 1 menit dengan tegangan keluaran 239 V dan arus 1,6 A.

Hasil pengujian dapat disimpulkan bahwa alat berkerja dengan tegangan 239 V dan arus 1,6 A. Nilai rata-rata tegangan 238,8 V AC dengan rata-rata arus 1,57 A pada saat kondisi pomp air aktif, penyemprotan dan penyiraman air ini dilakukan dengan waktu penyiraman sekitar 1 menit. Alat ini setelah melakukan penyemprotan sedangkan tingkat keberhasilan alat ini setelah diuji 100 % berhasil.

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1. Kesimpulan**

Berdasarkan hasil perancangan sistem penyemprotan pestisida otomatis dapat disimpulkan sebagai berikut:

1. Dari hasil pengujian RTC DS3231 memiliki selisih waktu 3 detik dari perbandingan waktu RTC dan waktu yang ada di jam waktu sekarang pada *handphone*.
2. Pengujian *relay* memiliki tegangan keluaran menuju pompa air sebesar 230 V AC pada saat kondisi *relay* terhubung.
3. Hasil pengujian aplikasi Blynk memiliki waktu *delay* 1 detik dalam pengiriman data sensor PZEM-004T ke aplikasi Blynk.
4. Dari hasil pengujian yang telah dilakukan sebanyak 15 kali percobaan pompa air berkerja dengan tegangan 239 V dan arus 1,6 A dan nilai rata-rata arus 1,57 A serta tegangan rata-rata 238,8.
5. Hasil pengujian *push button* tidak ditekan 3,14 V dan pada saat ditekan tegangan keluaran *push button* 0 V, hal ini dikarenakan kondisi *push button* menggunakan aktif *low*.
6. Hasil pengujian keseluruhan alat ini dapat berkerja sesuai dengan perancangan sistem yang dibuat dengan tingkat keberhasilan 100 %

#### **5.2. Saran**

Adapun saran untuk penelitian berikutnya agar perancangan alat dapat berjalan lebih baik yaitu:

1. Penelitian selanjutnya diharapkan dapat memberikan tempat khusus untuk cairan pestisidanya.
2. Penelitian selanjutnya juga dapat menambahkan *rain drop sensor*, agar mengetahui kondisi cuaca sehingga alat tersebut dapat menyemprotkan pestisida sesuai dengan kondisi cuaca.

3. Diharapkan penelitian selanjutnya dapat menambahkan modul *water censor* agar mengetahui kondisi air ketika air penuh atau berkurang, serta dapat juga menambahkan *flow water censor* agar mengetahui jumlah pemakaian debit air yang digunakan.

## DAFTAR PUSTAKA

- Anggraeni, N. T., & Fadlil, A. (2013). Sistem Identifikasi Citra Jenis Cabai (*Capsicum Annum L.*) Menggunakan Metode Klasifikasi City Block Distance. *Jurnal Sarjana Teknik Informatika*, 1(2), 409-418.
- Arifin, J., Zulita, L. N., & Hermawansyah. (2016). Perancangan Murottal Otomatis Menggunakan Mikrokontroler Arduino Mega 2560. *Jurnal Media Infotama*, 12(1), 89–98.
- Budiono, M. (2021). *Rancang Bangun Robot Penyemprot Pestisida Otonom Dengan Sistem Wall-Follower Pada Penyemprotan Tanaman Cabai (Tugas Akhir)*. Universitas Islam Negeri Sultan Syarif Kasim Riau, Pekanbaru.
- Dwiningtias, S. (2023). *Prototype Sistem Monitoring Semprot Pestisida Otomatis Berbasis IoT (Skripsi)*. Universitas Muhammadiyah Surakarta, Surakarta.
- Inzhagi, P. D., Muayyadi, A. A., & Fardan. (2023). Monitoring Sistem Pestisida Otomatis Pada Pertanian Cabai Berbasis Internet of Things (IoT). *e-Proceeding of Engineering*, 10(5), 4237–4245.
- Iqbar, M. Y., & Kartika, K. P. (2020). Rancang Bangun Lampu Portable Otomatis Menggunakan RTC Berbasis Arduino. *Jurnal Ilmiah Teknik Informatika*, 14(1), 61–72.
- Manullang, A. B., Saragih, Y., & Hidayat, R. (2021). Implementasi NodeMCU ESP8266 Dalam Rancang Bangun Sistem Keamanan Sepeda Motor Berbasis IoT. *Jurnal Informatika Dan Rekayasa Elektronika*, 4(2), 163-170.
- Munandar, A., Veronika, N. D. M., Abdulllah, D., & Sahputra, E. (2023). Perancangan Miniatur Mesin Pengisi Cairan Otomatis Menggunakan ESP32 Berbasis IoT (Internet of Things). *Jurnal Komitek*, 3(1), 69–78.
- Naldi, R. A. (2022). *Sistem Monitoring Dan Kontrol Pembibitan Kelapa Sawit Berbasis Internet of Things (Skripsi)*. Universitas Islam Negeri (UIN) Ar-Raniry Banda Aceh, Banda Aceh.
- Novaria, M. (2017). *Rancang Bangun Alat Anti Kebisingan Suara Guna Mendukung Etika Berkunjung Ke Rumah Sakit Berbasis Arduino Uno. (Tugas*

*Akhir*). Politeknik Negeri Sriwijaya, Palembang.

- Oka, I. K. D. A., Nirmala, B. P. W., & Putra, M. A. P. (2022). Model IoT Berbasis Fuzzy Tsukamoto Untuk Penyemprotan Pestisida Otomatis Pada Tanaman Sayur Kubis. *Progresif: Jurnal Ilmiah Komputer*, 18(2), 141–150.
- Pratikto, A., Hendrawan, W., & Agustini, N. P. (2022). Simulasi Kendali Dan Monitoring Daya Listrik Peralatan Rumah Tangga Berbasis ESP32. *Jurnal Alinier*, 3(1), 54–68.
- Rini, E. S. (2022). Analisa dan Rancang Bangun Pendeteksi Gangguan Fuse CUTE Out (FCO) Melalui Notifikasi Telegram Berbasis Global Positioning System (GPS) (*Skripsi*). Politeknik Negeri Bengkalis, Bengkalis.
- Samsinar, R., & Setiawan, A. (2023). Perancangan Alat Implementasi Internet of Things (IoT) Untuk Penyiraman Pestisida Dengan Metode Otomatis Berbasis Aplikasi Blynk Pada Tanaman. *Jurnal Elektronika Kendali Telekomunikasi Tenaga Listrik Komputer*, 6(2), 143–148.
- Sandi, G. H., & Fatma, Y. (2023). Pemanfaatan Teknologi Internet of Things (IoT) Pada Bidang Pertanian. *Jurnal Mahasiswa Teknik Informatika*, 7(1), 1–5.
- Santosa, S. P., & Nugroho, R. M. W. (2021). Rancang Bangun Alat Pintu Geser Otomatis Menggunakan Motor DC 24 V. *Jurnal Ilmiah Elektrokrina*, 9(1), 38–45.
- Sulfiani, N., & Firmawati, N. (2019). Rancang Bangun Sistem Penyemprot Tanaman Otomatis Berdasarkan Waktu Dengan Real Time Clock (RTC) Dan Sensor Ultrasonik Serta Notifikasi Via SMS. *Jurnal Ilmu Fisika*, 11(2), 62–71.
- Thooriq, M. A., Sujatmika, A. R., & Umami, I. (2023). Rancang Bangun Alat Penyiraman Dan Pembasmi Hama Otomatis Pada Tanaman Bayam Dengan Monitoring Berbasis Website. *Jurnal Sains dan Teknologi (JSIT)*, 3(1), 178–183.
- Utomo, T. P. (2019). Potensi Implementasi Internet of Things (IoT) Untuk Perpustakaan. *Buletin Perpustakaan Universitas Islam Indonesia*, 2(1), 1–18.

## LAMPIRAN

### Lampiran 1 Program Alat Keseluruhan

```
#define BLYNK_TEMPLATE_ID "TMPL6d4VFDRZB"
#define BLYNK_TEMPLATE_NAME "kasih air"
#define BLYNK_AUTH_TOKEN "7sjPQkN680UFsa51ljd_AhsFf7l5vGFC"

#define BLYNK_PRINT Serial
#include <Blynk.h>
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include "RTClib.h"
#include <Wire.h>
#include <PZEM004Tv30.h>
BlynkTimer timer;

char auth[] = BLYNK_AUTH_TOKEN;
char ssid[] = "Sona Othiwa"; // Nama Ssid Wifi
char pass[] = "jeJaK123"; // Password Wifi
#define relay D4
#define start D5
#define stop D6
PZEM004Tv30 pzemR(D3, D0);
RTC_DS3231 rtc;
char daysOfTheWeek[7][12] = {"Minggu", "Senin", "Selasa", "Rabu", "Kamis",
    "Jum'at", "Sabtu"};
int jam, menit, detik;
int tanggal, bulan, tahun;
float voltage,current;
```

```

String hari;

String harisiram1 = "Senin";
String harisiram2 = "Rabu";
String harisiram3 = "jum'at";

int jam1 = 17;
int jam2 = 18;
int jam3 = 19;
int menit1 = 47;
int menit2 = 49;
int menit3 = 51;
int detiksiram = 10;
int waktusiram = 20;//waktu siram dalam detik
int state=0;
bool relayActive = false;
bool blynkk = false;

float VR = 0;
float IR = 0;

void sendSensor()
{
  Blynk.virtualWrite(V0, voltage);
  Blynk.virtualWrite(V1, current);
}

void setup() {
  // put your setup code here, to run once:
  pinMode(relay,OUTPUT);
  pinMode(start,INPUT_PULLUP);
  pinMode(stop,INPUT_PULLUP);
}

```

```

digitalWrite(relay,HIGH);
Serial.begin(9600);
Blynk.begin(auth, ssid, pass);
if (! rtc.begin()) {
  Serial.println("Couldn't find RTC");
  Serial.flush();
  while (1) delay(10);

}
timer.setInterval(100L, sendSensor);
//rtc.adjust(DateTime(F(_DATE), F(TIME_)));
//rtc.adjust(DateTime(2024, 8, 10, 15, 41, 40));

}

void loop() {
  // put your main code here, to run repeatedly:
  Blynk.run();
  timer.run();
  DateTime now = rtc.now();
  jam   = now.hour();
  menit = now.minute();
  detik = now.second();
  tanggal = now.day();
  bulan  = now.month();
  tahun  = now.year();
  hari   = daysOfTheWeek[now.dayOfTheWeek()];
  Serial.print(String() + hari + ", " + tanggal + "-" + bulan + "-" + tahun);
  Serial.print(" ");
  Serial.print(String() + jam + ":" + menit + ":" + detik);
  Serial.print(" ");
}

```

```

if ((hari == harisiram1 || hari == harisiram2 || hari == harisiram3) &&
    (jam == jam1 || jam == jam2 || jam == jam3) && (menit == menit1 || menit ==
menit2 || menit == menit3) && detik == detiksiram) {
    digitalWrite(relay,LOW);
} else if (detik == detiksiram + waktusiram) {
    digitalWrite(relay,HIGH);
}

/*if((hari == harisiram1 || hari == harisiram2 || hari == harisiram3) && (jam ==
jam1) && (menit == menit1) && (detik == detiksiram)){
    digitalWrite(relay,LOW);
} else if (detik == detiksiram + waktusiram) {
    digitalWrite(relay,HIGH);
}*/ //ini 1 kali dalam 1 hari siram

voltage = pzemR.voltage();
current = pzemR.current();
if(isnan(voltage)||isnan(current)){
    VR=0; Serial.print("Voltage: "); Serial.print(voltage); Serial.print("V");
Serial.print(" ");
    IR=0; Serial.print("Current: "); Serial.print(current); Serial.print("A");
Serial.println(" Error Reading Voltage and Current");
}
else {
    VR=voltage;Serial.print("Voltage:
"); Serial.print(voltage); Serial.print("V"); Serial.print(" ");
    IR=current;Serial.print("Current:
"); Serial.print(current); Serial.println("A");
}
Serial.print(" ");

```

```
if(!blynk){  
  if(digitalRead(start)==LOW){  
    digitalWrite(relay,LOW);  
  } else if(digitalRead(stop)==LOW){  
    digitalWrite(relay,HIGH);  
  }  
  delay(200);  
}}
```

```
BLYNK_WRITE(V2) {  
  int pinValue = param.asInt();  
  digitalWrite(relay, pinValue);  
}
```

## **Lampiran 2 Perbaikan Dan Saran**



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET DAN TEKNOLOGI

POLITEKNIK NEGERI BENGKALIS

JURUSAN TEKNIK ELEKTRO

Jalan Bathin Alam, Sungai alam Bengkalis-Riau 28714

Telepon (0766) 24566, Faximile (0766) 800 1000

Laman: <http://www.polbeng.ac.id>



LEMBAR SARAN DAN PERBAIKAN SIDANG TA/SKRIPSI

T A : 2023/ 2024

Nama : Indah Tra Alfina  
NIM : .....  
Judul : .....

Nama Dosen Pembimbing / Dosen Penguji\* : Syaiful Amri

Materi perbaikan dari Dosen Pembimbing / Dosen Penguji\* :

+ p...

| Pengesahan dari Dosen Pembimbing / Dosen Penguji* |                  |                   |                  |
|---|------------------|-------------------|------------------|
| Sebelum perbaikan                                 |                  | Setelah perbaikan |                  |
| Tanggal   | <u>16/8/2024</u> | Tanggal           | <u>22/8/2024</u> |
| Tanda Tangan                                      |                  | Tanda Tangan      |                  |

- CATATAN : 1. Form ini mohon dikembalikan ke Koordinator setelah pelaksanaan sidang selesai.  
2. Tanda \* = coret salah satu



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET DAN TEKNOLOGI  
**POLITEKNIK NEGERI BENGKALIS**  
**JURUSAN TEKNIK ELEKTRO**  
 Jalan Bathin Alam, Sungai alam Bengkalis-Riau 28714  
 Telepon (0766) 24566, Faximile (0766) 800 1000  
 Laman: <http://www.polbeng.ac.id>



LEMBAR SARAN DAN PERBAIKAN SIDANG TA/ ~~SKRIPSI~~

T A : 2023/ 2024

Nama : Indah Tara alfa  
 NIM : \_\_\_\_\_  
 Judul : \_\_\_\_\_

Nama Dosen Pembimbing / Dosen Penguji\* : Agustian

Materi perbaikan dari Dosen Pembimbing / Dosen Penguji\* :  
Alasan.

| Perbaikan dari Dosen Pembimbing / Dosen Penguji* |                              |
|--|------------------------------|
| Sebelum perbaikan<br>Tanggal                     | Setelah perbaikan<br>Tanggal |
| 16/8/24  | 21/8/24                      |
| Tanda Tangan                                     | Tanda Tangan                 |

CATATAN : 1. Form ini mohon dikembalikan ke Koordinator setelah pelaksanaan sidang selesai.  
 2. Tanda \* = coret salah satu



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET DAN TEKNOLOGI  
**POLITEKNIK NEGERI BENGKALIS**  
**JURUSAN TEKNIK ELEKTRO**

Jalan Bathin Alam, Sungai alam Bengkalis-Riau 28714  
 Telepon (0766) 24566, Faximile (0766) 800 1000  
 Laman: <http://www.polbeng.ac.id>



LEMBAR SARAN DAN PERBAIKAN SIDANG TA/SKRIPSI

T A : 2023/ 2024

Nama : Indah Tina Alfarra  
 NIM : .....  
 Judul : .....

Nama Dosen Pembimbing / Dosen Penguji\* : Abdul Hadi

Materi perbaikan dari Dosen Pembimbing / Dosen Penguji\* :

1. Penulisan di perbaikan
2. ....

*(Handwritten signature)*

| Pengesahan dari Dosen Pembimbing / Dosen Penguji* |                    |                   |                    |
|---|--------------------|-------------------|--------------------|
| Sebelum perbaikan                                 | <u>16/8/2024</u>   | Setelah perbaikan | <u>21/8/2024</u>   |
| Tanggal   |                    | Tanggal           |                    |
| Tanda Tangan                                      | <i>(Signature)</i> | Tanda Tangan      | <i>(Signature)</i> |

CATATAN : 1. Form ini mohon dikembalikan ke Koordinator setelah pelaksanaan sidang selesai.  
 2. Tanda \* = coret salah satu

### **Lampiran 3 Lembar Asestensi**



**JURUSAN TEKNIK ELEKTRO  
POLITEKNIK NEGERI BENGKALIS**

Jl. Bathin Alam, Sungai Alam, Bengkalis, Riau, 28711  
Telepon (+62766) 24566, Fax (+62766) 800 1000  
Laman [http:// www.polbeng.ac.id](http://www.polbeng.ac.id) Email : polbeng@polbeng.ac.id

**LEMBAR ASISTENSI SKRIPSI**

Nama : Indah Tria Alfina  
NIM : 3103211297  
Jurusan/Prodi : Teknik Elektro/D3 Teknik Elektronika  
Judul : Implementasi Sistem Penyemprotan Pestisida otomatis berdasarkan tanaman Cabai berbasis Internet of Things (IoT)

| No. | Hari/Tanggal        | Uraian  | Paraf Pembimbing |
|-----|---------------------|---|------------------|
| 1   | Senin 12/2/2024     | Pembahasan tentang perbaikan tata tulis           | d                |
| 2   | Kamis 28/5/2024     | Pembahasan tentang flowchat Sistem                | d                |
| 3   | Senin 01/03/2024    | Pembahasan tentang Revisi Proposal Tugas Akhir I  | d                |
| 4   | Senin 07/06/2024    | Pembahasan tentang Revisi Proposal Tugas Akhir II | d                |
| 5   | Selasa 2/07/2024    | Pembahasan tentang Progres Alat TA                | d                |
| 6   | Rabu 12 Juni/2024   | Pembahasan tentang Alat                           | d                |
| 7   | Jumat 8/ Juli /2024 | Pembahasan tentang Program Alat                   | d                |
| 8   | Selasa 30/6/2024    | Pembahasan tentang Alat                           | d                |
| 9   | Senin 3/6/2024      | Pembahasan tentang alat                           | d                |
| 10  | Rabu 12/6/2024      | Pembahasan tentang alat                           | d                |
| 11  | Rabu 14/08/2024     | Pembahasan tentang Pengambilan data Keseluruhan   | d                |
| 12  | Kamis 15/08/2024    | Pembahasan tentang hasil pengujian alat           | d                |

Bengkalis, 15 Agustus 2024  
Dosen Pembimbing,

(Hikmatul Anri, S.ST., M.T.)  
NIP. 198803062018031001