

**RANCANG BANGUN SISTEM PENYIRAMAN JAMUR
TIRAM UNTUK OPTIMASI PERTUMBUHAN BERBASIS
INTERNET OF THINGS**

LAPORAN TUGAS AKHIR



oleh

**Maulana Iqbal Muhlisin
NIM E32231115**

**PROGRAM STUDI TEKNIK KOMPUTER
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI JEMBER
2026**

**RANCANG BANGUN SISTEM PENYIRAMAN JAMUR
TIRAM UNTUK OPTIMASI PERTUMBUHAN BERBASIS
INTERNET OF THINGS**

LAPORAN TUGAS AKHIR



sebagai salah satu syarat untuk memperoleh gelar Ahli Madya (A.Md)
di Program Studi Teknik Komputer
Jurusan Teknologi Informasi

oleh

Maulana Iqbal Muhlisin

NIM E32231115

**PROGRAM STUDI TEKNIK KOMPUTER
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI JEMBER
2026**

KEMENTERIAN PENDIDIKAN TINGGI, SAINS, DAN TEKNOLOGI
POLITEKNIK NEGERI JEMBER
JURUSAN TEKNOLOGI INFORMASI

RANCANG BANGUN SISTEM PENYIRAMAN JAMUR
TIRAM UNTUK OPTIMASI PERTUMBUHAN BERBASIS
INTERNET OF THINGS

Maulana Iqbal Muhlisin (E32231115)

Telah Diuji pada Tanggal 01 Juli 2026
dan Dinyatakan Memenuhi Syarat

Ketua Penguji,

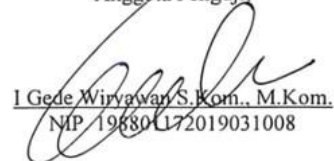


Denny Wijanarko, ST, MT.
NIP. 19780908 200501 1 001

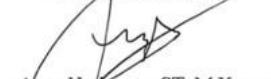
Sekretaris Penguji,


Agus Hariyanto, ST, M.Kom.
NIP. 19780817 200312 1 005

Anggota Penguji,


I Gede Wiryawan, S.Kom., M.Kom.
NIP. 198804172019031008

Dosen Pembimbing,


Agus Hariyanto, ST, M.Kom.
NIP. 19780817 200312 1 005

Mengesahkan

Ketua Jurusan Teknologi Informasi


Indra Yulit Riskiawan, S.Kom, M.Cs.
NIP. 19830203 200604 1 003

SURAT PERNYATAAN MAHASISWA

Saya yang bertanda tangan dibawah ini:

Nama : Maulana Iqbal Muhlisin

NIM : E32231115

Menyatakan dengan sebenar-benarnya bahwa segala pernyataan dalam Laporan Akhir yang berjudul **"Rancang Bangun Sistem Penyiraman Jamur Tiram Untuk Optimasi Pertumbuhan Berbasis Internet of Things"** Merupakan gagasan dan hasil karya saya sendiri dengan arahan dosen pembimbing dan belum pernah diajukan ke dalam bentuk apapun pada perguruan tinggi manapun.

Semua informasi yang digunakan telah dinyatakan secara jelas dan dapat diperiksa kebenarannya. Sumber informasi yang berasal dari kutipan karya yang diterbitkan oleh penulis lain telah disebutkan dalam naskah dan dicantumkan dalam Daftar Pustaka pada bagian akhir Tugas Akhir ini.

Jember, 01 Juli 2026



Maulana Iqbal Muhlisin
NIM E32231115



**PERNYATAAN
PERSETUJUAN PUBLIKASI
KARYA ILMIAH UNTUK KEPENTINGAN
AKADEMIS**

Yang bertandatangan dibawah ini, saya:

Nama : Maulana Iqbal Muhlisin
NIM : E32231115
Program Studi : Teknik Komputer
Jurusan : Teknologi Informasi

Demi pengembangan Ilmu Pengetahuan, saya menyetujui untuk memberikan kepada UPT. Perpustakaan Politeknik Negeri Jember, Hak Bebas Royalti Non-Eksklusif (*Non-Exclusive Royalty Free Right*) atas Karya Ilmiah berupa Laporan Akhir saya yang berjudul :

**RANCANG BANGUN SISTEM PENYIRAMAN JAMUR TIRAM UNTUK
OPTIMASI PERTUMBUHAN BERBASIS INTERNET OF THINGS**

Dengan Hak Bebas Royalti Non-Eksklusif ini UPT. Perpustakaan Politeknik Negeri Jember berhak menyimpan, mengalih media atau format, mengelola dalam bentuk Pangkalan Data (*Database*), mendistribusikan karya dan menampilkannya atau mempublikasikannya di Internet atau media lain untuk kepentingan akademis tanpa perlu meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis atau pencipta. Saya bersedia untuk menanggung secara pribadi tanpa melibatkan pihak Politeknik Negeri Jember, segala bentuk tuntutan hukum yang timbul atas pelanggaran Hak Cipta dalam karya ilmiah ini. Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Jember
Pada Tanggal : 17 Juni 2026

Yang menyatakan,



Maulana Iqbal Muhlisin
NIM. E32231115

MOTTO

"Barang siapa bertakwa kepada Allah, niscaya Dia akan membukakan jalan keluar baginya dan memberinya rezeki dari arah yang tidak disangka-sangka."
(QS. At-Talaq: 2–3)

HALAMAN PERSEMBAHAN

Alhamdulillahirabbil'alamin, segala puji dan syukur hanya milik Allah SWT, Tuhan semesta alam, yang atas izin dan ridho-Nya karya ini dapat terselesaikan. Shalawat serta salam senantiasa tercurahkan kepada Nabi Muhammad SAW, suri teladan terbaik sepanjang masa. Dengan penuh rasa hormat, cinta, dan syukur, karya ini saya persembahkan kepada semua pihak yang telah memberikan doa, dukungan, semangat, serta motivasi selama proses penyusunan tugas akhir ini. Tanpa kehadiran dan bantuan mereka, perjalanan dalam menyelesaikan karya ini tentu tidak akan mudah. Semoga segala kebaikan yang telah diberikan mendapatkan balasan terbaik dari Allah SWT.

1. Orang Tua Tercinta, yang tidak pernah berhenti mendoakan, tidak pernah lelah mendukung, dan tidak pernah meminta balasan atas semua yang telah diberikan. Setiap tetes keringat, pengorbanan, dan doa kalian adalah kekuatan terbesar yang mengantarkan saya hingga mampu menyelesaikan tugas akhir ini. Karya ini saya persembahkan sebagai wujud rasa syukur dan terima kasih atas kasih sayang yang tak ternilai.
2. Bapak Agus Hariyanto, S.T., M.Kom., selaku Dosen Pembimbing yang dengan penuh kesabaran telah memberikan bimbingan, arahan, masukan, serta motivasi selama proses penyusunan tugas akhir. Setiap ilmu dan pengalaman yang Bapak berikan menjadi bekal yang sangat berharga bagi saya, baik dalam menyelesaikan penelitian maupun dalam menghadapi dunia kerja di masa mendatang.
3. Teman-teman seperjuangan, yang telah memberikan semangat, bantuan, serta kebersamaan selama proses penyusunan tugas akhir. Terima kasih atas setiap diskusi, masukan, dan dukungan yang telah diberikan, sehingga penelitian mengenai *Rancang Bangun Sistem Penyiraman Jamur Tiram untuk Optimasi Pertumbuhan Berbasis Internet of Things* dapat diselesaikan dengan baik.
4. Almamater tercinta, Politeknik Negeri Jember, yang telah menjadi tempat saya menimba ilmu, mengembangkan kemampuan, serta memperoleh

banyak pengalaman berharga selama masa perkuliahan. Semoga ilmu yang telah diperoleh dapat bermanfaat bagi masyarakat dan menjadi bekal untuk terus berkarya di masa depan.

Akhir kata, saya menyadari bahwa keberhasilan dalam menyelesaikan tugas akhir ini tidak terlepas dari doa, bantuan, dan dukungan berbagai pihak. Semoga karya ini dapat menjadi referensi bagi penelitian selanjutnya dalam pengembangan sistem penyiraman otomatis berbasis *Internet of Things* untuk optimasi pertumbuhan jamur tiram.

RINGKASAN

Rancang Bangun Sistem Penyiraman Jamur Tiram Untuk Optimasi Pertumbuhan Berbasis *Internet Of Things*: Maulana Iqbal Muhlisin, NIM E32231115, Tahun 2026, Teknik Komputer, Teknologi Informasi, Politeknik Negeri Jember, Agus Hariyanto S.T., M.Kom.

Penelitian ini membahas perancangan dan implementasi sistem penyiraman jamur tiram berbasis *Internet of Things* (IoT) yang bertujuan menjaga kondisi kelembapan kumbung agar tetap optimal. Sistem menggunakan ESP32 sebagai mikrokontroler utama, sensor DHT22 untuk mengukur kelembapan, *relay* sebagai pengendali pompa air, serta *Supabase* sebagai media monitoring berbasis *cloud*. Data kondisi lingkungan dikirim secara *real-time* melalui jaringan *Wi-Fi* sehingga dapat dipantau dari jarak jauh menggunakan aplikasi monitoring.

Sistem dirancang memiliki mode otomatis dan manual. Pada mode otomatis, pompa air akan aktif ketika kelembapan berada di bawah batas yang ditentukan dan berhenti saat kondisi kelembapan telah mencapai rentang ideal untuk pertumbuhan jamur tiram, yaitu sekitar 70–90%. Sementara itu, pada mode manual, pengguna dapat mengontrol perangkat secara langsung melalui aplikasi sesuai kebutuhan.

Hasil implementasi menunjukkan bahwa seluruh komponen sistem dapat bekerja dan terintegrasi dengan baik. Sistem mampu melakukan monitoring kelembapan secara *real-time* serta mengendalikan proses penyiraman secara otomatis sesuai kondisi lingkungan. Dengan adanya sistem ini, proses budidaya jamur tiram menjadi lebih efisien karena mengurangi ketergantungan pada penyiraman manual dan memudahkan pengawasan kondisi kumbung. Penerapan teknologi IoT pada sistem yang dikembangkan diharapkan dapat membantu meningkatkan produktivitas budidaya jamur tiram secara lebih efektif dan terkontrol.

PRAKATA

Puji syukur ke hadirat Allah SWT, atas segala limpahan rahmat, karunia, dan hidayah yang telah diberikan, sehingga penulis dapat menyusun dan menyelesaikan Tugas Akhir dengan judul "*Rancang Bangun Sistem Penyiraman Jamur Tiram Untuk Optimasi Pertumbuhan Berbasis Internet Of Things*" dengan baik dan tepat pada waktunya.

Tugas akhir ini disusun sebagai salah satu syarat untuk memperoleh gelar Ahli Madya (A.Md.) pada Program Studi Diploma III Teknik Komputer, Jurusan Teknologi Informasi, Politeknik Negeri Jember. Penyusunan Tugas Akhir ini bertujuan untuk menerapkan ilmu pengetahuan dan keterampilan yang telah diperoleh selama masa perkuliahan dalam bentuk perancangan dan implementasi sistem penyiraman berbasis *Internet of Things* (IoT) pada budidaya jamur tiram. Penyusunan Tugas Akhir ini tidak terlepas dari bantuan, dukungan, bimbingan, serta arahan dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan rasa terima kasih yang sebesar-besarnya kepada:

1. Allah SWT yang telah memberikan kesehatan, kemudahan, dan kelancaran sehingga penulis dapat menyelesaikan Tugas Akhir ini dengan tepat pada waktunya.
2. Kedua orang tua dan keluarga yang senantiasa memberikan doa, dukungan, motivasi, serta semangat kepada penulis.
3. Bapak Saiful Anwar, S.TP., M.P. selaku Direktur Politeknik Negeri Jember.
4. Bapak Hendra Yufit Riskiawan, S.Kom., M.Cs. selaku Ketua Jurusan Teknologi Informasi.
5. Bapak Agus Hariyanto, S.T., M.Kom. selaku Dosen Pembimbing yang telah memberikan bimbingan, arahan, masukan, serta motivasi selama proses penyusunan Tugas Akhir.
6. Bapak Denny Wijanarko, S.T., M.T. selaku Dosen Panelis yang telah memberikan saran dan masukan untuk penyempurnaan Tugas Akhir ini.
7. Seluruh dosen dan tenaga kependidikan Politeknik Negeri Jember yang telah memberikan ilmu pengetahuan dan pelayanan akademik selama masa studi.

8. Teman-teman seperjuangan serta semua pihak yang telah membantu dan memberikan bantuan, dukungan, semangat, serta kebersamaan selama proses penyusunan Tugas Akhir ini.

Penulis menyadari bahwa Tugas Akhir ini masih memiliki berbagai kekurangan dan keterbatasan. Oleh karena itu, penulis mengharapkan kritik dan saran yang bersifat membangun sebagai bahan evaluasi dan perbaikan di masa yang akan datang. Penulis berharap Tugas Akhir ini dapat memberikan manfaat, menambah wawasan, serta menjadi referensi bagi pembaca, khususnya dalam pengembangan teknologi *Internet of Things* pada bidang budidaya jamur tiram.

Jember, 01 Juli 2026

Penulis,



Maulana Iqbal Muhlisin

DAFTAR ISI

	Halaman
HALAMAN SAMPUL.....	i
HALAMAN JUDUL	ii
HALAMAN PENGESAHAN.....	iii
SURAT PERNYATAAN MAHASISWA.....	iv
SURAT PERNYATAAN PUBLIKASI	v
HALAMAN MOTTO	vi
HALAMAN PERSEMBAHAN	vii
RINGKASAN	ix
PRAKATA	x
DAFTAR ISI.....	xii
DAFTAR GAMBAR.....	xv
DAFTAR TABEL	xvi
DAFTAR LAMPIRAN	xvii
BAB 1. PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan	2
1.4 Manfaat.....	2
BAB 2. TINJAUAN PUSTAKA.....	3
2.1 Internet of Things (IoT) dalam Pertanian	3
2.2 Sensor Suhu dan Kelembapan (DHT22).....	3
2.3 Mikrokontroler (ESP32/NodeMCU)	4
2.3.1 Pin Daya (<i>Power</i>).....	5
2.3.2 Pin Digital (GPIO)	5
2.3.3 Pin Analog (ADC).....	6
2.3.4 Pin DAC (<i>Output Analog</i>).....	6
2.3.5 Pin Komunikasi.....	6
2.3.6 Pin Khusus	7

2.4 Modul Relay.....	7
2.4.1 Pin Kontrol	8
2.4.2 Terminal Output Relay	8
2.5 Stepdown LM2596	9
2.6 Prinsip Kerja Sistem	10
2.7 State of the Art	10
BAB 3. METODE KEGIATAN	12
3.1 Waktu dan Tempat Penelitian	12
3.1.1 Waktu Penelitian.....	12
3.1.2 Tempat Penelitian	12
3.2 Alat dan Bahan.....	12
3.2.1 Alat.....	13
3.2.2 Bahan	14
3.3 Metodologi Penelitian	15
3.3.1 Analisis Kebutuhan	15
3.3.2 Perancangan Alat dan Sistem	17
3.3.3 Implementasi Alat dan Sistem	21
3.3.4 Evaluasi dan Pengujian Alat dan Sistem	22
3.4 Jadwal Pelaksanaan	25
BAB 4. HASIL DAN PEMBAHASAN	26
4.1 Analisis Kebutuhan.....	26
4.2 Perancangan Sistem dan Alat	27
4.2.1 Perancangan Sistem.....	27
4.2.2 Perancangan Alat.....	28
4.3 Implementasi Alat dan Sistem	29
4.3.1 Implementasi Alat	30
4.3.2 Implementasi Sistem	32
4.3.3 Implementasi Sistem Monitoring Berbasis IoT.....	36
4.4 Evaluasi dan Pengujian Sistem dan Alat	39
4.4.1 Pengujian Kalibrasi Kelembapan	39
4.4.2 Pengujian Sistem Penyiraman Otomatis	40
4.4.3 Validasi Sensor DHT22 dan <i>Hygrometer</i>	41
4.4.4 Kinerja Sistem dalam Menjaga Kondisi Optimal Pertumbuhan Jamur Tiram	43

BAB 5. PENUTUP	50
5.1 Kesimpulan	50
5.2 Saran	51
DAFTAR PUSTAKA	52
LAMPIRAN.....	54

DAFTAR GAMBAR

	Halaman
2.1 Sensor DHT22	3
2.2 Mikrokontroler ESP32	4
2.3 Modul relay	7
2.4 Stepdown LM2596.....	9
3.1 Metodologi penelitian	15
3.2 Skematik Sistem.....	18
3.3 Tata letak penempatan komponen sistem	19
4.1 Flowchart Sistem	27
4.2 Perancangan alat	28
4.3 Gambar alat.....	31
4.4 Kumbung jamur tiram.....	32
4.5 Logo antigravity.....	34
4.6 Halaman login.....	34
4.7 Halaman register	34
4.8 Halaman histori.....	35
4.9 Halaman dashboard.....	36
4.10 Tampilan dashboard aplikasi mobile	37
4.11 Tampilan database	37
4.12 Kondisi lampu menyala	41
4.13 Kondisi nozzle menyala	41
4.14 Hasil pengukuran kelembapan rendah	42
4.15 Hasil pengukuran kelembapan tinggi.....	42
4.16 Screenshot database 21 Mei 2026.....	46
4.17 Screenshot database 22 Mei 2026.....	46
4.18 Screenshot database 23 Mei 2026.....	46
4.19 Screenshot database 24 Mei 2026.....	47
4.20 Hasil monitoring data real-time	48

DAFTAR TABEL

	Halaman
2.1 Pin DHT22	4
2.2 Pin daya ESP32	5
2.3 Pin digital ESP32	5
2.4 Pin analog ESP32	6
2.5 Pin DAC ESP32	6
2.6 Pin komunikasi ESP32	6
2.7 Pin khusus ESP32	7
2.8 Pin kontrol relay	8
2.9 Pin output relay	8
2.10 Pin stepdown LM2596	9
2.11 State of the art penelitian	11
3.1 Nama alat perangkat keras	13
3.2 Nama alat perangkat lunak	13
3.3 Nama bahan	14
3.4 Kalibrasi Kelembapan Udara	20
3.5 Jadwal pelaksanaan	25
4.1 Parameter kelembapan sistem	39
4.2 Hasil respons sistem penyiraman otomatis	40
4.3 Validasi data kelembapan	41
4.4 Hasil pengamatan kelembapan	43
4.5 Perbandingan penyiraman manual dan otomatis	47

DAFTAR LAMPIRAN

	Halaman
Lampiran 1 Kode Program.....	54

BAB 1. PENDAHULUAN

1.1 Latar Belakang

Budidaya jamur tiram merupakan salah satu kegiatan pertanian yang memiliki prospek pengembangan yang baik karena jamur tiram memiliki nilai gizi tinggi serta permintaan pasar yang terus meningkat. Menurut Eteruddin *et al.*, (2024) pertumbuhan jamur tiram sangat dipengaruhi oleh kondisi lingkungan, khususnya suhu dan kelembapan udara di dalam kumbung. Pengelolaan lingkungan yang kurang tepat dapat menyebabkan pertumbuhan jamur tidak optimal, menurunkan kualitas hasil panen, bahkan menyebabkan kegagalan produksi. Proses penyiraman jamur tiram masih banyak dilakukan secara manual sehingga bergantung pada pengalaman pembudidaya dan berpotensi menimbulkan ketidakkonsistenan dalam menjaga kondisi lingkungan tumbuh.

Perkembangan teknologi informasi dan komunikasi khususnya *Internet of Things* (IoT), membuka peluang penerapan sistem otomatis dalam bidang pertanian termasuk pada budidaya jamur tiram. Pemanfaatan sensor suhu dan kelembapan yang terintegrasi dengan mikrokontroler dan jaringan internet memungkinkan kondisi lingkungan dipantau secara *real-time* serta dikendalikan secara otomatis (Pramana *et al.*, 2025). Sistem penyiraman mampu menyesuaikan waktu dan durasi penyiraman sesuai kondisi lingkungan, sehingga penggunaan air menjadi lebih efisien dan kestabilan lingkungan kumbung dapat terjaga. Penerapan teknologi ini juga diharapkan dapat mengurangi ketergantungan pada penyiraman manual dan meminimalkan kesalahan akibat faktor manusia.

Penelitian ini bertujuan untuk merancang dan membangun sistem penyiraman jamur tiram berbasis *internet of things* yang mampu memantau serta mengendalikan kelembapan kumbung secara otomatis. Sistem yang dikembangkan diharapkan dapat mengatur proses penyiraman sesuai dengan kebutuhan pertumbuhan jamur tiram, menjaga kondisi lingkungan kumbung tetap optimal, serta meningkatkan efisiensi dan keakuratan penyiraman dibandingkan dengan metode konvensional, sehingga dapat mendukung optimasi pertumbuhan dan produktivitas jamur tiram.

1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut, didapatkan rumusan masalah sebagai berikut.

1. Bagaimana merancang dan membangun sistem penyiraman jamur tiram berbasis *Internet of Things* untuk memantau kelembapan kumbung secara *real-time*?
2. Bagaimana sistem tersebut mengendalikan proses penyiraman secara otomatis sesuai dengan kebutuhan pertumbuhan jamur tiram?
3. Bagaimana kinerja sistem dalam meningkatkan efisiensi penyiraman dan menjaga kondisi lingkungan agar tetap optimal dibandingkan dengan metode manual?

1.3 Tujuan

Berdasarkan rumusan masalah tersebut, didapatkan tujuan penelitian sebagai berikut.

1. Untuk merancang dan membangun sistem penyiraman jamur tiram berbasis *Internet of Things* yang mampu memantau dan mengendalikan kelembapan kumbung jamur secara otomatis.
2. Untuk mengimplementasikan sensor dan aktuator untuk mengatur proses penyiraman sesuai dengan kebutuhan pertumbuhan jamur tiram.
3. Untuk menguji kinerja sistem dalam menjaga kondisi lingkungan agar tetap optimal untuk pertumbuhan jamur tiram.

1.4 Manfaat

Berikut merupakan manfaat dari penelitian yang dapat diperoleh.

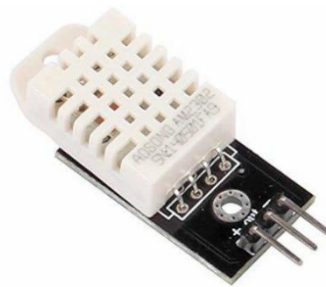
1. Membantu pembudidaya jamur tiram dalam mengelola proses penyiraman secara lebih efektif dan terkontrol.
2. Mengoptimalkan pertumbuhan dan kualitas hasil panen jamur tiram melalui pengaturan lingkungan yang sesuai.
3. Mengurangi beban kerja dan kesalahan manusia pada penyiraman manual.

BAB 2. TINJAUAN PUSTAKA

2.1 Internet of Things (IoT) dalam Pertanian

Perkembangan teknologi *Internet of Things* (IoT) telah mendorong inovasi di berbagai sektor, termasuk pertanian, untuk meningkatkan efisiensi, efektivitas, dan kemudahan pengelolaan. Menurut Sandi dan Fatma (2023) IoT dapat mempermudah pengawasan dan pengendalian sistem pertanian, sehingga hasil dan kualitas produksi dapat ditingkatkan secara signifikan. Penelitian ini bertujuan untuk mengoptimalkan pertumbuhan jamur melalui pengendalian kelembaban secara otomatis. Sistem ini memadukan monitoring lingkungan, penyiraman otomatis, dan notifikasi *real-time*, sehingga petani dapat mengatur kondisi pertumbuhan jamur tanpa harus berada di dalam kumbung secara langsung. Dengan pendekatan ini, diharapkan sistem dapat meningkatkan efektivitas dan efisiensi budidaya jamur tiram serta menjadi langkah awal penerapan teknologi IoT dalam mendukung modernisasi pertanian di Indonesia.

2.2 Sensor Suhu dan Kelembapan (DHT22)



Gambar 2.1 Sensor DHT22

Sensor suhu dan kelembapan merupakan komponen penting dalam sistem pemantauan lingkungan karena berfungsi untuk memperoleh data kondisi aktual secara *real-time* yang dibutuhkan dalam pengendalian lingkungan tumbuh. Data suhu dan kelembapan yang dihasilkan digunakan sebagai dasar pengambilan keputusan pada sistem penyiraman agar kondisi lingkungan dapat dijaga tetap optimal sesuai kebutuhan tanaman atau jamur. Sensor DHT22 adalah komponen yang sering digunakan dalam aplikasi pemantauan lingkungan termasuk pertanian, karena kemampuannya untuk mengukur suhu

dan kelembapan dengan akurasi yang baik (Yonatan *et al.*, 2025).

Tabel 2.1 Pin DHT22

No	Nama Pin	Fungsi	Keterangan
1.	Ground	Sebagai <i>ground</i> (massa)	Hubungkan ke GND Arduino/ESP
2.	Data	Mengirim data kelembapan	Mengirim data ke pin digital mikrokontroler
3.	VCC	Catu daya sensor	3.3V (ESP) atau 5V (Arduino)

a. GND (Ground)

Merupakan pin yang berfungsi sebagai jalur referensi negatif atau massa rangkaian. Pin ini melengkapi rangkaian listrik sensor dan memastikan aliran arus bekerja dengan stabil. Tanpa GND, sensor tidak akan bekerja dengan benar meskipun VCC terhubung.

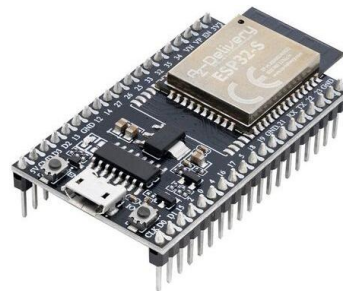
b. Data

Merupakan pin yang mengirimkan informasi kelembapan yang diukur oleh sensor ke mikrokontroler. Data ini dikirim dalam bentuk digital, sehingga mikrokontroler bisa membacanya secara langsung menggunakan *library* atau program khusus.

c. VCC

Merupakan pin catu daya yang memberikan tegangan listrik ke sensor agar rangkaian internalnya bisa bekerja. Tegangan ini biasanya 3,3V untuk ESP32 atau 5V untuk Arduino. Tanpa VCC, sensor tidak akan aktif dan tidak bisa mengukur suhu atau kelembapan.

2.3 Mikrokontroler (ESP32/NodeMCU)



Gambar 2.2 Mikrokontroler ESP32

Mikrokontroler berperan sebagai pusat kendali dalam sistem penyiraman

karena berfungsi untuk mengolah data yang diperoleh dari sensor serta mengendalikan aktuator sesuai dengan keputusan yang telah ditetapkan. Dalam sistem berbasis *Internet of Things* (IoT), mikrokontroler juga bertanggung jawab dalam proses komunikasi data antara perangkat keras dan platform pemantauan, sehingga memungkinkan pengawasan dan pengendalian sistem dilakukan secara *real-time*. Pemilihan mikrokontroler yang memiliki kemampuan konektivitas jaringan menjadi faktor penting dalam mendukung kinerja sistem secara keseluruhan. ESP32 NodeMCU merupakan salah satu mikrokontroler yang dilengkapi dengan modul WiFi dan Bluetooth 4.0. ESP32 NodeMCU yang dapat terhubung dengan WiFi digunakan sebagai mikrokontroler IoT (Pradito *et al.*, 2022).

2.3.1 Pin Daya (*Power*)

Tabel 2.2 Pin daya ESP32

Nama Pin	Fungsi	Keterangan
VIN	Input tegangan	Digunakan untuk memasukkan tegangan 5V dari adaptor atau sumber eksternal
5V	Tegangan 5V	Output 5V dari USB atau input ke rangkain
3V3	Tegangan 3.3V	Output 3.3 V untuk sensor dan modul
GND	Ground	Jalur refrensi negatif/massa rangkain

Pin daya pada ESP32 berfungsi untuk menyediakan tegangan bagi papan dan komponen yang terhubung. Pin 3V3 digunakan sebagai output tegangan 3,3V untuk sensor dan modul, sedangkan pin 5V dan VIN digunakan sebagai input tegangan 5V dari adaptor atau USB untuk menyalakan ESP32. Pin GND berfungsi sebagai jalur ground agar arus listrik dapat mengalir dengan baik dan sistem bekerja stabil.

2.3.2 Pin Digital (GPIO)

Tabel 2.3 Pin digital ESP32

Nama Pin	Fungsi	Keterangan
GPIO 0-39	I/O digital	Dapat digunakan sebagai input atau output digital
GPIO 34-39	<i>Input</i> saja	Hanya bisa membaca

GPIO 6-11	Tidak digunakan	data, tidak bisa output Dipakai internal untuk memori <i>flash</i> ESP32
-----------	-----------------	---

Pin ini bisa digunakan sebagai input atau output digital. Contoh sebagai input: membaca tombol atau sensor digital. Contoh sebagai output yaitu menyalakan *LED*, menggerakkan relay, atau menyalakan pompa. Beberapa pin hanya input. Pin tertentu tidak digunakan karena dipakai internal untuk memori (GPIO 6–11).

2.3.3 Pin Analog (ADC)

Tabel 2.4 Pin analog ESP32

Nama Pin	Fungsi	Keterangan
GPIO 0-39	Input analog	Membaca data sensor analog seperti kelembapan tanah
GPIO 34-39	ADC + DAC	Bisa sebagai input analog dan output analog

ESP32 bisa membaca sinyal analog dari sensor, misal kelembapan tanah. Pin ADC tersedia di GPIO 32–39 dan beberapa pin lain (GPIO 25–26). Beberapa pin juga bisa menjadi DAC, untuk output analog.

2.3.4 Pin DAC (*Output Analog*)

Tabel 2.5 Pin DAC ESP32

Nama Pin	Fungsi	Keterangan
GPIO 25	DAC <i>output</i>	Menghasilkan sinyal analog
GPIO 26	DAC <i>output</i>	Digunakan untuk kontrol analog

DAC (*Digital-to-Analog Converter*) adalah fitur yang memungkinkan ESP32 mengubah sinyal digital (0 dan 1) menjadi sinyal analog (tegangan variabel).

2.3.5 Pin Komunikasi

Tabel 2.6 Pin komunikasi ESP32

Nama Pin	Fungsi	Keterangan
GPIO 21	SDA	Jalur data komunikasi I2C
GPIO 22	SQL	Jalur <i>clock</i> komunikasi I2C
GPIO 5	CS	Selektor <i>slave</i> SPI
GPIO 18	SCK	<i>Clock</i> SPI
GPIO 23	MOSI	Data dari master ke <i>slave</i>
GPIO 19	MISO	Data dari <i>slave</i> ke master
GPIO 1	TX	<i>Transmit</i> data serial

GPIO 3	RX	Receive data serial
I2C → SDA (GPIO 21) & SCL (GPIO 22). Digunakan untuk modul seperti LCD I2C atau sensor I2C lain. SPI → GPIO 5 (CS), 18 (SCK), 23 (MOSI), 19 (MISO). Digunakan untuk modul memori, display, dan sebagainya. UART → TX (GPIO 1), RX (GPIO 3). Digunakan untuk komunikasi serial dengan komputer atau perangkat lain		

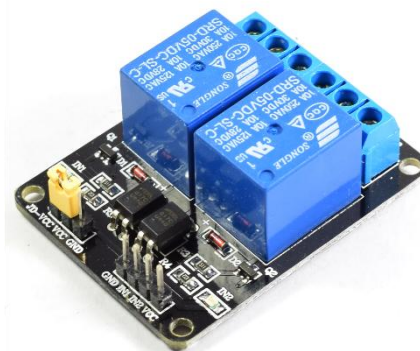
2.3.6 Pin Khusus

Tabel 2.7 Pin khusus ESP32

Nama Pin	Fungsi	Keterangan
EN	<i>Enable</i>	Untuk reset atau mengaktifkan ESP32
GPIO 0 (BOOT)	<i>Mode upload</i>	Digunakan saat upload program

Pin khusus pada ESP32 terdiri dari pin EN (*Enable*) dan GPIO 0 (BOOT). Pin EN berfungsi untuk mengaktifkan atau mereset ESP32, sehingga dapat digunakan untuk memulai ulang sistem. Sementara itu, pin GPIO 0 (BOOT) digunakan untuk masuk ke mode *upload* program, sehingga ESP32 dapat diprogram dari komputer.

2.4 Modul Relay



Gambar 2.3 Modul relay

Modul relay merupakan komponen elektronik yang berfungsi sebagai saklar otomatis untuk menyambung dan memutus aliran arus listrik dengan menggunakan sinyal kendali berdaya kecil dari mikrokontroler (Nurhuda dan Khoyi'in, 2024). Relay memungkinkan perangkat bertegangan dan berarus besar, seperti pompa air, dapat dikendalikan secara aman tanpa terhubung langsung dengan rangkaian kontrol. Modul relay memiliki dua kondisi kerja utama, yaitu *normally open* (NO) dan *normally close* (NC), dimana pada

kondisi NO kontak relay terbuka saat tidak aktif dan akan menutup ketika diaktifkan, sedangkan pada kondisi NC kontak relay tertutup saat tidak aktif dan akan terbuka ketika diaktifkan, sehingga relay dapat digunakan secara fleksibel sesuai kebutuhan sistem otomasi.

2.4.1 Pin Kontrol

Tabel 2.8 Pin kontrol relay

Nama Pin	Fungsi	Keterangan
VCC	Catu daya modul	Tegangan 5V/3.3V sesuai modul
GND	Ground	Ground dari mikrokontroler
IN	Sinyal kontrol	Pin dari ESP32/Arduino untuk mengaktifkan relay

VCC → Catu daya modul relay. Biasanya 5V. Memberi energi untuk menggerakkan kumparan relay sehingga bisa switch ON/OFF. GND → Ground modul relay. Harus terhubung ke GND mikrokontroler agar aliran listrik modul stabil. IN / IN1 / IN2... → Pin kontrol yang menerima sinyal dari mikrokontroler. Saat pin ini diberi HIGH atau LOW (tergantung tipe modul), relay akan mengaktifkan saklar internal untuk menyalakan perangkat.

2.4.2 Terminal Output Relay

Tabel 2.9 Pin output relay

Nama Pin	Fungsi	Keterangan
COM	Common	Titik tengah saklar <i>relay</i>
NO (<i>Normally Open</i>)	Terbuka saat <i>off</i>	Terhubung ke COM saat relay aktif
NC (<i>Normally Closed</i>)	Tertutup saat <i>off</i>	Terhubung ke COM saat relay mati

Bagian ini terhubung ke perangkat yang akan dikontrol biasanya ada 3 terminal:

- COM (*Common*) → Titik tengah saklar. Jalur utama dari sumber listrik ke beban.
- NO (*Normally Open*) → Terhubung ke COM saat relay aktif / ON. Digunakan untuk menyalakan perangkat saat relay ON.

- c. NC (*Normally Closed*) → Terhubung ke COM saat relay mati / OFF. Biasanya tidak dipakai untuk sistem penyiraman, kecuali ingin perangkat aktif saat relay OFF.

2.5 Stepdown LM2596



Gambar 2.4 Stepdown LM2596

Stepdown (*buck converter*) berperan sebagai penurun dan penstabil tegangan listrik dari sumber daya yang lebih tinggi ke tegangan yang lebih rendah sesuai kebutuhan komponen sistem. Pada sistem IoT penyiraman jamur tiram, stepdown digunakan untuk menurunkan tegangan dari sumber daya seperti adaptor atau baterai menjadi tegangan yang aman, misalnya 5V atau 3,3V, agar dapat digunakan oleh mikrokontroler ESP32, sensor, dan modul pendukung lainnya. Dengan adanya stepdown, sistem memperoleh suplai daya yang stabil, lebih efisien, serta mampu melindungi komponen elektronik dari kerusakan akibat tegangan berlebih (Putra *et al.*, 2024).

Tabel 2.10 Pin stepdown LM2596

Nama Pin	Fungsi	Keterangan
VIN +	Sebagai pin untuk menerima tegangan masuk dari sumber listrik.	Tegangan yang masuk biasanya lebih besar, misalnya 9V, 12V, atau 24V dari adaptor atau baterai.
VIN -	Sebagai jalur ground dari sumber tegangan.	Harus terhubung ke ground rangkaian agar sistem dapat bekerja dengan stabil.
VOUT +	Sebagai pin keluaran	Tegangan keluar bisa

	tegangan yang sudah diturunkan.	diatur (misalnya 5V atau 3,3V) untuk mensuplai mikrokontroler dan sensor.
VOUT -	Sebagai ground dari tegangan keluaran.	Dihubungkan ke ground ESP32, sensor, dan modul lain pada sistem.

2.6 Prinsip Kerja Sistem

Prinsip kerja sistem dimulai ketika sensor DHT22 mengukur kelembaban udara di dalam kumbung jamur secara berkala. Data hasil pengukuran tersebut dikirim ke mikrokontroler ESP8266/ESP32 untuk diproses dan dibandingkan dengan nilai *set point* yang telah ditetapkan sebagai kondisi lingkungan yang optimal bagi pertumbuhan jamur. Prinsip kerja pengambilan keputusan dilakukan oleh mikrokontroler berdasarkan hasil perbandingan data sensor dengan set point. Apabila nilai kelembaban berada di bawah batas minimum, sistem secara otomatis mengaktifkan penyiraman. Sebaliknya, jika kelembaban telah mencapai atau melebihi batas maksimum, sistem akan menghentikan penyiraman untuk menjaga kestabilan lingkungan kumbung jamur. Prinsip kerja aktuasi dilakukan melalui modul relay yang berfungsi sebagai saklar elektronik untuk mengendalikan pompa air. Pompa akan menyemprotkan kabut air ke dalam kumbung sehingga kelembaban meningkat secara merata. Kemudian prinsip kerja sistem IoT memungkinkan data kelembaban dan status pompa dikirim melalui jaringan WiFi ke platform cloud untuk pemantauan secara *real-time* dan pengendalian jarak jauh.

2.7 State of the Art

State of the Art (SOTA) merupakan kajian sistematis terhadap penelitian terdahulu untuk menggambarkan perkembangan terkini suatu topik penelitian. SOTA digunakan untuk menunjukkan posisi penelitian ini dibandingkan dengan penelitian sebelumnya melalui perbandingan metode, variabel, dan hasil penelitian. Analisis tersebut bertujuan untuk mengidentifikasi celah penelitian yang menjadi dasar pengembangan penelitian ini.

Tabel 2.11 State of the art penelitian

No	Judul Penelitian	Peneliti (Tahun)	Metode/ Teknologi	Persamaan	Perbedaan
1.	Penerapan Sistem Penyiraman Otomatis Berbasis IoT pada Budidaya Jamur Tiram <i>Application of IoT-Based Automatic Watering System in Oyster Mushroom Cultivation</i>	Wirmando <i>et al.</i> (2025)	IoT berbasis ESP8266, sensor DHT22, RTC, pompa air, monitoring Android	Menggunakan sensor DHT22, mikrokontroler berbasis IoT, serta sistem penyiraman otomatis untuk menjaga kondisi lingkungan tumbuh	Sistem masih menggunakan satu titik sensor dan belum dilakukan evaluasi kinerja sistem secara teknis.
2.	Implementasi IoT dan Teknologi Cerdas dalam Pengendalian Suhu dan Kelembaban Budidaya Jamur Tiram	Anwar <i>et al.</i> (2025)	Sistem pengendalian suhu dan kelembaban berbasis IoT dengan aplikasi mobile	Sama-sama memanfaatkan IoT untuk pengendalian suhu dan kelembaban kumbung jamur tiram	Penelitian bersifat pengabdian masyarakat, belum fokus pada evaluasi kinerja sistem secara teknis
3.	Rancang Bangun Sistem Penyiraman Jamur Tiram untuk Optimasi Pertumbuhan Berbasis <i>Internet of Things</i>	Maulana Iqbal Muhlisin (2026)	Menerapkan penyiraman otomatis berbasis IoT, penggunaan sensor kelembaban serta mikrokontroler ESP32	Sistem mampu mengendalikan penyiraman otomatis dan menjaga kelembaban optimal (70–90%)	Penelitian ini menambahkan monitoring <i>real-time</i> , pengujian kinerja sistem, dan optimasi kelembaban pada rentang 70–90%

BAB 3. METODE KEGIATAN

3.1 Waktu dan Tempat Penelitian

3.1.1 Waktu Penelitian

Pelaksanaan penelitian dilakukan dalam jangka waktu enam bulan, dimulai pada Januari 2026 dan berakhir pada Juni 2026. Periode penelitian tersebut digunakan untuk melaksanakan seluruh rangkaian kegiatan, mulai dari pengumpulan dan kajian pustaka, perancangan sistem penyiraman, perakitan serta pemrograman perangkat IoT, pengujian fungsional dan kinerja sistem, hingga pengolahan data dan penyusunan laporan akhir. Tahapan penelitian disusun secara terstruktur guna memastikan setiap proses berjalan sesuai dengan tujuan penelitian yang telah direncanakan.

3.1.2 Tempat Penelitian

Kegiatan penelitian ini dilaksanakan di Desa Sogaan, Kabupaten Probolinggo, Jawa Timur, tepatnya pada lokasi budidaya jamur tiram milik masyarakat setempat. Lokasi tersebut dipilih karena memiliki kondisi kumbung yang representatif untuk penerapan sistem penyiraman berbasis *Internet of Things* (IoT). Selain itu, lingkungan penelitian memungkinkan sistem diuji secara langsung dalam kondisi operasional nyata, sehingga efektivitas pengendalian kelembaban udara terhadap pertumbuhan jamur tiram dapat diamati dan dievaluasi secara menyeluruh. Pemilihan lokasi ini juga mendukung penerapan teknologi tepat guna yang sesuai dengan kebutuhan petani setempat.

3.2 Alat dan Bahan

Alat dan bahan yang digunakan dalam penelitian ini terdiri atas perangkat keras dan perangkat lunak yang mendukung perancangan serta pengujian sistem penyiraman berbasis *Internet of Things* (IoT). Berikut merupakan uraian lengkap alat dan bahan yang digunakan.

3.2.1 Alat

a. Perangkat Keras

Tabel 3.1 Nama alat perangkat keras

No	Nama Alat	Spesifikasi/Fungsi	Jumlah
1.	ESP32 Dev Module	Mikrokontroler <i>dual-core</i> dengan <i>Wi-Fi</i> dan <i>Bluetooth</i> terintegrasi, tegangan logika 3,3V, sebagai pengendali utama sistem IoT.	1 unit
2.	Sensor Suhu dan Kelembapan DHT22	Mengukur kelembapan udara, tegangan kerja 3,3–5V.	1 unit
3.	Power Supply DC 12 V	Sumber catu daya utama untuk pompa air dan sistem penyiraman.	1 unit
4.	Modul <i>Step-Down</i> LM2596	Modul penurun tegangan dari 12V DC menjadi 5V DC untuk ESP32, sensor, <i>relay</i> .	1 unit
5.	Modul <i>Relay 2 Channel</i> 5 V	Saklar elektronik untuk mengontrol pompa air 12V menggunakan sinyal dari ESP3.	1 unit
6.	Pompa Air DC 12 V	Digunakan untuk memompa air dari tandon menuju sistem <i>sprinkler</i> .	1 unit
7.	Nozzle <i>Sprinkler</i>	Berfungsi menyemprotkan air ke area penyiraman.	4 unit

b. Perangkat Lunak

Tabel 3.2 Nama alat perangkat lunak

No	Nama Alat	Spesifikasi/Fungsi	Jumlah
1.	Arduino IDE	Perangkat lunak untuk menulis, mengompilasi, dan mengunggah program ke ESP3	1
2.	Library DHT22	Library untuk membaca	1

		data kelembapan dari sensor DHT22.	
3.	<i>Library WiFi</i>	ESP32 <i>Library</i> untuk menghubungkan ESP32 ke jaringan <i>Wi-Fi</i> .	1
4.	<i>Supabase Database</i>	<i>Realtime Platform cloud</i> untuk penyimpanan dan monitoring data sistem penyiraman secara daring.	1

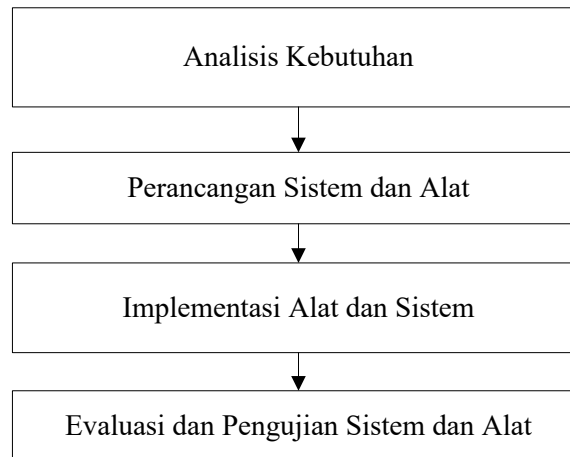
3.2.2 Bahan

Tabel 3.3 Nama bahan

No	Nama Alat	Spesifikasi/Fungsi	Jumlah
1.	Tandon air	Tempat penampungan air sebelum dialirkan ke sistem penyiraman.	1 unit
2.	Pipa air	Media penyalur air dari pompa menuju <i>nozzle</i> .	1 set
3.	PCB	Papan rangkaian yang berfungsi untuk menghubungkan dan menopang seluruh komponen supaya sistem bekerja secara rapi, stabil, dan aman.	1 set
4.	Kabel <i>Jumper</i>	Penghubung antar komponen (<i>male-male</i> dan <i>male-female</i>).	3 unit
5.	Kabel Daya DC	Kabel penyalur tegangan dari <i>power supply</i> ke pompa dan modul <i>step-down</i> .	1 unit
6.	<i>Cable Ties</i>	Pengikat kabel untuk menjaga kerapian dan keamanan instalasi rangkaian sistem.	1 <i>pack</i>

3.3 Metodologi Penelitian

Metodologi penelitian ini dilakukan secara berurutan dan bertahap dengan alur pelaksanaan penelitian sebagai berikut.



Gambar 3.1 Metodologi penelitian

3.3.1 Analisis Kebutuhan

Analisis kebutuhan merupakan tahap awal untuk mengidentifikasi permasalahan serta merumuskan kriteria sistem yang akan dikembangkan.

a. Analisis Permasalahan

Budidaya jamur tiram sangat dipengaruhi oleh kondisi suhu dan kelembaban udara di dalam kumbung. Kelembaban yang tidak terkontrol dapat menyebabkan pertumbuhan jamur menjadi kurang optimal, bahkan berpotensi menimbulkan kontaminasi apabila kondisi terlalu lembab. Proses penyiraman jamur tiram umumnya masih dilakukan secara manual sehingga intensitas dan waktu penyiraman sangat bergantung pada pengamatan serta ketelitian pekerja. Kondisi tersebut dapat menimbulkan ketidakkonsistenan dalam menjaga lingkungan tumbuh jamur tiram sehingga diperlukan sistem yang mampu melakukan pemantauan kondisi lingkungan secara *real-time* serta mengendalikan proses penyiraman secara otomatis sesuai kebutuhan pertumbuhan jamur tiram. Oleh karena itu, dirancang sistem penyiraman berbasis IoT yang dapat meningkatkan efisiensi penyiraman, menjaga kestabilan kelembaban, serta mengurangi ketergantungan terhadap metode manual.

b. Analisis Kebutuhan Fungsional Sistem

Kebutuhan fungsional sistem ditetapkan untuk memastikan bahwa sistem yang dirancang mampu memenuhi tujuan perancangan, yaitu melakukan pemantauan dan pengendalian kelembaban secara otomatis. Sistem mampu membaca data kelembaban udara di dalam kumbung jamur sebagai parameter utama dalam pengendalian penyiraman, kemudian memproses data tersebut melalui mikrokontroler sebagai pusat kendali untuk dibandingkan dengan nilai ambang batas yang telah ditentukan. Berdasarkan hasil pengolahan data, sistem dapat mengaktifkan dan menonaktifkan pompa air secara otomatis. Selain itu, sistem juga mampu mengirimkan informasi kelembaban serta status kerja pompa ke platform berbasis *cloud* sehingga pengguna dapat memantau kondisi kumbung melalui perangkat seluler atau web. Dengan demikian, seluruh fungsi sistem terintegrasi dalam satu kesatuan kerja yang mendukung budidaya jamur tiram secara lebih efisien dan terkontrol.

c. Analisis Kebutuhan Kinerja Sistem

Kebutuhan kinerja sistem berkaitan dengan kemampuan sistem dalam menjaga kondisi lingkungan agar tetap berada pada rentang optimal pertumbuhan jamur tiram, yaitu pada kelembaban 70%–90% RH sesuai hasil perancangan kalibrasi. Sistem mampu merespons perubahan kelembaban secara otomatis dan stabil apabila terjadi penyimpangan dari rentang tersebut, serta bekerja secara kontinu selama proses budidaya berlangsung dengan tingkat respons yang cukup cepat terhadap perubahan kondisi lingkungan. Selain itu, pembacaan sensor harus memiliki tingkat akurasi yang memadai sehingga keputusan pengaktifan maupun penghentian pompa air dapat dilakukan secara tepat. Dengan terpenuhinya kebutuhan kinerja tersebut, sistem diharapkan mampu meningkatkan efisiensi penggunaan air sekaligus menjaga kestabilan lingkungan kumbung secara berkelanjutan.

d. Analisis Kebutuhan Operasional dan Integrasi IoT

Sistem berbasis IoT memiliki kebutuhan operasional yang tidak hanya mencakup pengendalian secara lokal, tetapi juga integrasi dengan jaringan internet untuk mendukung proses monitoring. Sistem harus mampu terhubung dengan jaringan *Wi-Fi* agar data hasil pembacaan sensor dapat dikirimkan ke platform *cloud* secara *real-time* dan diakses oleh pengguna untuk memantau kondisi kelembaban dan status kerja pompa dari lokasi yang berbeda. Selain itu, sistem mampu beroperasi pada lingkungan kumbung dengan tingkat kelembaban tinggi sehingga diperlukan instalasi yang aman, stabil, dan tahan terhadap kondisi lembab. Suplai daya listrik juga harus terjaga agar seluruh komponen dapat bekerja secara optimal. Dengan terpenuhinya kebutuhan operasional dan integrasi IoT tersebut, sistem tidak hanya berfungsi sebagai alat penyiraman otomatis, tetapi juga menjadi sistem monitoring yang mendukung pengelolaan budidaya jamur tiram secara lebih modern dan terkontrol.

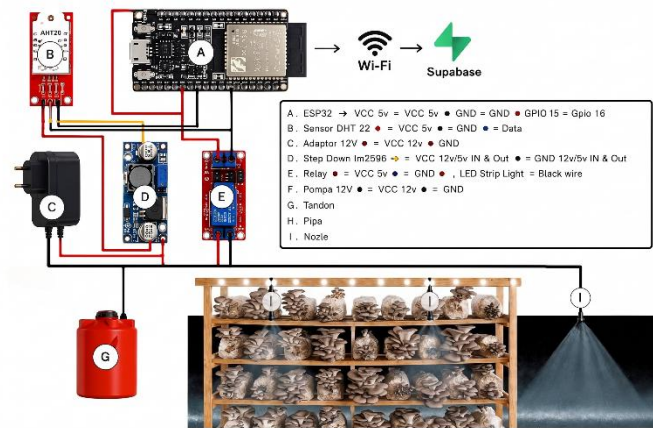
3.3.2 Perancangan Alat dan Sistem

Perancangan alat dan sistem dilakukan untuk menentukan alur kerja, komponen, dan hubungan antar perangkat dalam sistem penyiraman. Perancangan terdiri dari.

a. Skematik Sistem

Skematik sistem menggambarkan keseluruhan hubungan antara sensor, mikrokontroler, aktuator, dan platform IoT. Komponen utama yang digunakan meliputi sensor kelembaban (DHT22) untuk memonitor kondisi lingkungan, mikrokontroler (ESP32) sebagai pusat pengolahan data, pompa dan *solenoid valve* sebagai aktuator untuk penyiraman otomatis, *platform* IoT untuk monitoring dan kontrol jarak jauh. Setiap komponen saling terhubung dan bekerja secara terintegrasi untuk mendukung proses monitoring serta pengendalian kondisi lingkungan kumbung jamur. Melalui integrasi tersebut, sistem mampu melakukan pengambilan keputusan secara otomatis berdasarkan data yang diperoleh

dari sensor sehingga proses penyiraman dapat berlangsung lebih efektif dan efisien.



Gambar 3.2 Skematik Sistem

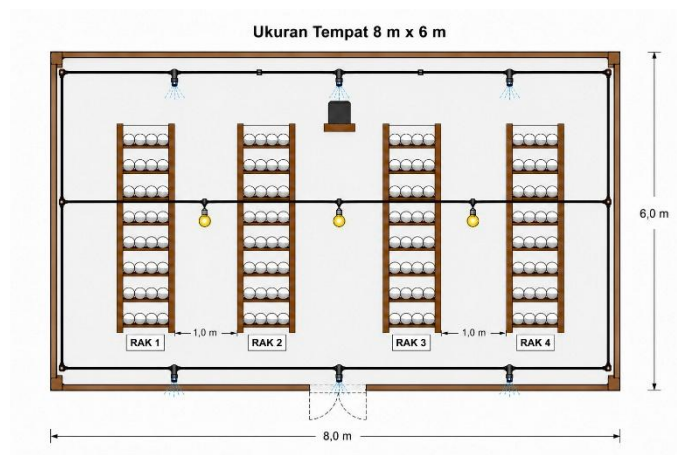
Berdasarkan Gambar 3.2 Skematik sistem menunjukkan rancangan sistem penyiraman jamur tiram berbasis IoT yang memanfaatkan integrasi sensor, mikrokontroler, aktuator, dan layanan komputasi awan untuk menjaga kondisi lingkungan kumbung jamur tetap optimal. Sensor DHT22 digunakan untuk mengukur kelembaban udara di dalam kumbung jamur secara berkala. Data hasil pengukuran tersebut dikirimkan ke mikrokontroler ESP32 sebagai pusat pengolahan dan pengendalian sistem. Nilai kelembaban yang diperoleh dibandingkan dengan ambang batas yang telah ditentukan sebagai kondisi optimal pertumbuhan jamur tiram.

ESP32 mengendalikan modul relay yang berfungsi sebagai saklar elektronik untuk mengaktifkan dan mematikan pompa air. Pompa air menyalurkan air dari tandon menuju pipa distribusi yang terhubung dengan beberapa *nozzle* kabut, sehingga proses penyiraman dilakukan dalam bentuk kabut halus guna meningkatkan kelembaban udara secara merata di dalam kumbung jamur. Modul *step-down* LM2596 digunakan untuk menurunkan dan menstabilkan tegangan dari sumber daya utama agar sesuai dengan kebutuhan kerja pompa dan komponen sistem lainnya, sehingga sistem dapat beroperasi secara stabil.

Data kelembaban udara serta status kerja pompa dikirimkan oleh ESP32 melalui koneksi Wi-Fi menuju platform *cloud Supabase* untuk

keperluan monitoring. Informasi tersebut dapat diakses secara *real-time* melalui aplikasi berbasis web atau perangkat seluler sehingga pengguna dapat memantau kondisi lingkungan kumbung jamur secara berkelanjutan. Rancangan sistem ini memungkinkan pengendalian penyiraman dilakukan secara otomatis berdasarkan kondisi aktual lingkungan, meningkatkan efisiensi penggunaan air, mendukung optimasi pertumbuhan jamur tiram berbasis IoT.

b. Tata Letak Penempatan Komponen Sistem pada Kumbung



Gambar 3.3 Tata letak penempatan komponen sistem

Berdasarkan gambar 3.3 kumbung jamur tiram dirancang dengan ukuran 8 m x 6 m yang di dalamnya terdapat empat rak baglog sebagai tempat media tanam jamur tiram yang kondisi kelembapannya harus dijaga pada rentang optimal 70–90%. Sistem penyiraman pada kumbung ini menggunakan enam *nozzle* yang terhubung melalui selang tanpa pipa, *nozzle* ini berfungsi sebagai keluaran dari pompa air yang dikendalikan oleh modul relay dan diprogram melalui mikrokontroler ESP32 berdasarkan hasil pembacaan sensor DHT22 terhadap kondisi kelembapan udara di dalam kumbung. Adapun box penempatan alat yang terletak di bagian belakang bawah kumbung merupakan lokasi penempatan perangkat keras sistem, seperti ESP32, modul relay, dan *stepdown* LM2596 yang berfungsi mengendalikan proses penyiraman secara otomatis maupun manual serta mengirimkan data kondisi lingkungan secara real-time ke platform monitoring berbasis *cloud Supabase* sehingga

kumbung pada gambar ini menjadi gambaran nyata dari lingkungan implementasi sistem penyiraman berbasis *Internet of Things*.

c. Kalibrasi Kelembapan Udara dan Penentuan Tindakan Sistem

Tabel 3.4 Kalibrasi Kelembapan Udara

Kondisi Kelembaban Udara	Nilai Sensor DHT22 (%RH)	Kategori Kelembaban	Tindakan Sistem
Sangat kering	<70%	Tidak optimal untuk pertumbuhan	Pompa menyala (penyiraman aktif)
Kering - lembab	70% - 80%	Mulai mendekati kondisi ideal	Pompa mati
Lembab (ideal)	80% - 90%	Kondisi optimal pertumbuhan jamur tiram	Pompa mati
Sangat lembab	>90%	Menyebabkan kontaminasi	Pompa mati (penerangan Aktif)

Berdasarkan Tabel 3.4 kondisi kelembaban diklasifikasikan berdasarkan nilai pembacaan sensor DHT22 yang digunakan sebagai acuan dalam pengendalian sistem penyiraman otomatis pada kumbung. Kelembapan di bawah 70% kondisi lingkungan dikategorikan sangat kering dan tidak optimal sistem mengaktifkan pompa untuk melakukan penyiraman guna meningkatkan kelembapan. Pada rentang 70%–80% kondisi mulai mendekati ideal, pada rentang 70%–90% merupakan kelembaban normal untuk jamur tiram (Rahman dan Muskhir, 2021).

Kelembaban udara pada rentang 80%–90% merupakan kondisi ideal yang optimal untuk pertumbuhan jamur tiram, sehingga pompa berada dalam kondisi mati karena tidak diperlukan penyiraman tambahan. Sementara itu, pada kelembaban di atas 90%, kondisi dikategorikan sangat lembab dan berisiko menyebabkan kontaminasi, sehingga sistem mematikan pompa sebagai langkah pencegahan. Pengaturan memastikan kelembaban berada dalam rentang optimal mendukung pertumbuhan jamur secara stabil.

3.3.3 Implementasi Alat dan Sistem

Perancangan alat sistem penyiraman jamur tiram berbasis *Internet of Things* (IoT) dirancang untuk mengontrol kelembaban media tanam secara otomatis agar pertumbuhan jamur lebih optimal. Sistem ini bekerja dengan memanfaatkan sensor, mikrokontroler dan aktuator yang saling terintegrasi.

a. Perakitan Mikrokontroler ESP32

Perakitan mikrokontroler ESP32 dilakukan sebagai pusat kendali sistem penyiraman jamur tiram berbasis *Internet of Things* (IoT). ESP32 berfungsi untuk menerima data dari sensor, memproses data serta mengendalikan aktuator dan komunikasi internet.

b. Pemasangan Sensor DHT22

Sensor DHT22 digunakan untuk mengukur kelembaban udara pada kumbung jamur tiram. Data dari sensor ini menjadi acuan utama dalam sistem penyiraman otomatis berbasis *Internet of Things* (IoT).

Koneksi pin sensor ke ESP32:

- a) VCC DHT22 → 3.3V ESP32
- b) DATA DHT22 → GPIO 4 ESP32
- c) GND DHT22 → GND ESP32
- d) Resistor 10 k Ω dipasang antara VCC dan DATA (jika DHT22 tanpa modul)

c. Pemasangan Modul Relay

Modul relay berfungsi untuk menghidupkan atau mematikan pompa air, *solenoid valve*, atau kipas. Dikendalikan otomatis oleh ESP32 berdasarkan data sensor.

Koneksi pin:

- a) VCC → 5V
- b) GND → GND ESP32
- c) IN → GPIO ESP32 (misalnya GPIO 26)

d. Perakitan Sistem Penyiraman Otomatis

Tujuan sistem penyiraman otomatis untuk mengendalikan penyiraman

secara otomatis berdasarkan kondisi lingkungan, sehingga kelembaban tetap optimal, penggunaan air lebih efisien, dan kebutuhan penyiraman manual dapat dikurangi.

Pin penyiraman otomatis

- a) COM (*Common*)
- b) NO (*Normally Open*)
- c) NC (*Normally Closed*)

e. Pemasangan Catu Daya

Output tegangan 5 volt dari modul *step-down* digunakan untuk menyuplai daya ke mikrokontroler ESP32. Tegangan 5 volt juga digunakan untuk mengoperasikan modul relay sebagai pengendali pompa air. Selain itu output 5 volt dimanfaatkan untuk mencatu sensor-sensor yang digunakan dalam sistem penyiraman otomatis.

3.3.4 Evaluasi dan Pengujian Alat dan Sistem

Sistem penyiraman otomatis dapat bekerja dengan baik, seluruh komponen berfungsi sesuai perancangan serta sistem mampu merespons perubahan kelembaban dengan mengaktifkan dan menonaktifkan pompa air secara otomatis. Pengujian sistem dilakukan untuk memastikan seluruh komponen sistem penyiraman otomatis berfungsi dengan baik dan sesuai perancangan, dimulai dari pengecekan catu daya untuk memastikan tegangan 12V dan keluaran 5V dari modul *step-down* stabil, dilanjutkan dengan pengujian ESP32 dalam menjalankan program, pengujian sensor dalam membaca kondisi lingkungan, serta pengujian modul relay dan pompa air untuk memastikan proses penyiraman dapat berjalan otomatis sesuai dengan kondisi kelembaban yang ditentukan.

a. Tujuan Pengujian

Berikut merupakan tujuan dari pengujian alat.

- a) Mengumpulkan penerapan sistem penyiraman otomatis berbasis iot.
- b) Meningkatkan efisiensi penggunaan air dalam proses penyiraman.
- c) Mengurangi ketergantungan pada penyiraman manual.

- d) Mendukung pertumbuhan tanaman atau jamur agar lebih optimal dan terkontrol.

b. Pengujian Catu Daya

Berikut merupakan pengujian dari catu daya.

- a) Mengukur tegangan aku sebagai sumber daya utama untuk memastikan berada pada kisaran 12 volt.
- b) Mengukur tegangan keluaran modul *step-down* untuk memastikan menghasilkan tegangan 5 volt.
- c) Memastikan tegangan 5 volt dapat menyuplai ESP32, modul relay dan sensor dengan stabil.
- d) Menarik kesimpulan bahwa sistem catu daya berfungsi dengan baik dan mendukung kinerja seluruh sistem.

c. Pengujian Sensor Kelembapan

Berikut merupakan pengujian dari sensor kelembapan

- a) Menghubungkan sensor kelembapan ke ESP32 dan memastikan sensor mendapatkan suplai tegangan yang sesuai.
- b) Menyalakan sistem dan mengamati data kelembapan yang ditampilkan pada serial monitor atau aplikasi.
- c) Mengubah kondisi lingkungan (lembab dan kering) untuk melihat respons perubahan nilai sensor.
- d) Membandingkan hasil pembacaan sensor dengan kondisi lingkungan sebenarnya.
- e) Menyimpulkan bahwa sensor kelembapan dapat bekerja dengan baik apabila nilai yang ditampilkan berubah sesuai kondisi lingkungan.

d. Pengujian Modul Relay

Berikut merupakan pengujian pada modul relay.

- a) Menghubungkan modul relay ke ESP32 dan memastikan suplai tegangan serta koneksi pin sudah sesuai.
- b) Mengunggah program uji untuk mengaktifkan dan menonaktifkan relay melalui GPIO ESP32.

- c) Mengamati indikator LED pada modul relay sebagai tanda relay aktif atau nonaktif.
 - d) Menghubungkan Pompa air ke terminal relay dan melakukan pengujian *on* atau *off*.
 - e) Menyimpulkan bahwa modul relay berfungsi dengan baik apabila mampu mengendalikan pompa sesuai perintah ESP32.
- e. Pengujian Pompa Air
- Berikut merupakan pengujian pompa air.
- a) Menghubungkan pompa air ke sumber daya dan modul relay sesuai dengan rangkaian yang telah dirancang .
 - b) Menyalakan sistem dan memberikan perintah *ON* melalui ESP32.
 - c) Mengamati apakah pompa air menyala dan mengalirkan air dengan normal.
 - d) Memberikan perintah *OFF* untuk memastikan pompa berhenti bekerja.
 - e) Menyimpulkan bahwa pompa air berfungsi dengan baik apabila dapat menyala dan mati sesuai perintah sistem tanpa gangguan.
- f. Pengujian Secara Keseluruhan
- Berikut merupakan pengujian sistem secara keseluruhan.
- a) Menyalakan sistem penyiraman otomatis dan memastikan seluruh komponen mendapatkan catu daya dengan baik.
 - b) Memastikan ESP32, sensor, modul relay dan pompa air terhubung serta berfungsi sesuai perencanaan air terhubung serta berfungsi sesuai perencanaan.
 - c) Mensimulasikan kondisi kelembaban rendah untuk melihat respons sistem dalam mengaktifkan pompa air.
 - d) Mensimulasikan kondisi kelembaban tinggi untuk memastikan pompa air berhenti secara otomatis.
 - e) Mengamati kestabilan kerja sistem selama proses pengujian untuk memastikan tidak terjadi gangguan atau kesalahan fungsi pada setiap komponen.

3.4 Jadwal Pelaksanaan

Berikut merupakan jadwal pelaksaannya yang akan dilaksanakan dalam penelitian ini.

Tabel 3.5 Jadwal pelaksanaan

No.	Metode Kegiatan	Bulan Ke					
		1	2	3	4	5	6
1.	Analisis Kebutuhan						
2.	Perancangan Sistem dan Alat						
3.	Implementasi Alat dan Sistem						
4.	Evaluasi dan Pengujian Sistem dan Alat						

BAB 4. HASIL DAN PEMBAHASAN

4.1 Analisis Kebutuhan

Sistem otomatis berbasis Internet of Things telah memenuhi kebutuhan yang telah ditentukan pada tahap analisis. Sistem mampu melakukan monitoring kelembapan udara di dalam kumbung, mengendalikan pompa air secara otomatis maupun manual, serta mengirimkan data ke Supabase secara real-time. Selain itu, perangkat keras dan perangkat lunak yang digunakan dapat terintegrasi dengan baik sehingga sistem dapat berjalan sesuai dengan tujuan penelitian.

Berdasarkan hasil implementasi dan pengujian yang telah dilakukan, setiap fungsi utama sistem mulai dari pembacaan data sensor, pengiriman data ke basis data, hingga pengendalian pompa berjalan sesuai rancangan tanpa kendala berarti. Hal ini menunjukkan bahwa sistem mampu mendukung proses penyiraman jamur tiram secara lebih efektif, sekaligus memudahkan pengguna dalam melakukan pemantauan dan pengendalian melalui aplikasi.

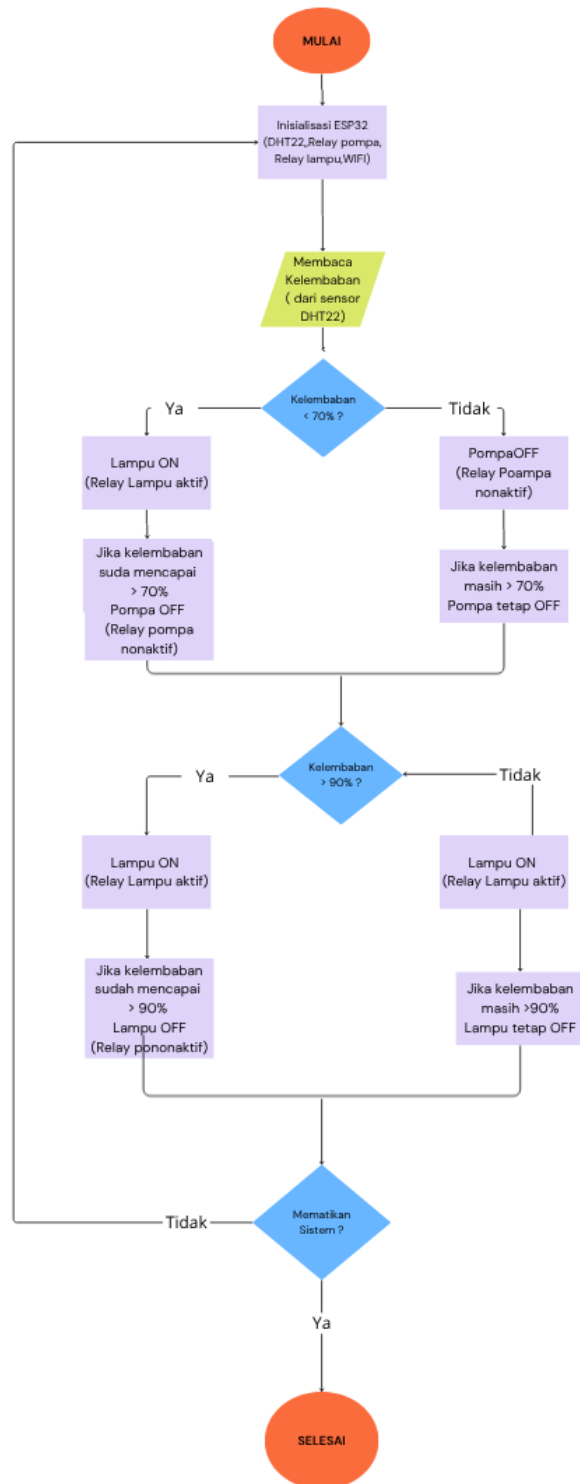
Sistem juga mampu memberikan informasi kondisi kelembapan kumbung secara cepat sehingga pengguna dapat mengetahui kondisi lingkungan budidaya tanpa harus melakukan pengecekan secara langsung. Penggunaan Supabase sebagai media penyimpanan data memungkinkan seluruh data hasil monitoring tersimpan secara terpusat dan dapat diakses kembali saat diperlukan. Fitur pengendalian pompa secara manual memberikan fleksibilitas kepada pengguna ketika diperlukan penyiraman di luar mode otomatis. Dengan demikian, hasil analisis kebutuhan membuktikan bahwa sistem yang dirancang telah memenuhi kebutuhan fungsional serta layak diterapkan sebagai solusi penyiraman jamur tiram berbasis IoT.

Selain memenuhi kebutuhan fungsional, sistem ini juga menunjukkan kinerja yang stabil, di mana data dari sensor dapat diproses dan ditampilkan pada aplikasi tanpa keterlambatan yang signifikan, menandakan komunikasi antara perangkat dan basis data berjalan dengan baik. Keandalan sistem ini diharapkan dapat membantu pengguna menjaga kondisi kelembapan kumbung agar tetap sesuai dengan kebutuhan pertumbuhan jamur tiram, sehingga berpotensi diterapkan sebagai solusi penyiraman otomatis yang lebih efisien dan praktis.

4.2 Perancangan Sistem dan Alat

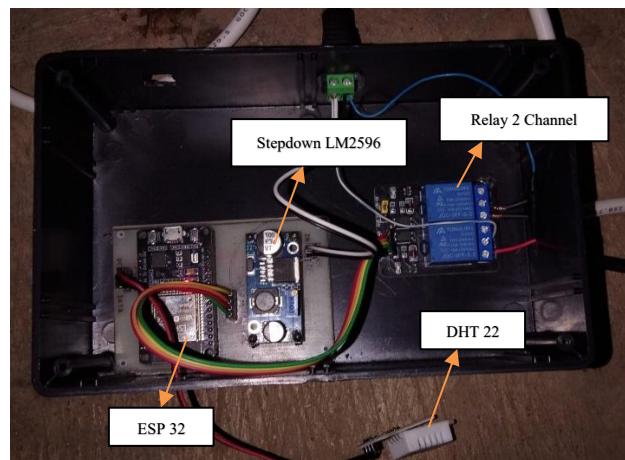
4.2.1 Perancangan Sistem

a) Flowchart



Gambar 4.1 Flowchart Sistem

4.2.2 Perancangan Alat



Gambar 4.2 Perancangan alat

Berdasarkan Gambar 4.2 dapat dijelaskan bahwa perangkat tersebut merupakan hasil perakitan alat sistem penyiraman jamur tiram berbasis *Internet of Things* yang terdiri dari empat komponen utama yang dirakit dalam satu kotak supaya menjadi satu kesatuan sistem yang ringkas. Komponen tersebut meliputi ESP32 yang berfungsi sebagai mikrokontroler utama sekaligus modul komunikasi Wi-Fi untuk memproses data dan menghubungkan sistem ke jaringan internet, *Stepdown* LM2596 yang berfungsi menurunkan tegangan input agar sesuai dengan kebutuhan daya komponen lain sehingga rangkaian bekerja secara stabil, Relay 2 Channel yang berfungsi sebagai saklar elektronik untuk mengontrol perangkat berdaya lebih besar seperti pompa air, serta sensor DHT22 yang berfungsi mendeteksi suhu dan kelembapan di area budidaya jamur tiram.

Keempat komponen tersebut saling terhubung melalui kabel-kabel penghubung sehingga dapat bekerja secara terintegrasi dan otomatis. Sensor DHT22 secara terus-menerus membaca kondisi suhu dan kelembapan lingkungan, kemudian data tersebut dikirim ke ESP32 untuk diolah dan dijadikan acuan pengambilan keputusan. Berdasarkan data tersebut, ESP32 akan mengirimkan sinyal ke Relay 2 Channel untuk mengaktifkan atau mematikan pompa air sesuai kebutuhan penyiraman. Dengan susunan alat seperti yang terlihat pada gambar, proses penyiraman jamur tiram dapat berjalan secara otomatis tanpa perlu campur tangan manual secara terus-menerus. Selain itu karena ESP32 memiliki fitur konektivitas internet, sistem ini juga dapat dipantau dan dikendalikan dari jarak jauh

sehingga sejalan dengan konsep penerapan teknologi IoT dalam mendukung proses budidaya jamur tiram yang lebih praktis dan efisien.

4.3 Implementasi Alat dan Sistem

Hasil rancang bangun sistem penyiraman jamur tiram berbasis *Internet of Things* (IoT) diperoleh melalui tahapan analisis kebutuhan, perancangan, implementasi, serta pengujian sistem. Sistem dikembangkan sebagai solusi terhadap proses penyiraman jamur tiram yang masih dilakukan secara manual, sehingga pengendalian suhu dan kelembapan di dalam kumbung sering kali kurang konsisten. Kondisi lingkungan yang tidak stabil dapat memengaruhi pertumbuhan jamur tiram karena kelembapan dan suhu merupakan faktor penting yang menentukan kualitas pertumbuhan dan produktivitas jamur (Arsella *et al.*, 2023).

Sistem yang dirancang menggunakan mikrokontroler ESP32 sebagai pusat kendali utama yang terintegrasi dengan sensor DHT22 untuk membaca kelembapan udara di dalam kumbung jamur secara *real-time*. Data hasil pembacaan sensor diproses oleh ESP32 dan dibandingkan dengan nilai ambang batas kelembapan yang telah ditentukan berdasarkan kebutuhan pertumbuhan jamur tiram. Apabila kelembapan berada di bawah kondisi optimal, sistem secara otomatis mengaktifkan pompa air melalui modul relay untuk melakukan penyiraman dalam bentuk kabut menggunakan *nozzle*. Sebaliknya, ketika kelembapan telah mencapai rentang ideal, sistem menghentikan proses penyiraman secara otomatis.

Rancangan sistem tidak hanya menerapkan sistem kendali otomatis, tetapi juga memanfaatkan teknologi IoT melalui koneksi *Wi-Fi* dan *platform Supabase* sebagai media penyimpanan serta monitoring data secara *real-time*. Informasi berupa kelembapan dan status pompa dikirim menuju *cloud* sehingga pengguna dapat melakukan pemantauan kondisi lingkungan kumbung dari lokasi yang berbeda. Integrasi perangkat keras dan perangkat lunak tersebut membentuk sistem penyiraman otomatis yang mampu menjaga kestabilan kondisi lingkungan kumbung agar tetap berada pada rentang optimal pertumbuhan jamur tiram.

Hasil implementasi menunjukkan bahwa seluruh komponen yang digunakan, meliputi ESP32, sensor DHT22, modul *relay*, pompa air, *step-down*

LM2596, serta sistem monitoring berbasis IoT, dapat terhubung dan bekerja sesuai rancangan. Integrasi sistem tersebut mendukung proses pengendalian penyiraman otomatis sehingga mampu mengurangi ketergantungan terhadap penyiraman manual serta meningkatkan efisiensi pemantauan kondisi lingkungan kumbung jamur tiram (Jannah *et al.*, 2026).

4.3.1 Implementasi Alat

Realisasi perangkat keras sistem penyiraman otomatis dilakukan berdasarkan rancangan skematik sistem yang telah disusun pada tahap perancangan alat dan sistem. Tahap ini bertujuan untuk mengintegrasikan seluruh komponen utama agar dapat bekerja sebagai satu kesatuan sistem yang mampu melakukan monitoring kelembapan serta pengendalian penyiraman secara otomatis. Perakitan dilakukan dengan menghubungkan komponen masukan (*input*), pengendali (*controller*), aktuator (*output*) serta sistem komunikasi berbasis IoT sehingga proses penyiraman dapat berjalan sesuai kebutuhan pertumbuhan jamur tiram. Perangkat keras yang digunakan dalam sistem ini terdiri atas mikrokontroler ESP32 *Dev Module*, sensor DHT22, modul relay 2 *channel*, pompa air DC 12 *volt*, modul *step-down* LM2596, *nozzle* serta komponen pendukung seperti selang distribusi dan kabel penghubung. Seluruh komponen tersebut dirangkai berdasarkan hubungan *input–proses–output* agar sistem dapat bekerja secara terintegrasi dan sesuai dengan fungsinya.

ESP32 digunakan sebagai pusat kendali utama yang bertugas menerima data dari sensor DHT22, mengolah hasil pembacaan, serta mengendalikan aktuator berdasarkan parameter yang telah ditentukan. ESP32 dipilih karena telah dilengkapi modul *Wi-Fi* terintegrasi sehingga mendukung penerapan sistem berbasis IoT dan monitoring secara *real-time* (Suherman *et al.*, 2025). Sensor DHT22 dipasang di dalam kumbung jamur sebagai komponen masukan yang membaca kelembapan udara secara berkala, kemudian data tersebut dikirim ke ESP32 untuk diproses sesuai kondisi lingkungan aktual.

Modul relay berfungsi sebagai saklar elektronik yang menghubungkan dan memutus aliran listrik menuju pompa air berdasarkan perintah ESP32. Ketika kelembapan berada di bawah batas minimum yang ditentukan, relay akan

mengaktifkan pompa air DC 12 volt untuk mengalirkan air menuju *nozzle* kabut, sehingga air yang keluar berbentuk butiran halus dan penyebaran kelembapan di dalam kumbung menjadi lebih merata. Sementara itu modul *step-down* LM2596 berfungsi menurunkan tegangan sumber daya utama menjadi tegangan kerja yang sesuai sehingga ESP32, sensor DHT22, dan relay dapat bekerja secara optimal dan stabil tanpa gangguan akibat tegangan berlebih. Dengan demikian, kestabilan tegangan yang dijaga oleh modul ini turut mendukung keandalan sistem secara keseluruhan dalam jangka panjang (Hidayah *et al.*, 2024).



Gambar 4.3 Gambar alat

Berdasarkan Gambar 4.3 hasil realisasi menunjukkan bahwa sebagian besar komponen sistem IoT ditempatkan menjadi satu kesatuan di dalam *box control*. Komponen yang ditempatkan di dalam *box* meliputi ESP32, modul *relay*, modul *step-down* LM2596, serta rangkaian pendukung sistem IoT lainnya. Penempatan seluruh komponen tersebut dalam satu *box* bertujuan menjaga kerapian instalasi, mempermudah proses pemeliharaan, serta melindungi komponen elektronik dari kondisi lingkungan kumbung yang memiliki tingkat kelembapan tinggi. Pompa air tidak ditempatkan di dalam *box control*, melainkan dipisahkan dari rangkaian utama karena pompa yang digunakan merupakan pompa celup (*submersible pump*) yang bekerja secara langsung di dalam media air. Pemisahan tersebut dilakukan untuk menghindari risiko kerusakan pada komponen elektronik akibat kontak langsung dengan air serta menjaga keamanan sistem selama proses pengoperasian.

Air dari pompa dialirkan melalui selang distribusi yang terhubung menuju area kumbung jamur. Selang dipasang pada sisi-sisi area budidaya dan dihubungkan dengan *nozzle* penyemprot untuk mendistribusikan kabut air secara merata. Selain

sistem penyiraman, lampu dipasang pada sisi-sisi kumbung sebagai komponen pendukung sistem untuk membantu kondisi lingkungan budidaya agar tetap sesuai kebutuhan pertumbuhan jamur tiram. Hasil perakitan menunjukkan bahwa seluruh komponen dapat terhubung dan bekerja sesuai fungsinya masing-masing sehingga sistem mampu mendukung proses monitoring dan pengendalian penyiraman otomatis berbasis IoT pada budidaya jamur tiram.



Gambar 4.4 Kumbung jamur tiram

4.3.2 Implementasi Sistem

Realisasi perangkat lunak sistem dilakukan untuk mendukung proses monitoring dan pengendalian penyiraman otomatis pada budidaya jamur tiram berbasis IoT. Perangkat lunak berperan dalam pembacaan data sensor, pengolahan data, pengambilan keputusan, pengendalian aktuator serta pengiriman data ke *platform cloud*. Implementasi dilakukan menggunakan Arduino IDE untuk membuat program yang diunggah ke ESP32.

Program disusun mengikuti alur kerja sistem mulai dari inisialisasi perangkat, koneksi jaringan internet, pembacaan data sensor hingga pengendalian penyiraman otomatis. ESP32 membaca kelembapan udara melalui sensor DHT22 secara berkala, kemudian membandingkan hasil pembacaan dengan ambang batas kondisi optimal pertumbuhan jamur tiram. Beberapa *library* digunakan untuk mendukung proses ini yaitu *library* DHT untuk pembacaan sensor, *WiFi library* untuk koneksi internet, dan *Supabase library* untuk mengirim data kelembapan dan status pompa ke sistem monitoring berbasis *cloud*. Integrasi *library* tersebut memungkinkan sistem bekerja secara otomatis dan mendukung monitoring kondisi kumbung secara *real-time* (Jumadil, 2025).

Kode program 4.1 *Control Logic*

```

1) // ===== CONTROL LOGIC =====
2) if (isAutoMode) {
3)   // Logic: <70% Pump ON, Lamp OFF | >90% Lamp ON, Pump OFF
4)   | 70-90% NORMAL (OFF/OFF)
5)   if (kelembapan < 70.0) {
6)     digitalWrite(RELAY1, RELAY_ONgun); // Pompa Hidup
7)     digitalWrite(RELAY2, RELAY_OFF); // Lampu Mati
8)     statusRelay = "KERING / PENYIRAMAN";
9)   }
10)  else if (kelembapan > 90.0) {
11)    digitalWrite(RELAY1, RELAY_OFF); // Pompa Mati
12)    digitalWrite(RELAY2, RELAY_ON); // Lampu Nyala
13)    statusRelay = "TERLALU LEMBAB";
14)  }
15)  else {
16)    digitalWrite(RELAY1, RELAY_OFF); // Pompa Mati
17)    digitalWrite(RELAY2, RELAY_OFF); // Lampu Mati
18)    statusRelay = "NORMAL";
19)  }
20) } else {
21)   // Manual Mode (App Based)
22)   digitalWrite(RELAY1, manualPump ? RELAY_ON : RELAY_OFF);
23)   digitalWrite(RELAY2, manualLight ? RELAY_ON : RELAY_OFF);
24)   statusRelay = "MANUAL";
25) }

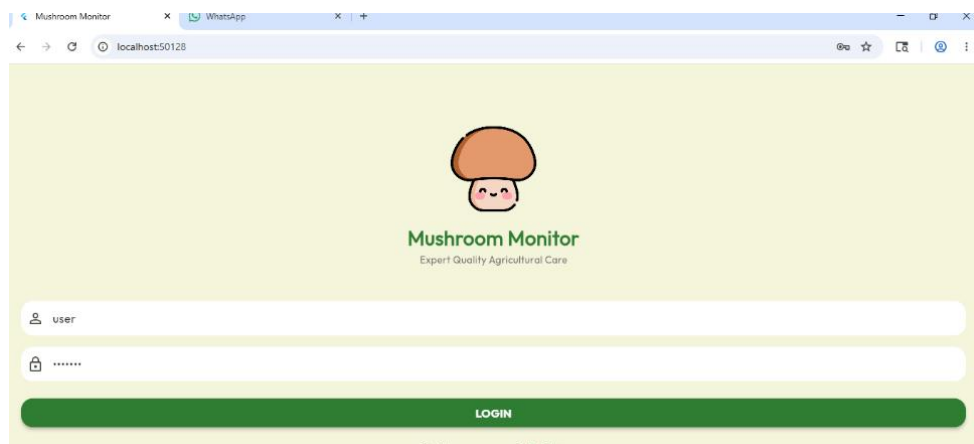
```

Berdasarkan potongan program tersebut, sistem menerapkan dua mode operasi yaitu otomatis dan manual. Pada mode otomatis nilai kelembapan menjadi parameter utama pengambilan keputusan, jika kelembapan di bawah 70% pompa aktif dan lampu mati (kondisi kering), jika kelembapan di atas 90% pompa mati dan lampu menyala (kondisi terlalu lembab), sedangkan pada rentang 70–90% kedua relay berada pada kondisi *off* karena lingkungan sudah normal (Rahman dan Muskhir, 2021). Selain mode otomatis, program juga menyediakan mode manual yang memungkinkan pengguna mengendalikan pompa dan lampu langsung melalui aplikasi tanpa bergantung pada pembacaan sensor.

Realisasi perangkat lunak juga mencakup pembuatan aplikasi monitoring berbasis *mobile* yang berfungsi sebagai antarmuka pengguna dalam memantau kondisi lingkungan kumbung serta mengendalikan sistem penyiraman secara manual. Pembuatan aplikasi ini menggunakan Antigravity sebagai IDE resmi untuk mendukung proses perancangan antarmuka (UI) yang lengkap serta penulisan kode program aplikasi sebelum diterapkan pada perangkat sesungguhnya.

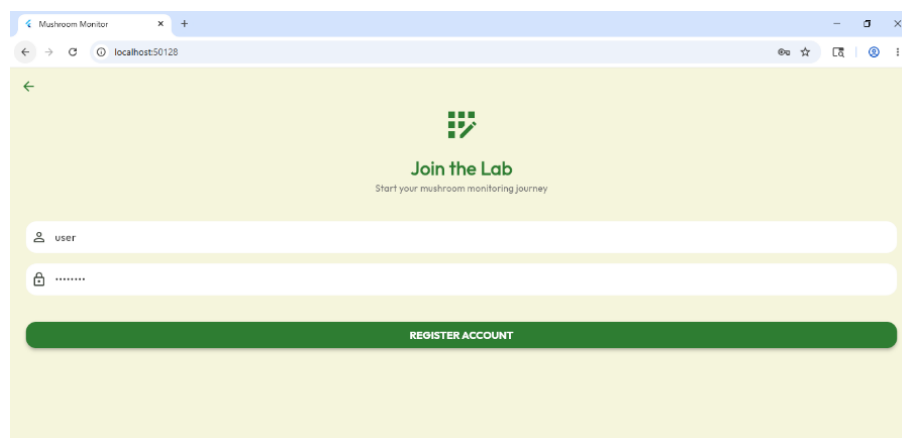


Gambar 4.5 Logo antigravity

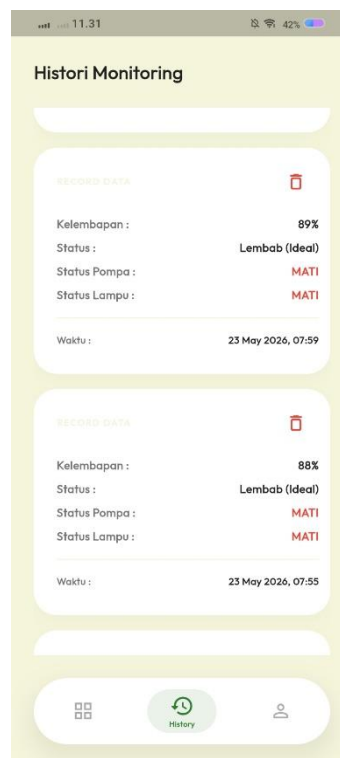


Gambar 4.6 Halaman login

Pada Gambar 4.6 menunjukkan tampilan halaman login pada aplikasi *Mushroom Monitor*. Halaman ini merupakan tampilan awal yang muncul saat aplikasi dijalankan, dimana pengguna diminta memasukkan *username* dan *password* pada kolom yang tersedia sebelum dapat mengakses fitur monitoring dan pengendalian sistem.

Gambar 4.7 Halaman *register*

Pada Gambar 4.7 menunjukkan tampilan halaman *register* aplikasi *Mushroom Monitor*. Halaman ini digunakan oleh pengguna baru untuk membuat akun dengan mengisi *username* dan *password* sebelum akun dapat digunakan untuk melakukan *login*. Proses pendaftaran ini bertujuan untuk memastikan setiap pengguna memiliki akses yang aman dan terverifikasi terhadap sistem monitoring dan pengendalian penyiraman otomatis pada budidaya jamur tiram.



Gambar 4.8 Halaman histori

Pada Gambar 4.8 menunjukkan tampilan halaman histori monitoring yang menampilkan data hasil pembacaan sensor berupa nilai kelembapan, status kondisi lingkungan, status pompa, status lampu serta waktu pengambilan data. Data-data tersebut ditampilkan secara berurutan berdasarkan waktu pencatatan, sehingga pengguna dapat dengan mudah memantau perubahan kondisi kumbung jamur dari waktu ke waktu. Selain itu, terdapat pula ikon tempat sampah pada setiap data yang berfungsi untuk menghapus riwayat data tertentu apabila pengguna ingin mengelola data histori yang tersimpan.



Gambar 4.9 Halaman *dashboard*

Pada Gambar 4.9 menunjukkan tampilan *dashboard* utama pada aplikasi *Mushroom Monitor*. Halaman ini menampilkan nilai kelembapan secara *real-time* dalam bentuk indikator lingkaran, beserta status kondisi lingkungan berupa keterangan "Sangat Lembab" dengan peringatan potensi kontaminasi. *Dashboard* juga menyediakan dua mode pengoperasian, yaitu mode otomatis dan manual serta menampilkan status aktual dari pompa air dan lampu pemanas.

4.3.3 Implementasi Sistem Monitoring Berbasis IoT

Implementasi sistem monitoring berbasis IoT dilakukan untuk mendukung proses pemantauan kondisi lingkungan kumbung jamur tiram secara *real-time*, sehingga pengguna dapat mengetahui kondisi kelembapan serta status kerja pompa tanpa melakukan pengamatan langsung di lokasi budidaya. Penerapan sistem ini mendukung pengendalian lingkungan yang lebih efektif sehingga kondisi pertumbuhan jamur tiram dapat dipantau secara berkelanjutan.

Implementasi monitoring memanfaatkan koneksi Wi-Fi pada ESP32 yang terhubung dengan *platform Supabase* sebagai media penyimpanan dan pertukaran

data berbasis *cloud*. Proses diawali ketika sensor DHT22 membaca kondisi kelembapan udara di dalam kumbung, kemudian data tersebut diproses oleh ESP32 dan dikirimkan secara otomatis menuju *Supabase*. Data yang tersimpan selanjutnya ditampilkan pada media monitoring sehingga pengguna dapat mengakses informasi kondisi lingkungan secara *real-time*. Selain sebagai media penyimpanan, *Supabase* juga berfungsi sebagai penghubung antara sistem perangkat keras dengan tampilan monitoring yang digunakan pengguna.



Gambar 4.10 Tampilan dashboard aplikasi mobile

Q	Filter by id, kelembapan, status... or ask AI	Sorted by 1 rule	RLS disabled	Role postgres		
	id	kelembapan ru...	status text	tanggal_upload timestamp	status_lampu text	status_pompa text
	96	81.20	MANUAL	2026-05-21 14:42:13	MENYALA - MANUAL	MATI
	97	81.20	MANUAL	2026-05-21 14:42:16	MATI	MATI
	98	81.60	MANUAL	2026-05-21 14:49:22	MENYALA - MANUAL	MATI
	99	81.90	Lembab (ideal)	2026-05-21 14:49:36	MENYALA - MANUAL	MATI
	100	81.40	MANUAL	2026-05-21 14:50:06	MATI	MATI
	101	88.20	MANUAL	2026-05-21 19:10:27	MENYALA - MANUAL	MATI
	102	88.10	MANUAL	2026-05-21 19:10:40	MATI	MATI
	103	88.20	MANUAL	2026-05-21 19:10:42	MENYALA - MANUAL	MATI
	104	88.20	MANUAL	2026-05-21 19:10:44	MATI	MATI
	105	88.20	MANUAL	2026-05-21 19:10:46	MENYALA - MANUAL	MATI
	106	88.10	MANUAL	2026-05-21 19:10:48	MATI	MATI
	107	88.10	MANUAL	2026-05-21 19:11:04	MENYALA - MANUAL	MATI
	108	88.10	MANUAL	2026-05-21 19:11:12	MATI	MATI
	109	95.50	TERLALU LEMBAB	2026-05-21 19:15:04	MENYALA - OTOMATIS	MATI
	110	89.70	NORMAL	2026-05-21 19:15:29	MATI	MATI
	111	90.70	TERLALU LEMBAB	2026-05-21 21:25:16	MENYALA - OTOMATIS	MATI

Gambar 4.11 Tampilan database

Berdasarkan Gambar 4.10 dan Gambar 4.11 aplikasi monitoring menampilkan hasil pembacaan sensor dan kondisi sistem secara *real-time* selama alat dijalankan. Informasi yang ditampilkan meliputi nilai kelembapan lingkungan, status kondisi lingkungan, mode operasi sistem, kondisi pompa air, serta status lampu pemanas. Nilai kelembapan yang ditampilkan pada aplikasi sebesar 94% menunjukkan bahwa kondisi lingkungan berada pada kategori sangat lembap. Pada kondisi tersebut pompa berada pada status *off*, sedangkan lampu pemanas berada pada kondisi *on*. Respons tersebut menunjukkan bahwa sistem telah menjalankan logika pengendalian sesuai program yang dirancang, yaitu pompa tidak aktif ketika kelembapan tinggi dan sistem memberikan respons terhadap kondisi lingkungan melalui aktuator yang tersedia.

Selain menampilkan data kondisi lingkungan, aplikasi juga menyediakan pilihan mode otomatis dan manual. Mode otomatis memungkinkan sistem bekerja berdasarkan hasil pembacaan sensor, sedangkan mode manual memungkinkan pengguna mengendalikan perangkat secara langsung melalui aplikasi. Selain ditampilkan pada aplikasi, data hasil pembacaan sensor juga tersimpan dalam database. Setiap kali sensor membaca data baru, sistem akan menyimpannya sebagai baris baru pada tabel, sehingga histori kondisi kumbung selama alat beroperasi dapat dilihat dan ditelusuri kembali. Pada tabel tersebut juga terlihat adanya data dengan status "Manual" maupun "Otomatis" yang menunjukkan bahwa kedua mode operasi sistem tercatat dan tersimpan dengan baik ke dalam database. Berdasarkan jumlah data yang berhasil terekam, dapat disimpulkan bahwa proses pengiriman data dari sensor ke ESP32 kemudian diteruskan ke database Supabase berjalan secara stabil dan tidak terputus selama alat dioperasikan.

Hasil implementasi sistem monitoring menunjukkan bahwa proses pengiriman data dari sensor menuju ESP32, kemudian diteruskan ke *Supabase* dan ditampilkan pada aplikasi monitoring dapat berjalan dengan baik. Integrasi sistem monitoring berbasis IoT mampu mendukung proses pengawasan kondisi lingkungan kumbung secara *real-time* sehingga memudahkan pengguna dalam memantau kondisi pertumbuhan jamur tiram (Palanda, 2025).

4.4 Evaluasi dan Pengujian Sistem dan Alat

Implementasi pengendalian penyiraman otomatis dilakukan untuk mengetahui kemampuan sistem dalam merespons perubahan kondisi lingkungan kumbung jamur tiram berdasarkan parameter kelembapan yang telah ditentukan. Sistem dirancang agar mampu melakukan pengambilan keputusan secara otomatis melalui pembacaan sensor DHT22, kemudian memproses data menggunakan ESP32 dan mengendalikan aktuator berupa pompa air melalui modul *relay*. Pengendalian otomatis ini diterapkan untuk menjaga kondisi lingkungan kumbung tetap berada pada rentang optimal pertumbuhan jamur tiram sehingga proses budidaya dapat berjalan lebih efektif.

Pengujian dilakukan terhadap beberapa komponen dan sistem secara keseluruhan untuk mengetahui kinerja sistem dalam merespons perubahan kondisi lingkungan. Pengujian meliputi kalibrasi kelembapan, pengujian sensor DHT22, pengujian modul *relay* dan pompa air, serta pengujian sistem penyiraman otomatis secara menyeluruh. Hasil pengujian digunakan untuk mengetahui apakah sistem bekerja sesuai logika dan tujuan perancangan yang telah ditentukan sebelumnya.

4.4.1 Pengujian Kalibrasi Kelembapan

Pengujian kalibrasi dilakukan untuk menentukan batas kondisi lingkungan yang digunakan sebagai acuan dalam mengambil keputusan penyiraman otomatis (Rosi dan Amsori, 2025). Penentuan parameter dilakukan berdasarkan kebutuhan pertumbuhan jamur tiram khususnya pada kelembapan kumbung.

Tabel 4.1 Parameter kelembapan sistem

Parameter	Kondisi	Respons Sistem
<70%	Kelembapan rendah	Pompa aktif
70 – 90%	Kondisi Optimal	Pompa nonaktif
>90%	Kelembapan tinggi	Pompa nonaktif

Berdasarkan Tabel 4.1 sistem menggunakan kelembapan sebagai parameter utama dalam pengambilan keputusan penyiraman otomatis. Ketika nilai kelembapan berada di bawah kondisi optimal, ESP32 akan mengaktifkan *relay* sehingga pompa menyala dan melakukan penyemprotan kabut air. Sebaliknya, ketika kondisi kelembapan mencapai rentang ideal pertumbuhan jamur tiram, sistem menghentikan proses penyiraman secara otomatis. Logika keputusan sistem

diterapkan dalam kondisi *if-else* pada program ESP32 sehingga pengendalian berlangsung secara otomatis sesuai pembacaan sensor (Saputra, 2025).

4.4.2 Pengujian Sistem Penyiraman Otomatis

Pengujian sistem penyiraman otomatis dilakukan untuk mengetahui kemampuan sistem dalam menjalankan proses pengendalian berdasarkan hasil pembacaan sensor DHT22. Pengujian dilakukan dengan mengamati respons aktuator berupa relay, pompa air, dan lampu terhadap perubahan nilai kelembapan lingkungan yang terbaca oleh sistem. Pengujian ini juga digunakan untuk memastikan bahwa logika pengendalian yang diterapkan pada sistem dapat berjalan sesuai dengan parameter yang telah ditetapkan.

Tabel 4.2 Hasil respons sistem penyiraman otomatis

Kelembapan	Status Sistem	Pompa	Lampu
<70%	Kering	<i>on</i>	<i>off</i>
70 – 90%	Normal	<i>off</i>	<i>off</i>
>90%	Terlalu lembap	<i>off</i>	<i>on</i>

Berdasarkan Tabel 4.2 sistem menerapkan logika pengendalian berdasarkan rentang kelembapan yang telah ditentukan pada program. Ketika nilai kelembapan berada di bawah 70%, sistem mengategorikan kondisi lingkungan dalam keadaan kering sehingga *relay* pompa diaktifkan untuk menjalankan proses penyiraman otomatis (Subekti *et al.*, 2024). Sebaliknya apabila nilai kelembapan melebihi 90%, sistem mengaktifkan *relay* lampu sebagai indikator bahwa kondisi lingkungan berada pada tingkat kelembapan yang terlalu tinggi. Pada rentang kelembapan 70–90%, sistem mengategorikan kondisi lingkungan sebagai kondisi normal sehingga pompa dan lampu berada pada kondisi *off*. Logika pengendalian tersebut digunakan sebagai dasar dalam proses pengambilan keputusan untuk menjaga kondisi lingkungan kumbung secara otomatis. Pengujian yang dilakukan secara berulang pada setiap rentang kelembapan menunjukkan bahwa respons aktuator berjalan konsisten sesuai dengan logika program, tanpa ditemukan kondisi tumpang tindih antara status pompa dan lampu yang dapat menimbulkan kesalahan pengendalian. Dengan demikian dapat disimpulkan bahwa sistem mampu melakukan pengambilan keputusan secara otomatis dan akurat berdasarkan data kelembapan yang terbaca oleh sensor DHT22.



Gambar 4.12 Kondisi lampu menyala



Gambar 4.13 Kondisi nozzle menyala

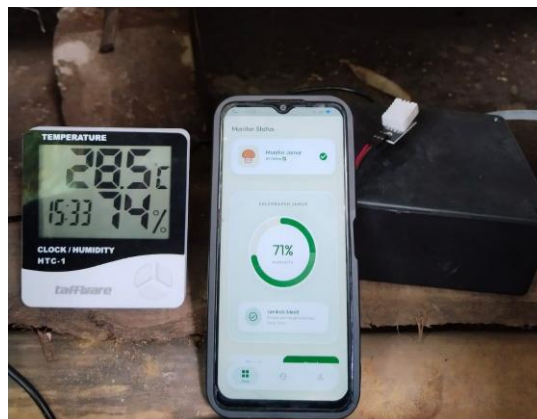
4.4.3 Validasi Sensor DHT22 dan *Hygrometer*

Tabel 4.3 Validasi data kelembapan

No.	DHT22 (%)	Hygrometer (%)	Keterangan
1.	71	74	Sesuai
2.	71	75	Sesuai
3.	72	74	Sesuai
4.	73	74	Sesuai
5.	75	76	Sesuai
6.	76	75	Sesuai
7.	76	79	Sesuai
8.	77	78	Sesuai
9.	78	80	Sesuai
10.	84	84	Sesuai
11.	85	85	Sesuai
12.	88	87	Sesuai
13.	91	90	Sesuai
14.	92	91	Sesuai
15.	94	94	Sesuai

Berdasarkan Tabel 4.3 hasil validasi sensor DHT22 terhadap alat ukur pembanding berupa *hygrometer* menunjukkan tingkat akurasi yang baik, dimana dari 15 kali pengujian seluruh data memiliki selisih kelembapan di bawah 5%,

sehingga dapat dikategorikan valid sesuai dengan ketentuan toleransi yang ditetapkan. Nilai kelembapan yang terbaca oleh sensor DHT22 berkisar antara 71% hingga 94%, sementara nilai yang terbaca oleh hygrometer berkisar antara 74% hingga 94% dengan selisih terbesar terjadi pada data ke-7 yaitu sebesar 3%. Pada beberapa pengujian, yaitu data ke-10, ke-11, dan ke-15, nilai yang terbaca oleh kedua alat menunjukkan hasil yang identik atau sesuai yang mengindikasikan tingkat presisi sensor DHT22 yang sangat mendekati alat ukur pembeding. Dengan demikian sensor DHT22 yang digunakan dalam penelitian ini terbukti memiliki tingkat akurasi yang dapat diandalkan untuk mengukur kelembapan udara, karena secara konsisten menghasilkan selisih pembacaan yang tidak melebihi batas toleransi 5% dibandingkan dengan *hygrometer* yang beredar di pasaran.



Gambar 4.14 Hasil pengukuran kelembapan rendah



Gambar 4.15 Hasil pengukuran kelembapan tinggi

4.4.4 Kinerja Sistem dalam Menjaga Kondisi Optimal Pertumbuhan Jamur Tiram

Analisis kinerja sistem dilakukan berdasarkan hasil implementasi dan pengujian sistem penyiraman otomatis berbasis *Internet of Things* yang telah dilakukan. Analisis difokuskan pada kemampuan sistem dalam menjaga kondisi lingkungan kumbung agar tetap berada pada rentang optimal pertumbuhan jamur tiram. Parameter yang dianalisis meliputi kestabilan kelembapan lingkungan, efisiensi proses penyiraman otomatis, serta kemampuan sistem monitoring dalam mengirimkan dan menampilkan data secara *real-time*. Hasil pengujian menunjukkan bahwa sistem mampu membaca kondisi kelembapan menggunakan sensor DHT22, memproses data melalui ESP32, serta menjalankan logika pengendalian sesuai parameter yang telah ditentukan. Selama proses pengujian, nilai kelembapan yang terbaca berada pada rentang 73–91%, variasi nilai tersebut menunjukkan bahwa sistem dapat mendeteksi perubahan kondisi lingkungan yang terjadi pada waktu pengamatan yang berbeda.

a) Stabilitas Kelembapan Kumbung Jamur

Kondisi kelembapan lingkungan menjadi salah satu faktor penting dalam pertumbuhan jamur tiram karena berpengaruh terhadap proses pembentukan tubuh buah dan perkembangan miselium. Sistem dirancang untuk mempertahankan kondisi kelembapan pada rentang 70–90% agar lingkungan kumbung tetap sesuai dengan kebutuhan pertumbuhan jamur (Rahman dan Muskhir, 2021). Kestabilan kondisi lingkungan tersebut menjadi indikator penting dalam menilai keberhasilan sistem pengendalian yang diterapkan pada kumbung jamur tiram.

Tabel 4.4 Hasil pengamatan kelembapan

Tanggal	Waktu Pengambilan	Kelembapan (%)	Status Pompa	Status Lampu
21 Mei 2026	14:42:13	81.20	Mati	Menyala (Manual)
	14:42:16	81.20	Mati	Mati
	14:49:22	81.60	Mati	Menyala (Manual)
	14:49:36	81.90	Mati	Menyala (Manual)
	14:50:06	81.40	Mati	Mati
	19:10:27	88.20	Mati	Menyala (Manual)
	19:10:40	88.10	Mati	Mati
	19:10:42	88.20	Mati	Menyala (Manual)
	19:10:44	88.20	Mati	Mati

	19:10:46	88.20	Mati	Menyala (Manual)
	19:10:48	88.10	Mati	Mati
	19:11:04	88.10	Mati	Menyala (Manual)
	19:11:12	88.10	Mati	Mati
	19:15:04	95.50	Mati	Menyala (Manual)
	19:15:29	89.70	Mati	Mati
	21:25:16	90.70	Mati	Menyala (Manual)
22 Mei 2026	17:05:58	81.10	Mati	Menyala (Manual)
	17:06:08	81.40	Mati	Mati
	17:06:14	81.50	Mati	Menyala (Manual)
	17:06:23	82.00	Mati	Mati
	17:06:27	81.20	Mati	Menyala (Manual)
	17:06:30	81.20	Mati	Mati
	17:07:54	80.80	Mati	Menyala (Manual)
	17:11:59	80.80	Mati	Menyala (Manual)
	17:12:02	80.70	Mati	Mati
	17:12:04	80.50	Mati	Menyala (Manual)
	17:12:06	80.70	Mati	Mati
	17:33:31	82.30	Mati	Menyala (Manual)
	17:35:53	82.40	Mati	Menyala (Manual)
	17:36:00	82.80	Mati	Mati
	17:37:17	83.60	Mati	Menyala (Manual)
	17:40:37	83.80	Mati	Menyala (Manual)
	17:40:48	84.10	Mati	Mati
	17:57:46	85.40	Mati	Mati
	18:03:37	83.60	Mati	Menyala (Manual)
	18:04:27	84.30	Mati	Mati
23 Mei 2026	15:37:25	0.00	Menyala (Otomatis)	Mati
	15:46:51	87.60	Mati	Menyala (Manual)
	15:46:55	87.60	Mati	Mati
	15:46:58	87.50	Mati	Menyala (Manual)
	15:47:24	87.10	Mati	Mati
	15:47:29	87.20	Mati	Menyala (Manual)
	15:47:50	87.00	Mati	Mati
24 Mei 2026	07:35:50	81.40	Mati	Menyala (Manual)
	07:35:52	81.50	Mati	Mati
	07:35:56	81.10	Menyala (Manual)	Mati
	07:35:58	81.10	Mati	Mati
	07:36:01	80.80	Menyala (Manual)	Mati
	07:36:03	80.90	Mati	Mati
	07:36:05	80.80	Menyala (Manual)	Mati
	07:36:10	79.90	Mati	Mati

07:37:58	81.30	Menyala (Manual)	Mati
07:38:03	80.90	Mati	Mati
07:38:07	80.80	Mati	Menyala (Manual)
07:38:49	79.70	Mati	Mati
07:51:21	86.30	Mati	Mati
07:55:34	88.10	Mati	Mati
07:59:18	89.40	Mati	Mati
08:02:11	90.40	Mati	Menyala (Otomatis)
08:02:20	90.40	Mati	Menyala (Otomatis)

Berdasarkan Tabel 4.4 hasil pembacaan sensor menunjukkan bahwa nilai kelembapan selama pengujian berada pada rentang 79,70–95,50%. Nilai kelembapan terendah tercatat pada tanggal 24 Mei 2026 pagi hari, sedangkan nilai tertinggi terjadi pada tanggal 21 Mei 2026 malam hari. Perbedaan ini menunjukkan bahwa kelembapan di dalam kumbung berubah-ubah tergantung waktu pengamatan. Meskipun begitu sebagian besar nilai kelembapan masih berada pada rentang 70–90% yang sesuai untuk pertumbuhan jamur tiram. Selama pengujian status pompa sebagian besar tercatat mati karena kelembapan sudah berada dalam kondisi ideal, sehingga penyiraman tambahan tidak diperlukan (Fissabillillah *et al.*, 2025). Pada tanggal 23 Mei 2026 pukul 15.37 tercatat nilai kelembapan 0,00% dengan status pompa menyala otomatis, namun hal ini bukan disebabkan oleh kondisi kumbung yang benar-benar kering, melainkan karena alat baru saja diaktifkan sehingga sensor belum sempat membaca data kelembapan secara stabil. Selain itu pompa juga diaktifkan secara manual pada tanggal 24 Mei 2026 pagi hari untuk menguji fungsi kontrol manual pada sistem. Pada tanggal 24 Mei 2026 pukul 08.02 nilai kelembapan mencapai 90,40% dan sistem langsung menyalakan lampu secara otomatis sesuai logika program yang telah dibuat. Dari data ini dapat dilihat bahwa sistem IoT yang dirancang sudah mampu memantau dan mengontrol kondisi kumbung dengan cukup baik, sehingga kelembapan dapat dijaga tetap stabil untuk mendukung pertumbuhan jamur tiram (Saputra, 2025).

id	kelembapan	status	tanggal_upload	status_tanpa	status_pompa
96	81.20	MANUAL	2026-05-21 14:42:13	MENYALA - MANUAL	MATI
97	81.20	MANUAL	2026-05-21 14:42:16	MATI	MATI
98	81.60	MANUAL	2026-05-21 14:49:22	MENYALA - MANUAL	MATI
99	81.90	Lembab (Ideal)	2026-05-21 14:49:36	MENYALA - MANUAL	MATI
100	81.40	MANUAL	2026-05-21 14:50:06	MATI	MATI
101	88.20	MANUAL	2026-05-21 19:10:27	MENYALA - MANUAL	MATI
102	88.10	MANUAL	2026-05-21 19:10:40	MATI	MATI
103	88.20	MANUAL	2026-05-21 19:10:42	MENYALA - MANUAL	MATI
104	88.20	MANUAL	2026-05-21 19:10:44	MATI	MATI
105	88.20	MANUAL	2026-05-21 19:10:46	MENYALA - MANUAL	MATI
106	88.10	MANUAL	2026-05-21 19:10:48	MATI	MATI
107	88.10	MANUAL	2026-05-21 19:11:04	MENYALA - MANUAL	MATI
108	88.10	MANUAL	2026-05-21 19:11:12	MATI	MATI
109	95.50	TERLALU LEMBAB	2026-05-21 19:15:04	MENYALA - OTOMATIS	MATI
110	89.70	NORMAL	2026-05-21 19:15:29	MATI	MATI
111	90.70	TERLALU LEMBAB	2026-05-21 21:25:16	MENYALA - OTOMATIS	MATI

Gambar 4.16 Screenshot database 21 Mei 2026

id	kelembapan	status	tanggal_upload	status_tanpa	status_pompa
119	81.10	MANUAL	2026-05-22 17:05:58	MENYALA - MANUAL	MATI
120	81.40	MANUAL	2026-05-22 17:06:08	MATI	MATI
121	81.50	MANUAL	2026-05-22 17:06:14	MENYALA - MANUAL	MATI
122	82.00	MANUAL	2026-05-22 17:06:23	MATI	MATI
123	81.20	MANUAL	2026-05-22 17:06:27	MENYALA - MANUAL	MATI
124	81.20	MANUAL	2026-05-22 17:06:30	MATI	MATI
125	80.80	MANUAL	2026-05-22 17:07:54	MENYALA - MANUAL	MATI
126	80.80	MANUAL	2026-05-22 17:11:59	MENYALA - MANUAL	MATI
127	80.70	MANUAL	2026-05-22 17:12:02	MATI	MATI
128	80.50	MANUAL	2026-05-22 17:12:04	MENYALA - MANUAL	MATI
129	80.70	MANUAL	2026-05-22 17:12:06	MATI	MATI
130	82.30	MANUAL	2026-05-22 17:33:31	MENYALA - MANUAL	MATI
131	82.40	MANUAL	2026-05-22 17:35:53	MENYALA - MANUAL	MATI
132	82.80	MANUAL	2026-05-22 17:36:00	MATI	MATI
133	83.60	MANUAL	2026-05-22 17:37:17	MENYALA - MANUAL	MATI
134	83.80	MANUAL	2026-05-22 17:40:37	MENYALA - MANUAL	MATI
135	84.10	MANUAL	2026-05-22 17:40:48	MATI	MATI

Gambar 4.17 Screenshot database 22 Mei 2026

id	kelembapan	status	tanggal_upload	status_tanpa	status_pompa
139	85.40	MANUAL	2026-05-22 17:57:46	MATI	MATI
140	83.60	MANUAL	2026-05-22 18:03:37	MENYALA - MANUAL	MATI
141	84.30	MANUAL	2026-05-22 18:04:27	MATI	MATI
142	0.00	KERING / PENYIRAMAN	2026-05-23 15:37:25	MATI	MENYALA - OTOMATIS
143	87.60	MANUAL	2026-05-23 15:46:51	MENYALA - MANUAL	MATI
144	87.60	MANUAL	2026-05-23 15:46:55	MATI	MATI
145	87.50	MANUAL	2026-05-23 15:46:58	MENYALA - MANUAL	MATI
146	87.10	MANUAL	2026-05-23 15:47:24	MATI	MATI
147	87.20	MANUAL	2026-05-23 15:47:29	MENYALA - MANUAL	MATI
148	87.00	MANUAL	2026-05-23 15:47:50	MATI	MATI

Gambar 4.18 Screenshot database 23 Mei 2026

id	kelembapan	status	waktu	pompa	pompa
150	81.50	MANUAL	2026-05-24 07:35:52	MATI	MATI
151	81.10	MANUAL	2026-05-24 07:35:56	MATI	MENYALA - MANUAL
152	81.10	MANUAL	2026-05-24 07:35:58	MATI	MATI
153	80.80	MANUAL	2026-05-24 07:36:01	MATI	MENYALA - MANUAL
154	80.90	MANUAL	2026-05-24 07:36:03	MATI	MATI
155	80.80	MANUAL	2026-05-24 07:36:05	MATI	MENYALA - MANUAL
156	79.90	MANUAL	2026-05-24 07:36:10	MATI	MATI
157	81.30	MANUAL	2026-05-24 07:37:58	MATI	MENYALA - MANUAL
158	80.90	MANUAL	2026-05-24 07:38:03	MATI	MATI
159	80.80	MANUAL	2026-05-24 07:38:07	MENYALA - MANUAL	MATI
160	79.70	MANUAL	2026-05-24 07:38:49	MATI	MATI
161	86.30	Lembab (ideal)	2026-05-24 07:51:21	MATI	MATI
162	88.10	Lembab (ideal)	2026-05-24 07:55:34	MATI	MATI
163	89.40	Lembab (ideal)	2026-05-24 07:59:18	MATI	MATI
164	90.40	TERLALU LEMBAB	2026-05-24 08:02:11	MENYALA - OTOMATIS	MATI
165	90.40	Sangat Lembab	2026-05-24 08:02:20	MENYALA - OTOMATIS	MATI

Gambar 4.19 Screenshot database 24 Mei 2026

b) Efisiensi Penyiraman Otomatis

Penerapan sistem penyiraman otomatis memungkinkan proses penyiraman dilakukan berdasarkan kondisi lingkungan aktual yang terbaca oleh sensor. Berbeda dengan metode manual yang umumnya dilakukan secara berkala tanpa mempertimbangkan kondisi lingkungan secara langsung, sistem otomatis hanya akan melakukan penyiraman ketika kelembapan berada di bawah batas yang telah ditentukan (Daniswara dan Harja, 2025).

Tabel 4.5 Perbandingan penyiraman manual dan otomatis

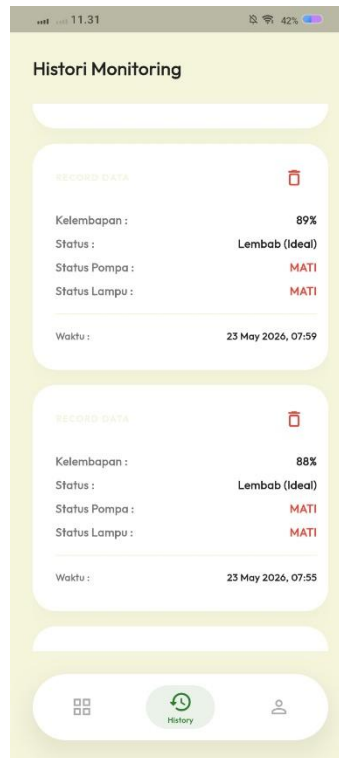
Parameter	Manual	Otomatis
Dasar penyiraman	Jadwal Pengguna	Pembacaan sensor
Frekuensi penyiraman	Terjawab	Sesuai kondisi
Potensi penggunaan air	Lebih tinggi	Lebih efisien

Berdasarkan Tabel 4.5 sistem penyiraman otomatis memiliki potensi penggunaan air yang lebih efisien karena proses penyiraman tidak dilakukan secara terus-menerus. Kondisi lingkungan kumbungjamur telah berada pada rentang kelembapan optimal sehingga sistem tidak melakukan penyiraman tambahan. Kondisi tersebut menunjukkan bahwa proses penyiraman berjalan sesuai kebutuhan lingkungan dan berpotensi mengurangi frekuensi penyiraman yang tidak diperlukan serta mendukung penggunaan air yang lebih efisien.

c) Kinerja Monitoring *Real-Time*

Monitoring *real-time* dilakukan untuk mengetahui kemampuan sistem dalam mengirimkan data kondisi lingkungan menuju platform monitoring berbasis *Internet of Things*. Sistem monitoring memanfaatkan koneksi *Wi-Fi* pada ESP32

yang terhubung dengan *Supabase* sehingga data hasil pembacaan sensor dapat dikirim dan ditampilkan secara otomatis pada aplikasi monitoring (Azis dan Sujono, 2026).



Gambar 4.20 Hasil monitoring data real-time

Berdasarkan Gambar 4.20 sistem monitoring berhasil menampilkan data hasil pembacaan sensor berupa nilai kelembapan, status kondisi lingkungan, status pompa, status lampu, serta waktu pengambilan data. Tampilan histori monitoring menunjukkan bahwa data hasil pembacaan tersimpan secara berurutan sehingga pengguna dapat melakukan pemantauan kondisi lingkungan kumbung secara berkala. Hasil monitoring menunjukkan nilai kelembapan sebesar 89% dan 88% dengan status kondisi "Lembab (Ideal)". Nilai tersebut menunjukkan bahwa kondisi lingkungan kumbung masih berada pada rentang kelembapan yang sesuai untuk pertumbuhan jamur tiram. Pada kondisi tersebut status pompa dan lampu berada pada keadaan mati (*OFF*) karena sistem mengategorikan kondisi lingkungan berada pada kondisi normal sehingga tidak diperlukan proses penyiraman tambahan maupun perlakuan lainnya.

Informasi waktu yang ditampilkan pada histori monitoring menunjukkan bahwa sistem berhasil melakukan pencatatan dan pengiriman data secara berkala. Hal tersebut menunjukkan bahwa proses komunikasi data dari sensor DHT22 menuju ESP32, kemudian diteruskan ke *Supabase* dan ditampilkan pada aplikasi monitoring dapat berjalan dengan baik. Hasil analisis menunjukkan bahwa sistem monitoring berbasis *Internet of Things* mampu menjalankan proses pengiriman dan penyajian data secara *real-time*. Integrasi ESP32, jaringan *Wi-Fi*, dan *Supabase* mampu mendukung proses pemantauan kondisi lingkungan kumbung secara efisien sehingga pengguna dapat memperoleh informasi kondisi lingkungan secara langsung tanpa harus berada di lokasi budidaya.

Selain itu, tampilan histori monitoring memudahkan pengguna dalam melakukan penelusuran terhadap perubahan kondisi kelembapan yang terjadi dari waktu ke waktu. Data yang tersimpan secara berkelanjutan juga dapat dimanfaatkan sebagai bahan evaluasi untuk mengetahui pola perubahan kelembapan di dalam kumbung. Dengan adanya riwayat data tersebut, pengguna dapat menentukan langkah yang tepat apabila terjadi perubahan kondisi lingkungan yang tidak sesuai dengan kebutuhan budidaya. Hasil pengujian ini menunjukkan bahwa sistem tidak hanya berfungsi sebagai alat monitoring, tetapi juga sebagai media pencatatan data yang mendukung proses pengambilan keputusan. Oleh karena itu, fitur histori monitoring yang terintegrasi dengan *Supabase* mampu meningkatkan efektivitas pemantauan kondisi lingkungan kumbung jamur tiram secara berkelanjutan.

Selain itu, sistem monitoring juga mampu menyajikan informasi dengan antarmuka yang mudah dipahami oleh pengguna. Seluruh data yang ditampilkan diperbarui secara otomatis sesuai dengan hasil pembacaan sensor terbaru. Kondisi tersebut menunjukkan bahwa proses sinkronisasi data antara perangkat keras dan aplikasi monitoring telah berjalan dengan baik selama pengujian. Dengan demikian, sistem yang dikembangkan mampu memberikan kemudahan dalam memantau kondisi lingkungan kumbung secara efektif, akurat, dan berkesinambungan.

BAB 5. PENUTUP

5.1 Kesimpulan

Berdasarkan hasil perancangan dan pengujian sistem penyiraman jamur tiram berbasis IoT yang telah dilakukan, diperoleh kesimpulan sebagai berikut.

1. Sistem berhasil dirancang dan dibangun menggunakan ESP32 sebagai mikrokontroler utama yang terintegrasi dengan sensor DHT22 untuk membaca kelembapan udara di dalam kumbung. Hasil perakitan menunjukkan keempat komponen utama yaitu ESP32, sensor DHT22, modul relay, dan *step-down* LM2596 dapat bekerja secara terintegrasi dalam satu box control. Pemantauan *real-time* terwujud melalui koneksi Wi-Fi pada ESP32 yang mengirim data ke platform Supabase, kemudian ditampilkan pada aplikasi mobile *Mushroom Monitor*. Hasil validasi sensor DHT22 terhadap *hygrometer* pembanding menunjukkan selisih pembacaan konsisten di bawah 5%, sehingga sensor terbukti akurat dan andal untuk keperluan pemantauan kelembapan.
2. Sistem pengendalian yang dikembangkan mampu mengendalikan proses penyiraman secara otomatis berdasarkan hasil pembacaan sensor DHT22 yaitu mengaktifkan pompa saat kelembapan di bawah 70% (kering), mematikan pompa saat kelembapan berada pada rentang 70–90% (kondisi ideal), dan mengaktifkan lampu saat kelembapan di atas 90% (terlalu lembap) sebagai respons pencegahan kontaminasi. Sistem juga menyediakan mode manual agar pengguna dapat mengendalikan perangkat sesuai kebutuhan.
3. Kinerja sistem terbukti mampu menjaga kelembapan kumbung tetap berada pada rentang optimal, yaitu 79,70–95,50% selama pengujian, dengan pompa hanya aktif saat benar-benar diperlukan. Kinerja sistem juga menunjukkan efisiensi penyiraman yang lebih baik dibandingkan metode manual, karena proses penyiraman dilakukan sesuai pembacaan sensor secara aktual sehingga mengurangi frekuensi penyiraman yang tidak diperlukan. Selain itu, kinerja sistem dalam monitoring *real-time* berjalan stabil tanpa terputus, sehingga mendukung pengawasan kondisi lingkungan kumbung secara berkelanjutan dan lebih konsisten dibandingkan pengamatan manual oleh pekerja.

5.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan, berikut merupakan beberapa saran untuk pengembangan penelitian selanjutnya.

1. Pengembangan sistem dapat dilakukan dengan menggunakan sensor yang memiliki tingkat akurasi lebih tinggi serta melakukan validasi hasil pembacaan menggunakan alat pembanding seperti *thermohygrometer*, sehingga data kelembapan yang diperoleh dapat dianalisis dengan tingkat ketelitian yang lebih baik.
2. Pengembangan pada sisi perangkat keras dapat dilakukan melalui penggunaan catu daya adaptor dengan kapasitas arus yang lebih besar dan stabil agar seluruh komponen sistem seperti ESP32, *relay*, dan pompa air, dapat bekerja secara optimal selama proses pengoperasian berlangsung.
3. Sistem monitoring berbasis *Internet of Things* dapat dikembangkan dengan penambahan fitur notifikasi otomatis, penyimpanan data (*data logging*), serta visualisasi grafik kondisi lingkungan sehingga proses pemantauan dapat dilakukan secara lebih efektif.
4. Pengujian sistem perlu dilakukan dalam jangka waktu yang lebih panjang serta pada kondisi lingkungan yang lebih beragam untuk mengetahui kestabilan sistem, ketahanan alat, dan konsistensi kinerja dalam mendukung budidaya jamur tiram secara berkelanjutan.

DAFTAR PUSTAKA

- Arsella, S., Fadhli, M., & Lindawati, L. (2023). Optimasi pertumbuhan jamur tiram melalui monitoring suhu dan kelembaban menggunakan teknologi IoT. *Jurnal RESISTOR (Rekayasa Sistem Komputer)*, 6(1), 34-42.
- Azis, A., & Sujono, S. (2026). Implementasi Mikrokontroler ESP32 Dan Sensor Kualitas Air Dalam Sistem Monitoring Lingkungan Berbasis Internet of Things. *Jurnal Manajemen Sistem Informasi*, 5(1), 1-13.
- Daniswara, I. Z., & Harja, A. Z. (2025). *Sistem Penyiraman Otomatis di Botanical SmartPark SMA UII* (Doctoral dissertation, Universitas Islam Indonesia).
- Eteruddin, H., Dini, I. R., & Huda, F. (2024). Pengaruh Suhu dan Kelembaban terhadap Produktivitas Jamur Tiram. *Jurnal Teknik*, 18(2), 1-5
- Fissabilillah, D. A., Harianto, H., Susanto, P., & Kusumawati, W. I. (2025). Perancangan dan implementasi sistem pemantauan serta pengendalian kelembapan pada alat pengering produk pangan berbasis IoT. *INFOTECH: Jurnal Informatika & Teknologi*, 6(2), 179-188.
- Hidayah, R. R., Nurcahyo, S., & Dewatama, D. (2024). Implementasi pengaturan suhu menggunakan mikrokontroler ESP32. *Metrotech (Journal of Mechanical and Electrical Technology)*, 3(3), 106-115.
- Jannah, S. W., Fathira, D. S. U., & Astutik, S. N. (2026). Pelatihan Alat Monitoring dan Kontrol Kelembapan Kumbung Jamur Berbasis IOT di Desa Binor. *Babakti: Journal of Community Engagement*, 3(1), 12-19.
- Jumadil, D. M. (2025). *Perancangan Sistem Irigasi Cerdas pada Tanaman Hias Berbasis Internet Of Things* (Doctoral dissertation, Universitas Islam Indonesia).
- Nurhuda, M., & Khosyi'in, M. (2024). Perancangan Alat Uji Relay Thermal Trafo Tenaga Berbasis Arduino Mega 2560 Dan Sensor Suhu Thermocouple Type K. *Syntax Idea*, 6(10), 6485-6499.
- Palanda, M. (2025). *Desain Sistem Monitoring Kumbung Jamur Tiram dengan Integrasi Sensor Suhu dan Kelembaban Udara Berbasis IoT* (Doctoral dissertation, Universitas Kristen Indonesia Toraja).
- Pradito, R., Nugraha, S., & Suhendra, T. (2022). Perancangan Autogate Menggunakan Mikrokontroler ESP32 Nodemcu. *Student Online Journal (SOJ) UMRAH-Teknik*, 3(1), 28-30.

- Pramana, R., Pinandito, A. M., Septiana, T., & Syamsudin, M. S. (2025). Analisis Dampak dan Tantangan Pemanfaatan Sensor Kelembaban Tanah dalam sistem Irigasi Otomatis Berbasis IoT di Sektor Pertanian Indonesia. *Qomaruna: Journal of Multidisciplinary Studies*, 3(1), 24-31.
- Putra, D. R., Azhari, S. K., & Fiki, A. (2024). rancang bangun bel cerdas cermat berbasis arduino AtMEGA 328. *Jurnal Teknologi Rekayasa Instalasi Listrik (JTRIL)*, 1(1), 21-30.
- Rahman, R. A., & Muskhir, M. (2021). Monitoring pengontrolan suhu dan kelembaban kumbung jamur tiram. *JTEIN: Jurnal Teknik Elektro Indonesia*, 2(2), 266-272.
- Rosi, F. R. F., & Amsori, A. (2025). Sistem Penyiraman Otomatis Pada Tanaman Bawang Merah Dengan Sensor Kelembaban Tanah Berbasis Arduino Uno. *Karapan Network Journal: Journal Computer Technology and Mobile Ad Hoc Network*, 1(01).
- Saputra, A. D. (2025). *Implementasi Sistem Otomatis Penyiraman Jamur Tiram Berbasis Mikrokontroler ESP32 dan Database Firebase* (Doctoral dissertation, Universitas Islam Indonesia).
- Subekti, A. B., Romdoni, A. M., & Sidiq, J. M. (2024, July). Rancang Bangun Sistem Monitoring Suhu Dan Kelembapan Dengan Kontrol Pompa Air Berbasis Internet Of Things Pada Kebun Budidaya Jamur. In *Prosiding Seminar Nasional Teknologi Informasi dan Bisnis* (pp. 173-176).
- Suherman, S., Hermansyah, H., & Syahpita, J. (2025). Sistem Alarm Deteksi Gerak Berbasis IoT Menggunakan Sensor PIR dan ESP32 dengan Notifikasi Telegram Real-Time. *Jurnal Komputer Teknologi Informasi Sistem Informasi (JUKTISI)*, 4(2), 1469-1476.
- Yonatan, Y. K., Mukti, A. T., & Firmansyah, R. N. (2025). Analisa Penerapan Sensor Suhu Menggunakan Arduino Uno di dalam Sistem Pertanian GreenHouse: Sensor DHT22. *Journal of Engineering Environmental Energy and Science*, 4(1), 39-46.

LAMPIRAN

Lampiran 1. Kode Program

```
1) #include <WiFi.h>
2) #include <WiFiClientSecure.h>
3) #include <HTTPClient.h>
4) #include <PubSubClient.h>
5) #include <DHT.h>
6) #include <time.h>
7) // ===== SUPABASE =====
8) const char* supabase_url =
    "https://hpmicdhjyboyeofphgae.supabase.co";
9) const char* supabase_key =
    "sb_publishable_OB8e9y1000z3Y5xQ828YvA_T3jD16zT";
10)
11) // ===== WIFI =====
12) const char* ssid = "Ozie 1";
13) const char* password = "followkami";
14)
15) // ===== TIME (NTP) =====
16) const char* ntpServer = "pool.ntp.org";
17) const long  gmtOffset_sec = 7 * 3600; // WIB (GMT+7)
18) const int   daylightOffset_sec = 0;
19)
20) // ===== MQTT =====
21) const char* mqtt_server = "broker.hivemq.com";
22)
23) WiFiClient espClient;
24) PubSubClient client(espClient);
25)
26) // ===== DHT =====
27) #define DHTPIN 4
28) #define DHTTYPE DHT22
29) DHT dht(DHTPIN, DHTTYPE);
30)
31) // ===== RELAY =====
32) #define RELAY1 27
33) #define RELAY2 26
34)
35) // Relay aktif LOW
36) #define RELAY_ON  LOW
37) #define RELAY_OFF HIGH
38)
39) // ===== THRESHOLD =====
40) float batasPanas = 70.0;
41) float batasLembab = 90.0;
42)
43) // ===== GLOBAL STATE =====
```

```

44) unsigned long lastMsg = 0;
45) String statusRelay = "INIT";
46) bool isAutoMode = true;
47) bool manualPump = false;
48) bool manualLight = false;
49)
50) // State Tracking for Upload
51) bool prevPump = false;
52) bool prevLight = false;
53) unsigned long lastPeriodicUpload = 0;
54)
55) // ===== SUPABASE UPLOAD =====
56) void uploadToSupabase(float hum, String status, String lamp,
    String pump) {
57) if (WiFi.status() == WL_CONNECTED) {
58) // Ambil waktu saat ini
59) struct tm timeinfo;
60) String timeStr = "";
61) if (getLocalTime(&timeinfo)) {
62) char timeStringBuff[50];
63) strftime(timeStringBuff, sizeof(timeStringBuff), "%H:%M:%S",
    &timeinfo);
64) timeStr = String(timeStringBuff);
65) }
66)
67) WiFiClientSecure clientSecure;
68) clientSecure.setInsecure();
69) HTTPClient http;
70)
71) String url = String(supabase_url) + "/rest/v1/humidity";
72) http.begin(clientSecure, url);
73) http.addHeader("Content-Type", "application/json");
74) http.addHeader("apikey", supabase_key);
75) http.addHeader("Authorization", "Bearer " +
    String(supabase_key));
76) http.addHeader("Prefer", "return=minimal");
77)
78) String json = "{\"kelembapan\":\"" + String(hum) +
79) (timeStr != "" ? "\",\"tanggal_upload\":\"" + timeStr + "\" :
    "\") +
80) "\",\"status\":\"" + status + "\" +
81) "\",\"status_lampu\":\"" + lamp + "\" +
82) "\",\"status_pompa\":\"" + pump + "\"}";
83)
84) Serial.println("[Supabase] Uploading data...");
85) int httpCode = http.POST(json);
86)
87) if (httpCode >= 200 && httpCode < 300) {
88) Serial.printf("[Supabase] Berhasil Terkirim! Status: %d\n",

```



```

        httpCode);
89) } else if (httpCode > 0) {
90) String response = http.getString();
91) Serial.printf("[Supabase] Gagal Mengirim. HTTP Code: %d\n",
        httpCode);
92) Serial.printf("[Supabase] Alasan: %s\n", response.c_str());
93) } else {
94) Serial.printf("[Supabase] Gagal Koneksi. Error: %s\n",
        http.errorToString(httpCode).c_str());
95) }
96)
97) http.end();
98) } else {
99) Serial.println("[Supabase] Gagal: WiFi tidak terhubung!");
100) }
101) }
102)
103) // ===== WIFI =====
104) void setup_wifi() {
105) delay(10);
106) Serial.println("Connecting to WiFi...");
107)
108) WiFi.begin(ssid, password);
109)
110) while (WiFi.status() != WL_CONNECTED) {
111) delay(500);
112) Serial.print(".");
113) }
114)
115) Serial.println("\nWiFi connected!");
116) }
117)
118) // ===== MQTT CALLBACK =====
119) void callback(char* topic, byte* payload, unsigned int
        length) {
120) String message = "";
121) for (int i = 0; i < length; i++) {
122) message += (char)payload[i];
123) }
124)
125) Serial.print("Message arrived [");
126) Serial.print(topic);
127) Serial.print("] ");
128) Serial.println(message);
129)
130) if (String(topic) == "esp32/statusiqbal/mode") {
131) isAutoMode = (message == "auto");
132) Serial.println(isAutoMode ? "Mode: AUTO" : "Mode: MANUAL");
133) }

```

```

134) else if (String(topic) == "esp32/statusiqbal/pump") {
135) manualPump = (message == "on");
136) Serial.println(manualPump ? "Manual Pump: ON" : "Manual Pump:
    OFF");
137) }
138) else if (String(topic) == "esp32/statusiqbal/light") {
139) manualLight = (message == "on");
140) Serial.println(manualLight ? "Manual Light: ON" : "Manual
    Light: OFF");
141) }
142) }
143)
144) // ===== MQTT RECONNECT =====
145) void reconnect() {
146) while (!client.connected()) {
147) Serial.print("Connecting MQTT...");
148)
149) // Create a unique client ID using MAC address
150) String clientId = "ESP32MJ-" + WiFi.macAddress();
151)
152) if (client.connect(clientId.c_str(), "esp32/statusiqbal", 0,
    true, "offline")) {
153) Serial.println("CONNECTED to MQTT ");
154) client.publish("esp32/statusiqbal", "online", true);
155)
156) // Subscribe to control topics
157) client.subscribe("esp32/statusiqbal/mode");
158) client.subscribe("esp32/statusiqbal/pump");
159) client.subscribe("esp32/statusiqbal/light");
160) } else {
161) Serial.print("FAILED, rc=");
162) Serial.print(client.state());
163) Serial.println(" trying again in 5s...");
164) delay(5000);
165) }
166) }
167) }
168)
169) void setup() {
170) Serial.begin(115200);
171)
172) pinMode(RELAY1, OUTPUT);
173) pinMode(RELAY2, OUTPUT);
174)
175) digitalWrite(RELAY1, RELAY_OFF);
176) digitalWrite(RELAY2, RELAY_OFF);
177)
178) dht.begin();
179)

```

```

180) setup_wifi();
181)
182) // Konfigurasi waktu dari NTP
183) configTime(gmtOffset_sec, daylightOffset_sec, ntpServer);
184)
185) client.setServer(mqtt_server, 1883);
186) client.setCallback(callback);
187) }
188)
189) void loop() {
190) if (!client.connected()) {
191) reconnect();
192) }
193) client.loop();
194)
195) unsigned long now = millis();
196)
197) if (now - lastMsg > 2000) {
198) lastMsg = now;
199)
200) float suhu = dht.readTemperature();
201) float kelembapan = dht.readHumidity();
202)
203) if (isnan(suhu) || isnan(kelembapan)) {
204) Serial.println("Gagal baca DHT!");
205) return;
206) }
207)
208) // ===== CONTROL LOGIC =====
209) if (isAutoMode) {
210) // Logic: <70% Pump ON, Lamp OFF | >90% Lamp ON, Pump OFF |
        70-90% NORMAL (OFF/OFF)
211) if (kelembapan < 70.0) {
212) digitalWrite(RELAY1, RELAY_ON); // Pompa Hidup
213) digitalWrite(RELAY2, RELAY_OFF); // Lampu Mati
214) statusRelay = "KERING / PENYIRAMAN";
215) }
216) else if (kelembapan > 90.0) {
217) digitalWrite(RELAY1, RELAY_OFF); // Pompa Mati
218) digitalWrite(RELAY2, RELAY_ON); // Lampu Nyala
219) statusRelay = "TERLALU LEMBAB";
220) }
221) else {
222) digitalWrite(RELAY1, RELAY_OFF); // Pompa Mati
223) digitalWrite(RELAY2, RELAY_OFF); // Lampu Mati
224) statusRelay = "NORMAL";
225) }
226) } else {
227) // Manual Mode (App Based)

```

```

228) digitalWrite(RELAY1, manualPump ? RELAY_ON : RELAY_OFF);
229) digitalWrite(RELAY2, manualLight ? RELAY_ON : RELAY_OFF);
230) statusRelay = "MANUAL";
231) }
232)
233) // ===== STATUS REPORTING =====
234) bool currentPump = (digitalRead(RELAY1) == RELAY_ON);
235) bool currentLight = (digitalRead(RELAY2) == RELAY_ON);
236)
237) // Upload logic: Action transition OR Periodic (30 mins) if
    Normal
238) unsigned long nowMillis = millis();
239) bool actionDetected = (currentPump != prevPump ||
    currentLight != prevLight);
240)
241) // Interval 30 menit = 1800000 ms
242) bool isPeriodicTime = (nowMillis - lastPeriodicUpload >=
    1800000);
243)
244) if (actionDetected) {
245) Serial.println("[Reporting] Deteksi Perubahan Status!");
246) }
247)
248) if (actionDetected || (statusRelay == "NORMAL" &&
    isPeriodicTime)) {
249) String statusLampuUpload = currentLight ? (isAutoMode ?
    "MENYALA - OTOMATIS" : "MENYALA - MANUAL") : "MATI";
250) String statusPompaUpload = currentPump ? (isAutoMode ?
    "MENYALA - OTOMATIS" : "MENYALA - MANUAL") : "MATI";
251)
252) Serial.println("[Reporting] Memulai Upload ke Supabase...");
253) uploadToSupabase(kelembapan, statusRelay, statusLampuUpload,
    statusPompaUpload);
254) lastPeriodicUpload = nowMillis;
255) }
256)
257) // MQTT Reporting
258) client.publish("esp32/dht      ibapan",
    String(kelembapan).c_str
259) client.publish("esp32/relay/status", statusRelay.c_str());
260) client.publish("esp32/statusiqbal/pump", currentPump ? "on" :
    "off", true);
261) client.publish("esp32/statusiqbal/light", currentLight ? "on"
    : "off", true);
262) client.publish("esp32/statusiqbal/mode", isAutoMode ? "auto"
    : "manual", true);
263)
264) // Save states for next comparison
265) prevPump = currentPump;

```

```
266) prevLight = currentLight;
267)
268) // Serial Debug
269) Serial.print("Mode: "); Serial.print(isAutoMode ? "AUTO" :
    "MANUAL");
270) Serial.print(" | Hum: "); Serial.print(kelembapan);
271) Serial.print("% | Pump: "); Serial.print(currentPump ? "ON" :
    "OFF");
272) Serial.print(" | Light: "); Serial.print(currentLight ? "ON"
    : "OFF");
273) Serial.print(" | Status: "); Serial.println(statusRelay);
274) }
275) }
```