

## IMPLEMENTASI SISTEM PEMANTAUAN DAN PENGENDALIAN PADA BUDIDAYA JAMUR TIRAM BERBASIS *INTERNET OF THINGS* MENGGUNAKAN LOGIKA FUZZY

Irsan Nur Hidayat<sup>\*1</sup>, Muhamad Bahrul Ulum<sup>2</sup>

<sup>1,2</sup>Universitas Esa Unggul Jakarta, Jakarta Barat  
Email: <sup>1</sup>irsannh@student.esaunggul.ac.id, <sup>2</sup>m.bahrul\_ulum@esaunggul.ac.id  
<sup>\*</sup>Penulis Korespondensi

(Naskah masuk: 21 Januari 2025, diterima untuk diterbitkan: 15 Januari 2026)

### Abstrak

Jamur tiram saat ini mulai banyak dibudidayakan oleh masyarakat perkotaan. Namun, suhu dan kelembapan yang ada di perkotaan kurang sesuai untuk pertumbuhan jamur tiram yang membutuhkan suhu 24 – 27 °C dan kelembapan 80 – 90%. Penelitian ini bertujuan untuk mengembangkan sistem berbasis *Internet of Things* dan Logika Fuzzy dengan Metode Tsukamoto yang dapat memantau serta mengendalikan kondisi suhu dan kelembapan pada kumbung jamur tiram. Pengujian membuktikan bahwa sistem mampu memantau kondisi suhu dan kelembapan secara *real time* dan menginformasikannya kepada petani. Selain itu, sistem juga terbukti dapat menjaga kondisi kumbung pada suhu rata-rata 25,96 °C dan kelembapan rata-rata 87,18%. Penggunaan sistem ini juga berdampak positif terhadap keberhasilan pertumbuhan jamur tiram jika dibandingkan dengan jamur tiram dalam kumbung manual yang mengalami gagal panen. Dengan demikian, sistem ini telah berhasil mendukung keberhasilan budidaya jamur tiram pada lingkungan perkotaan.

**Kata kunci:** *Internet of Things, Jamur Tiram, Logika Fuzzy, Metode Fuzzy Tsukamoto*

## IMPLEMENTATION OF MONITORING AND CONTROL SYSTEM FOR OYSTER MUSHROOM CULTIVATION BASED ON INTERNET OF THINGS USING FUZZY LOGIC

### Abstract

Oyster mushrooms are currently being cultivated by many urban communities. However, the temperature and humidity in urban areas are not suitable for the growth of oyster mushrooms, which require a temperature of 24–27 °C and humidity of 80–90%. This study aims to develop an Internet of Things and Fuzzy Logic-based system Fuzzy Logic using the Tsukamoto method to monitor and control temperature and humidity conditions in oyster mushroom greenhouses. Testing demonstrated that the system can monitor temperature and humidity conditions in real time and communicate this information to farmers. Additionally, the system has been proven to maintain the cultivation house conditions at an average temperature of 25.96°C and an average humidity of 87.18%. The use of this system also has a positive impact on the success of oyster mushroom growth when compared to oyster mushrooms in manual sheds that experience crop failure. Thus, this system has successfully supported the cultivation of oyster mushrooms in urban environments.

**Keywords:** *Internet of Things, Oyster Mushroom, Fuzzy Logic, Tsukamoto Fuzzy Method*

### 1. PENDAHULUAN

Indonesia kerap kali dijuluki sebagai negara agraris. Ini disebabkan karena negara ini lekat dengan pertanian. Dengan tanah yang luas serta subur, membuat banyak penduduknya menggantungkan kehidupannya pada sektor ini.

Sebelumnya, pertanian memang identik dilakukan di desa. Akan tetapi, saat ini kultur untuk melakukan kegiatan pertanian di daerah perkotaan mulai berkembang. Konsep ini disebut dengan

istilah pertanian perkotaan atau *urban farming*. Dengan konsep ini, masyarakat dapat memanfaatkan ruang-ruang sempit yang ada di kota untuk dijadikan sebagai lahan pertanian.

Ada banyak jenis tanaman yang dapat tumbuh dengan baik dengan konsep ini. Salah satunya adalah Jamur Tiram. Tanaman yang memiliki nama latin *Pleurotus Ostreatus* ini banyak dibudidayakan karena memang sesuai dengan konsep *urban farming* yang tidak memerlukan lahan besar. Selain itu, harga bibit yang terjangkau juga menjadi alasan

di balik meningkatnya budidaya jamur tiram (Triono, 2020).

Dalam membudidayakan tanaman ini, ada kondisi ideal yang harus dipenuhi. Kondisi tersebut meliputi suhu yang berkisar 24 – 27 derajat celsius dan kelembapan ideal yang berkisar 80 – 90 persen. Apabila kondisi ideal tidak tercapai maka jamur tiram tidak dapat tumbuh dengan baik. Ini akan berdampak pada kualitas panen yang menjadi kurang baik (Pujianto dan Fadlilah, 2023).

Masalah muncul ketika jamur tiram dibudidayakan melalui konsep *urban farming*. Masalah ini ialah kurang idealnya kondisi iklim di daerah perkotaan jika dibandingkan dengan desa. Ini bisa berdampak pada pertumbuhan jamur tiram yang kurang maksimal. Oleh karenanya, diperlukan sebuah sistem yang andal dalam mengendalikan suhu maupun kelembapan pada kumbung tempat budidaya jamur tiram.

Saat ini, *Internet of Things* (IoT) dan kecerdasan artifisial menjadi konsep teknologi yang cukup populer dalam sektor pertanian. Dua teknologi ini dinilai mampu merevolusi sistem pertanian ke arah yang lebih produktif.

IoT sendiri adalah teknologi yang memungkinkan berbagai perangkat elektronik saling terhubung, melakukan komputasi, serta menyimpan data yang diharapkan dapat memenuhi kebutuhan manusia (Wu dan Chen, 2024). Adapun kecerdasan artifisial sendiri didefinisikan sebagai teknologi yang memungkinkan sebuah mesin dapat mengerjakan berbagai tugas yang membutuhkan kepandaian yang dimiliki oleh manusia (Gignac dan Szodorai, 2024).

Dalam sektor pertanian, IoT dan kecerdasan artifisial lazimnya dimanfaatkan dalam proses pengambilan keputusan. Sensor akan mengukur data-data lingkungan seperti suhu dan kelembapan sebelum kemudian dianalisis untuk mengambil keputusan yang diperlukan lingkungan pertanian, misalnya dalam irigasi maupun pengendalian hama (Muhammed et al., 2024).

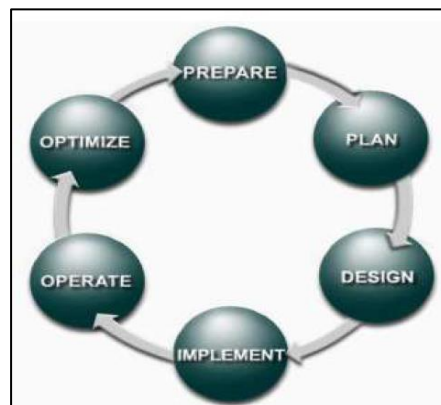
Untuk mengambil keputusan, ada banyak pendekatan yang dapat dimanfaatkann. Satu di antaranya ialah logika *fuzzy*. Konsep ini dipopulerkan oleh Lotfi Zadeh pada 1965. Logika *fuzzy* bermanfaat untuk menyelesaikan masalah yang bersifat tidak pasti. Logika *fuzzy* menggunakan kebenaran parsial dalam menentukan solusi, yang membuat sebuah nilai tidak dapat dianggap benar atau salah sepenuhnya, melainkan dapat di antara keduanya (Shabara et al., 2024).

Sebelum ini, terdapat beberapa penelitian untuk menyelesaikan masalah terkait suhu dan kelembapan dalam bidang pertanian. Dani et al. (2022) dalam penelitiannya memanfaatkan konsep logika *fuzzy* metode *mamdani* untuk mengendalikan kondisi kumbung jamur tiram, di samping penggunaanya juga tetap dapat memantaunya menggunakan platform *blynk*. Logika *fuzzy* metode *mamdani* juga digunakan oleh Ubaidillah, Istiadi dan Mukhsim

(2020) dalam penelitiannya untuk memantau dan mengendalikan kumbung jamur tiram. Hanya saja pemantauannya dilakukan melalui web ESP-LINK. Adapun dalam penelitian lainnya yang dilakukan oleh Irwanto et al. (2024), IoT dan logika *fuzzy* metode *mamdani* juga digunakan dalam pengendalian suhu, kelembapan, kelembapan tanah, hingga intensitas cahaya dalam proses pembuatan media tanam (substrat) pada jamur. Adapun logika *fuzzy* metode *sugeno* digunakan oleh Awi, Rabi dan Dirgantara (2022), dalam penelitiannya untuk mengendalikan suhu, kelembapan, serta pH pada kumbung jamur kuping, di mana datanya sendiri dikirimkan menuju *server Node-Red*.

Berdasarkan uraian masalah di atas maka sebuah penelitian dilakukan guna membuat sistem pemantauan dan pengendalian pada suhu serta kelembapan yang ada pada kumbung jamur tiram, dengan memanfaatkan konsep IoT serta logika *fuzzy* dengan metode *tsukamoto* yang belum banyak digunakan sebelum ini. Selain itu, sebuah aplikasi *mobile* juga akan dibuat dengan menggunakan *flutter* SDK dan *firebase* untuk penyimpanannya. Pembuatan aplikasi *mobile* ini dapat memberikan fleksibilitas berupa tampilan yang lebih sesuai dengan keinginan pengguna, juga memudahkan pengguna jika dibandingkan dengan menggunakan *web*. Pengguna juga dapat mengendalikan aktuator-aktuator yang ada melalui aplikasi tersebut. Melalui penelitian ini, diharapkan agar kondisi suhu serta kelembapan dapat terkendali dan terjaga pada kondisi yang ideal serta akan berdampak pada pertumbuhan serta hasil panen yang lebih maksimal dari jamur tiram itu sendiri.

## 2. METODE PENELITIAN



Gambar 1. Metode PPDIIO

Penelitian ini dilakukan dengan memanfaatkan metode *Prepare, Plan, Design, Implementation, Operate, and Optimize* (PPDIIO). Metode ini sendiri ialah pendekatan penelitian yang dikeluarkan oleh Cisco (Heriyani dan Ernawati, 2024). Metode ini dapat meningkatkan ketersediaan dan menciptakan implementasi yang solid (Haeruddin et al., 2022). Meskipun PPDIIO lazimnya digunakan dalam jaringan komputer, tetapi metode ini juga

dapat digunakan dalam pengembangan sistem IoT, seperti pada penelitian yang dilakukan Heriyani dan Ernawati (2024). Metode ini terbagi atas enam tahapan utama, seperti yang dapat dilihat pada Gambar 1.

### 2.1 Tahap *Prepare*

Pada tahap pertama, yaitu *Prepare*, proses pengumpulan data dilakukan guna mengidentifikasi permasalahan yang terjadi dalam proses pembudidayaan jamur tiram. Pengumpulan data dilakukan dengan tiga metode, yaitu studi literatur, observasi, serta wawancara. Studi literatur dilakukan dengan mengumpulkan data pustaka dari berbagai penelitian sebelumnya. Observasi dilakukan dengan cara melakukan mengamati langsung proses pembudidayaan jamur tiram. Adapun wawancara dilakukan dengan petani yang selama ini membudidayakan jamur tiram guna mengetahui kesulitan yang dialami oleh para petani ketika membudidayakan jamur tiram.

Dari proses pengumpulan data ditemukan bahwa masalah berupa belum adanya sistem otomatis yang dapat memantau serta mengendalikan kondisi suhu dan kelembapan yang ada pada kumbung jamur tiram secara akurat. Selama ini, petani hanya melakukan pengendalian manual yang dianggap tidak begitu efektif serta efisien disebabkan hanya berdasarkan intuisi dan hanya dilakukan terjadwal selama dua hingga tiga kali sehari. Apabila terjadi perubahan kondisi secara mendadak di luar jadwal pengendalian maka tidak ada sistem yang bisa merespons perubahan kondisi tersebut secara cepat. Kondisi ini akan menghambat budidaya itu sendiri.

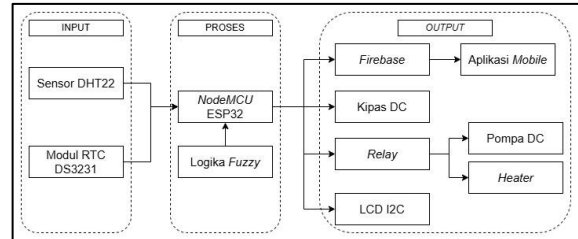
### 2.2 Tahap *Plan*

Pada tahap kedua atau tahap *plan*, identifikasi kebutuhan dilakukan berdasarkan analisis masalah pada tahap sebelumnya. Identifikasi kebutuhan meliputi kebutuhan fungsional, non-fungsional, serta komponen yang akan digunakan.

Dari sisi kebutuhan fungsional, sistem harus mampu mendeteksi kondisi suhu dan kelembapan melalui sensor dan dapat menginformasikannya kepada pengguna, baik melalui aplikasi *mobile* maupun melalui layar yang disediakan pada kumbung jamur tiram. Selain itu, sistem juga harus mampu melakukan pengendalian terhadap kondisi suhu serta kelembapan secara otomatis dan akurat dengan kondisi ideal yang diharapkan, berdasarkan data yang didapatkan dari hasil pemantauan yang dilaksanakan.

Dari sisi kebutuhan non-fungsional, sistem harus mampu beroperasi selama sehari penuh. Selain itu, sistem juga harus dapat diintegrasikan dengan aplikasi *mobile* agar memudahkan petani dalam memantau kondisi kumbung melalui ponsel yang dimilikinya.

Adapun dari sisi kebutuhan perangkat atau komponen yang akan digunakan, sistem akan dibangun berdasarkan beberapa komponen elektronika, sebagaimana yang terdapat pada Blok Diagram yang ada pada Gambar 2.



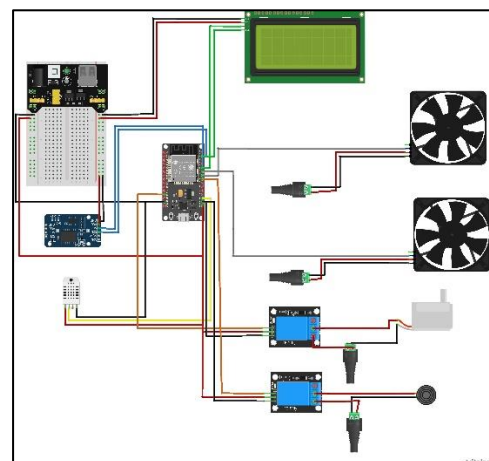
Gambar 2. Blok Diagram Sistem

Sensor DHT22 berperan untuk mendeteksi kondisi suhu dan kelembapan yang ada, ditambah Modul RTC DS3231 yang digunakan untuk mendeteksi waktu ketika pengukuran berlangsung. Data-data yang didapatkan dari dua komponen tersebut akan diproses oleh mikrokontroler *NodeMCU ESP32*. Pertama, data dikirimkan ke basis data *Firestore Realtime Database*, agar dapat diakses oleh pengguna melalui aplikasi *mobile*. Selepas itu, mikrokontroler juga akan membuat keputusan pengendalian dengan memanfaatkan konsep logika *fuzzy*. Dengan logika *fuzzy*, mikrokontroler akan menentukan kecepatan kipas, durasi pompa, serta durasi *heater* guna mengendalikan suhu dan kelembapan yang ada.

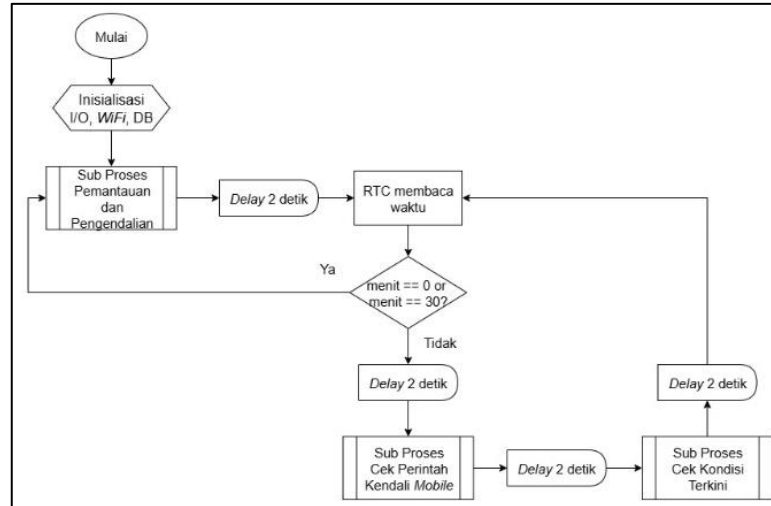
### 2.3 Tahap *Design*

Pada tahap ketiga atau tahap *design*, serangkaian perancangan akan dilakukan. Perancangan tersebut meliputi komponen elektronika, alur kerja sistem, basis data, antarmuka yang akan digunakan, wadah serta tata letak yang akan digunakan, hingga dari sisi logika *fuzzy* yang akan digunakan.

#### 2.3.1 Desain Komponen Elektronika



Gambar 3. Desain Komponen Elektronika

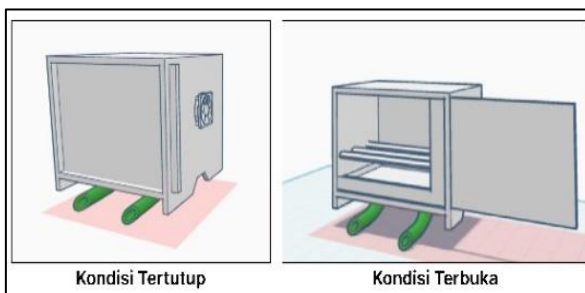


Gambar 6. Flowchart Utama Sistem

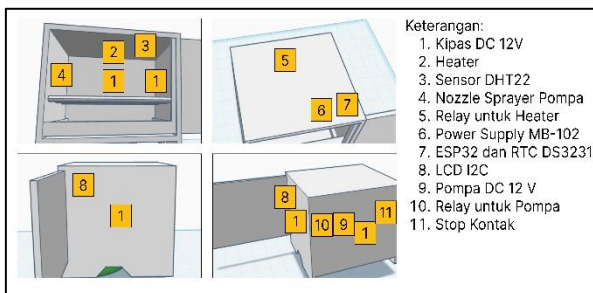
Desain komponen elektronika bertujuan untuk merancang bagaimana komponen elektronika yang akan digunakan saling terkoneksi. Desain ini merupakan penjawantahan dari blok diagram yang sebelumnya telah dibuat. Desain ini sendiri dapat dilihat pada Gambar 3.

### 2.3.2 Desain Mekanik

Desain mekanik berisi rancangan wadah beserta penempatan komponen-komponen yang akan digunakan dalam sistem ini. Adapun desain mekanik untuk wadah beserta tata letak dari komponen-komponen yang digunakan dapat dilihat pada Gambar 4 dan 5.



Gambar 4. Desain Wadah Sistem



Gambar 5. Desain Tata Letak Komponen

### 2.3.2 Desain Alur Kerja Sistem

Desain alur kerja ini menggambarkan bagaimana sistem akan bekerja nantinya. Desain alur

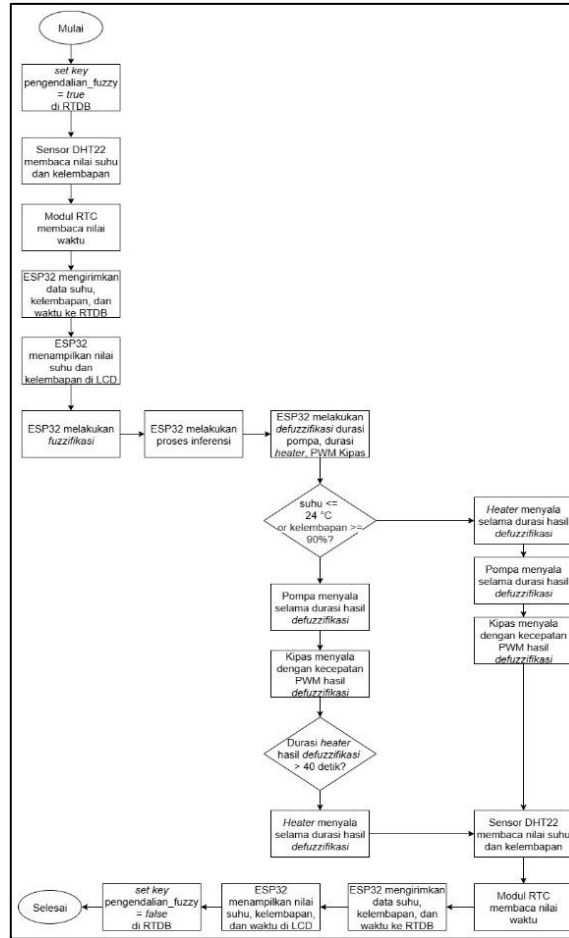
kerja sistem ini digambarkan dengan *flowchart* atau diagram alir. *Flowchart* ini dapat dilihat pada Gambar 6.

Sistem ini sendiri memiliki tiga sub-proses utama yang berjalan. Tiga Sub-proses ini terdiri dari sub-proses pemantauan dan pengendalian, sub-proses cek perintah kendali manual, serta sub-proses cek kondisi terkini.

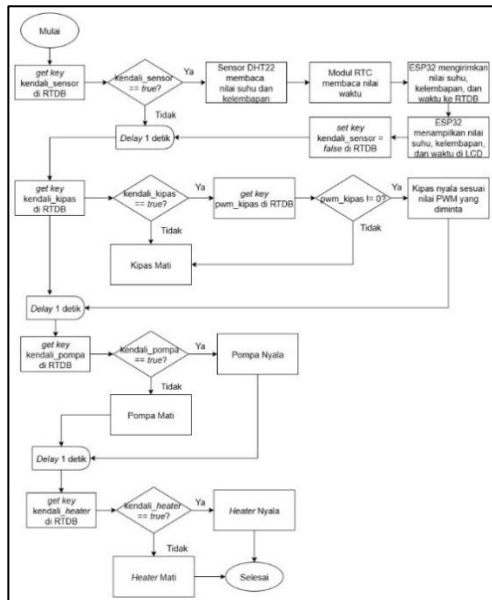
Sub-proses pemantauan dan pengendalian adalah sub-proses yang didefinisikan sebagai proses yang dilakukan oleh mikrokontroler untuk mengukur nilai suhu dan kelembapan terkini pada kumbung jamur tiram, menyimpan hasil pengukuran ke dalam *firebase realtime database*, dan menampilkan nilai hasil pengukuran ke LCD yang terletak di kumbung jamur tiram. Selain itu, mikrokontroler juga melakukan inferensi menggunakan logika *fuzzy* serta melakukan pengendalian berdasarkan hasil inferensi yang dilakukan. Terdapat tiga aktuator yang digunakan, yaitu Kipas, Pompa, dan *Heater*. *Heater* hanya akan menyala jika *output* durasi pompa yang dihasilkan dari logika *fuzzy* bernilai lebih dari 40 detik. Selain itu, *heater* juga akan menyala terlebih dahulu apabila suhu 24 °C atau kurang serta kelembapan lebih dari 90%. Jika kedua kondisi tersebut tidak terjadi maka hanya pompa dan kipas saja yang menyala secara bersamaan berdasarkan nilai durasi pompa dan PWM kipas yang didapatkan sebelumnya. *Flowchart* sub-proses ini sendiri dapat dilihat pada Gambar 7.

Sub-proses kedua adalah sub-proses cek perintah kendali *mobile*. Sub-proses ini bertujuan untuk menjabarkan proses yang dilakukan oleh mikrokontroler untuk mengecek apakah terdapat perintah yang diberikan melalui aplikasi *mobile*.

Perintah yang dapat diberikan oleh aplikasi *mobile* meliputi pengecekan kondisi suhu dan kelembapan terkini, menyalakan kipas, pompa, atau *heater*. Adapun *flowchart* sub-proses yang kedua dapat dilihat pada Gambar 8.



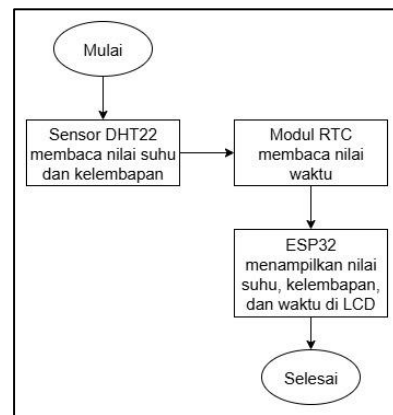
Gambar 7. Flowchart Sub-Proses Pemantauan dan Pengendalian



Gambar 8. Flowchart Sub-Proses Cek Perintah Kendali Mobile

Sub-proses terakhir dari keseluruhan sistem ini adalah sub-proses cek kondisi terkini. Sub-proses ini bertujuan untuk melakukan pengukuran suhu dan kelembapan secara berkala untuk ditampilkan ke LCD yang berada di kumbung jamur tiram yang

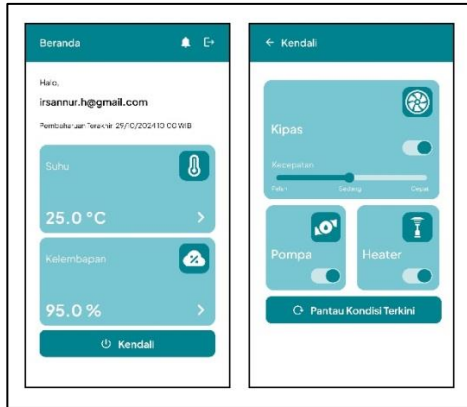
digunakan. Flowchart sub-proses terakhir ini dapat dilihat pada Gambar 9.



Gambar 9. Flowchart Sub-Proses Cek Kondisi Terkini

### 2.3.3 Desain Antarmuka Aplikasi Mobile

Desain antarmuka aplikasi mobile menggambarkan tata letak menu dan data yang akan ditampilkan pada pengguna dalam aplikasi mobile yang dibuat. Desain antarmuka dibuat dengan menggunakan tools Figma. Adapun desain antarmukanya dapat dilihat pada Gambar 10.



Gambar 10. Desain Antarmuka Aplikasi Mobile

### 2.3.4 Desain Logika Fuzzy

Desain terakhir yang dilakukan adalah logika fuzzy. Logika fuzzy akan digunakan untuk membuat keputusan terkait pengendalian suhu dan kelembapan. Dalam merancang logika fuzzy, ada beberapa pendekatan atau metode yang bisa dimanfaatkan. Adapun penelitian ini sendiri menggunakan metode *tsukamoto*. Metode ini sendiri menggunakan rata-rata terbobot dalam penghitungan hasil akhirnya (Kuswanto et al., 2023).

Dalam merancang logika fuzzy dengan pendekatan *tsukamoto*, ada lima tahapan yang mesti dilakukan, yaitu pembentukan himpunan fuzzy, fuzzifikasi, pembentukan aturan, inferensi, dan defuzzifikasi yang menjadi langkah terakhir dari keseluruhan proses ini.

Pembentukan himpunan fuzzy dilakukan untuk merepresentasikan variabel-variabel fuzzy yang digunakan (Setiyawan, Arbansyah dan Latipah, 2023). Himpunan fuzzy yang digunakan untuk merepresentasikan variabel input dan output pada sistem ini dapat dilihat melalui Tabel 1 dan 2.

Tabel 1. Pembentukan Himpunan Fuzzy (Variabel Input)

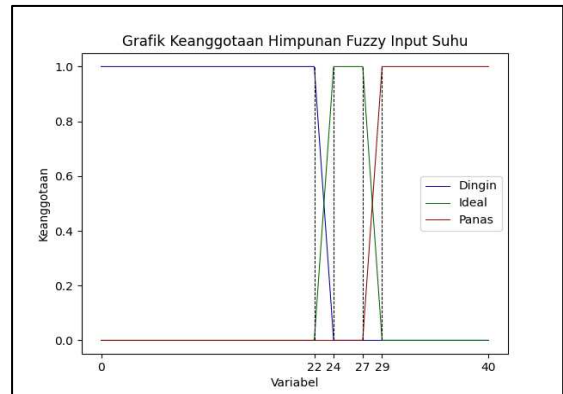
Variabel	Satuan	Himpunan	Domain
Suhu	°C	Dingin	0-24
		Ideal	22-29
		Panas	27-40
		Rendah	0-60
Kelembapan	%	Sedang	55-80
		Ideal	75-95
		Tinggi	90-100

Tabel 2. Pembentukan Himpunan Fuzzy (Variabel Output)

Variabel	Satuan	Himpunan	Domain
Kecepatan Kipas	PWM	Pelan	0-140
		Agak Cepat	60-220
		Cepat	140-255
		Sangat Sebentar	0-180
Durasi Pompa	detik	Agak Sebentar	90-270
		Sebentar	180-420
		Agak Lama	300-540
		Lama	420-640
Durasi Heater	detik	Mati	0-40
		Sangat Sebentar	30-100
		Sebentar	90-180
		Agak Lama	170-270
		Lama	260-300

Tahap berikutnya adalah *fuzzifikasi*, yang merupakan proses untuk mengubah nilai input yang berbentuk *crisp* menjadi nilai *fuzzy* yang berupa derajat keanggotaan dari nilai *crisp* tersebut. Nilai derajat keanggotaan ini diperoleh melalui fungsi keanggotaan yang direpresentasikan dengan bentuk kurva (Setiyawan, Arbansyah dan Latipah, 2023).

Adapun *fuzzifikasi* dari input suhu dan kelembapan beserta fungsi keanggotaannya dapat dilihat pada Gambar 11 dan 12, serta Persamaan (1) hingga (7).

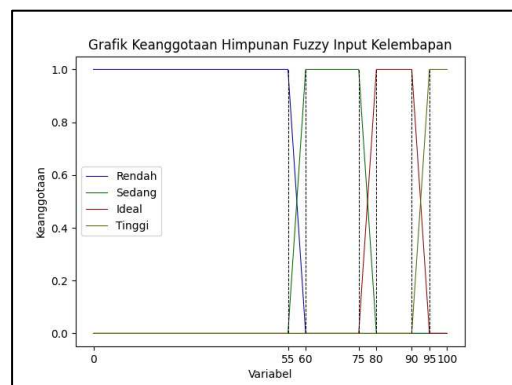


Gambar 11. Kurva Fuzzy Variabel Suhu

$$\mu_{Dingin}[x] = \begin{cases} 0; & x \geq 24 \\ \frac{24-x}{24-22}; & 22 < x < 24 \\ 1; & x \leq 22 \end{cases} \quad (1)$$

$$\mu_{Ideal}[x] = \begin{cases} 0; & x \geq 29 \text{ atau } x \leq 22 \\ \frac{x-22}{24-22}; & 22 < x < 24 \\ \frac{29-x}{29-27}; & 27 < x < 29 \\ 1; & 24 \leq x \leq 27 \end{cases} \quad (2)$$

$$\mu_{Panas}[x] = \begin{cases} 0; & x \leq 27 \\ \frac{29-x}{29-27}; & 27 < x < 29 \\ 1; & x \geq 29 \end{cases} \quad (3)$$



Gambar 12. Kurva Fuzzy Variabel Kelembapan

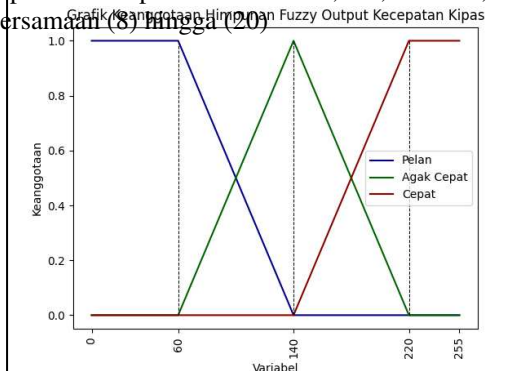
$$\mu_{Rendah}[x] = \begin{cases} 0; x \geq 60 \\ \frac{60-x}{60-55}; 55 < x < 60 \\ 1; x \leq 55 \end{cases} \quad (4)$$

$$\mu_{Sedang}[x] = \begin{cases} 0; x \geq 80 \text{ atau } x \leq 55 \\ \frac{x-55}{60-55}; 55 < x < 60 \\ \frac{80-x}{80-75}; 75 < x < 80 \\ 1; 60 \leq x \leq 75 \end{cases} \quad (5)$$

$$\mu_{Ideal}[x] = \begin{cases} 0; x \geq 95 \text{ atau } x \leq 75 \\ \frac{x-75}{80-75}; 75 < x < 80 \\ \frac{95-x}{95-90}; 90 < x < 95 \\ 1; 80 \leq x \leq 90 \end{cases} \quad (6)$$

$$\mu_{Tinggi}[x] = \begin{cases} 0; x \leq 90 \\ \frac{95-x}{95-90}; 90 < x < 95 \\ 1; x \geq 95 \end{cases} \quad (7)$$

Selain *fuzzifikasi* untuk input, variabel-variabel *output* yang ada juga perlu dibentuk fungsi keanggotaannya. Dalam hal ini, variabel *output*-nya adalah PWM Kipas untuk mengatur kecepatan kipas, durasi pompa dan durasi *heater* yang bertujuan untuk mengatur seberapa lama aktuator pompa dan *heater* menyala. Durasi pompa dan *heater* akan menggunakan satuan detik. Adapun fungsi keanggotaan dari masing-masing variabel *output* dapat dilihat pada Gambar 13, 14, dan 15, serta Persamaan (8) hingga (20).

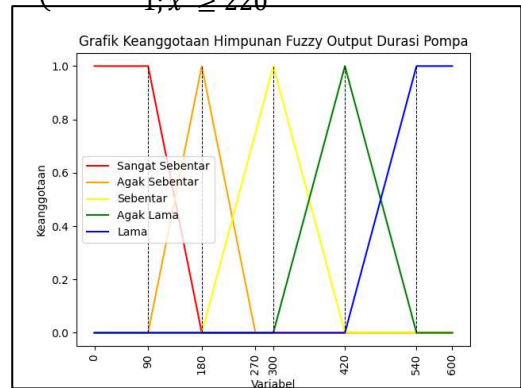


Gambar 13. Kurva Fuzzy Variabel Kecepatan Kipas

$$\mu_{Pelana}[z] = \begin{cases} 0; z \geq 140 \\ \frac{140-z}{140-60}; 60 < z < 140 \\ 1; z \leq 60 \end{cases} \quad (8)$$

$$\mu_{Agak\ Cepat}[z] = \begin{cases} 0; z \geq 220 \text{ atau } z \leq 60 \\ \frac{z-60}{140-60}; 60 < z < 140 \\ \frac{z-140}{220-140}; 140 < z < 220 \\ 1; z = 140 \end{cases} \quad (9)$$

$$\mu_{Cepat}[z] = \begin{cases} 0; z \leq 140 \\ \frac{z-140}{220-140}; 140 < z < 220 \\ 1; x \geq 220 \end{cases} \quad (10)$$



Gambar 14. Kurva Fuzzy Variabel Durasi Pompa

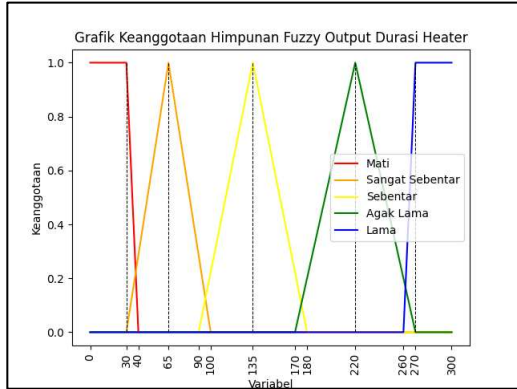
$$\mu_{Sangat\ Sebentar}[z] = \begin{cases} 0; z \geq 180 \\ \frac{180-z}{180-90}; 90 < z < 180 \\ 1; z \leq 90 \end{cases} \quad (11)$$

$$\mu_{Agak\ Sebentar}[z] = \begin{cases} 0; z \geq 270 \text{ atau } z \leq 180 \\ \frac{z-90}{180-90}; 90 < z < 180 \\ \frac{z-180}{270-180}; 180 < z < 270 \\ 1; z = 180 \end{cases} \quad (12)$$

$$\mu_{Sebentar}[z] = \begin{cases} 0; z \geq 420 \text{ atau } z \leq 180 \\ \frac{z-180}{300-180}; 180 < z < 300 \\ \frac{z-300}{420-300}; 300 < z < 420 \\ 1; z = 300 \end{cases} \quad (13)$$

$$\mu_{Agak\ Lama}[z] = \begin{cases} 0; z \geq 540 \text{ atau } z \leq 300 \\ \frac{z - 300}{420 - 300}; 300 < z < 420 \\ \frac{z - 420}{540 - 420}; 420 < z < 540 \\ 1; z = 420 \end{cases} \quad (14)$$

$$\mu_{Lama}[z] = \begin{cases} 0; z \leq 420 \\ \frac{z - 420}{540 - 420}; 420 < z < 540 \\ 1; z \geq 540 \end{cases} \quad (15)$$



Gambar 15. Kurva Fuzzy Variabel Durasi Heater

$$\mu_{Mati}[z] = \begin{cases} 0; z \geq 40 \\ \frac{40 - z}{40 - 30}; 30 < z < 40 \\ 1; z \leq 30 \end{cases} \quad (16)$$

$$\mu_{Sangat\ Sebarang}[z] = \begin{cases} 0; z \geq 100 \text{ atau } z \leq 30 \\ \frac{z - 30}{65 - 30}; 30 < z < 65 \end{cases} \quad (17)$$

$$\mu_{Sebarang}[z] = \begin{cases} 0; z \geq 180 \text{ atau } z \leq 90 \\ \frac{z - 90}{135 - 90}; 90 < z < 135 \\ \frac{z - 135}{180 - 135}; 135 < z < 180 \\ 1; z = 135 \end{cases} \quad (18)$$

$$\mu_{Agak\ Lama}[z] = \begin{cases} 0; z \geq 270 \text{ atau } z \leq 170 \\ \frac{z - 170}{220 - 170}; 170 < z < 220 \\ \frac{z - 220}{270 - 220}; 220 < z < 270 \\ 1; z = 220 \end{cases} \quad (19)$$

$$\mu_{Lama}[z] = \begin{cases} 0; z \leq 260 \\ \frac{z - 260}{270 - 260}; 260 < z < 270 \\ 1; x \geq 270 \end{cases} \quad (20)$$

Setelah *fuzzifikasi* dilakukan maka tahap berikutnya adalah membentuk aturan atau *rules*.

Aturan dibentuk dari kombinasi himpunan-himpunan pada variabel masukan atau input (Setiyawan, Arbansyah dan Latipah, 2023). Aturan ini bertujuan untuk mengevaluasi masing-masing derajat keanggotaan yang ada di setiap fungsi keanggotaan input. Nantinya, hasil evaluasi ini akan digunakan dalam menentukan hasil nilai *output* yang ada (Ubaidillah, Istiadi dan Mukhsim, 2020). Adapun aturan yang akan digunakan dalam sistem ini dapat dilihat pada Tabel 3.

Tabel 3. Pembentukan Aturan Logika Fuzzy

Input		Output		
Suhu	Kelembapan	Kecepatan Kipas	Durasi Pompa	Durasi Heater
Dingin	Rendah	Pelan	Lama	Lama
Dingin	Sedang	Pelan	Sebarang	Agak Lama
Dingin	Ideal	Pelan	Sangat Sebarang	Agak Lama
Dingin	Tinggi	Pelan	Mati	Sebarang
Ideal	Rendah	Agak Cepat	Lama	Mati
Ideal	Sedang	Agak Cepat	Agak Lama	Mati
Ideal	Ideal	Agak Cepat	Sangat Sebarang	Mati
Ideal	Tinggi	Agak Cepat	Mati	Sangat Sebarang
Panas	Rendah	Cepat	Lama	Mati
Panas	Sedang	Cepat	Agak Lama	Mati
Panas	Ideal	Cepat	Sebarang	Mati
Panas	Tinggi	Cepat	Sebarang	Sangat Sebarang

Tahapan berikutnya adalah inferensi. Tahap ini bertujuan untuk menerapkan aturan terhadap input yang diberikan dari proses *fuzzifikasi* (Lasut, Oslan dan Wibowo, 2024). *Output fuzzy* ini sendiri disebut dengan *fire strength*, yang ditentukan dari operasi dengan operator *AND* yang digunakan untuk membandingkan derajat keanggotaan dari masing-masing input (Sunardi, Yudhana dan Furizal, 2023). Adapun operasi ini sendiri dapat dilihat pada Persamaan (21).

$$\mu[z]_i = \min(\mu_{Suhu}[x], \mu_{Kelembapan}[x]) \quad (21)$$

Keterangan:

$\mu[z]_i$  = nilai *fire strength*

Min = nilai minimal

$\mu_{Suhu}[x]$  = nilai derajat keanggotaan input suhu

$\mu_{Kelembapan}[x]$  = nilai derajat keanggotaan input kelembapan

Tahapan yang paling akhir dari keseluruhan proses perancangan logika *fuzzy* ini adalah *defuzzifikasi*. Tahap ini bertujuan untuk mengubah *output* berbentuk *fuzzy* yang didapatkan dari tahap inferensi ke dalam nilai *output* yang berbentuk *crisp* (Ubaidillah, Istiadi dan Mukhsim, 2020). Dalam hal

ini, rata-rata terbobot dari keseluruhan *output* berbentuk *fuzzy* yang berasal dari seluruh aturan yang ada digunakan untuk *defuzzifikasi*, dengan persamaan yang digunakan adalah sebagai berikut.

$$Z = \frac{\sum \mu[z]_i \times z_i}{\sum \mu[z]_i} \quad (22)$$

Keterangan:

$Z$  = nilai *output*

$\mu[z]_i$  = nilai *fire strength* berdasarkan *rule* yang ada

$z_i$  = Nilai *output* sesuai *rule* atau aturan *fuzzy* yang ada.

## 2.4 Tahap Implement

Pada tahap keempat, atau tahap *implement*, penerapan akan dilakukan terhadap rancangan yang telah ditetapkan di tahap *design*. Implementasi dijalankan dengan memprogram mikrokontroler yang akan digunakan. Setelah itu, integrasi mikrokontroler akan dilakukan dengan komponen-komponen lain yang akan digunakan. Pemograman juga dilakukan untuk mengembangkan aplikasi *mobile* yang juga akan digunakan pada sistem ini.

## 2.5 Tahap Operate

Tahap kelima adalah Tahap *Operate*. Tahap ini bertujuan untuk mengoperasikan sistem yang sebelumnya telah diimplementasikan. Selain itu, pengujian juga akan dilakukan terhadap sistem. Serangkaian pengujian dilaksanakan guna mengetahui tingkat keberhasilan sistem dalam menyelesaikan masalah yang sebelumnya telah diidentifikasi. Pengujian tersebut meliputi pengujian koneksi, pengujian pengiriman data, pengujian akurasi logika *fuzzy*, pengujian efektivitas sistem dalam menjaga kondisi suhu dan kelembapan, pengujian perbandingan dengan kumbung manual, serta pengujian penghitungan konsumsi listrik dan air.

Pengujian koneksi bertujuan untuk mengetahui lama waktu yang dibutuhkan mikrokontroler untuk dapat terkoneksi dengan internet. Pengujian akan dilakukan dalam rentang 1 – 20 meter dari modem *WiFi*. Pengujian dinyatakan berhasil apabila mikrokontroler dapat terhubung dengan *WiFi* dalam waktu kurang dari 30 detik.

Selain pengujian koneksi, pengujian lainnya yang dilaksanakan adalah pengujian pengiriman data. Ini dilakukan guna mengetahui berapa lama waktu yang dibutuhkan oleh mikrokontroler dalam mengirimkan data menuju ke basis data yang digunakan, yaitu *Firestore Realtime Database*. Sama seperti pada pengujian koneksi, pengujian juga dilakukan pada rentang 1 – 20 meter dari modem *WiFi*. Pengujian dinyatakan berhasil apabila

mikrokontroler berhasil mengirim data menuju basis data dalam waktu kurang dari 20 detik.

Pengujian lainnya adalah pengujian respons sistem. Pengujian ini bertujuan guna mengetahui waktu yang dibutuhkan mikrokontroler dalam merespon perintah untuk menyalakan aktuator kipas, pompa, dan *heater* dari aplikasi *mobile*. Pengujian dinyatakan berhasil jika aktuator dapat menyala kurang dari 20 detik.

Pengujian berikutnya adalah akurasi penghitungan logika *fuzzy*. Pengujian ini dilakukan untuk menguji keakuratan hasil penghitungan logika *fuzzy* dengan program referensi. Dalam hal ini, sebuah program berbasis *python* dibuat untuk menjadi referensi pembandingan dengan hasil penghitungan logika *fuzzy* yang dilakukan oleh mikrokontroler. Untuk menghitung akurasi hasil penghitungan logika *fuzzy*, persamaan yang digunakan adalah sebagai berikut.

$$\text{Akurasi} = \frac{\sum \text{Data Akurat}}{\sum \text{Seluruh Data}} \times 100\% \quad (23)$$

Selain pengujian akurasi logika *fuzzy*, pengujian lainnya yang akan dilakukan adalah pengujian efektivitas sistem. Pengujian dilakukan dengan melihat bagaimana kemampuan sistem dalam menjaga kondisi ideal suhu dan kelembapan pada kumbung jamur tiram, yaitu dalam hal ini adalah 24 – 27 derajat celsius dan 80 – 90 persen berdasarkan pengoperasiannya selama 24 jam.

Pengujian keenam yang dilaksanakan penelitian ini adalah dengan mengomparasikan hasil panen jamur tiram yang dibudidayakan dengan sistem ini dengan jamur tiram yang dibudidayakan secara manual. Dalam hal ini, proses pengendalian terhadap suhu dan kelembapan secara manual dilakukan dengan cara melakukan penyiraman sebanyak tiga kali sehari secara teratur.

Pengujian terakhir yang dilaksanakan adalah pengujian untuk menghitung konsumsi listrik dan air dalam satu hari pemakaian. Hasil penghitungan ini dapat menjadi dasar untuk estimasi penghitungan biaya listrik dan air selama sebulan. Ini dapat memudahkan petani dalam memperkirakan berapa biaya operasional yang harus dikeluarkan setiap bulannya. Adapun persamaan yang digunakan dalam menghitung konsumsi listrik dan air sendiri adalah sebagai berikut.

$$\begin{aligned} \text{Energi (kWh)} \\ = \frac{\text{Daya (Watt)} \times \text{Waktu (Jam)}}{1000} \end{aligned} \quad (24)$$

$$\begin{aligned} \text{Volume (m}^3\text{)} \\ = \frac{\text{Debit (} \frac{\text{liter}}{\text{menit}} \text{)} \times \text{Waktu (menit)}}{1000} \end{aligned} \quad (25)$$

**2.6 Tahap Optimize**

Pada tahap keenam yang sekaligus tahapan terakhir ini, identifikasi kekurangan akan dilakukan terhadap sistem yang telah berjalan. Identifikasi kekurangan ini penting untuk dilakukan sebagai bahan evaluasi, bentuk pencegahan terhadap masalah baru di kemudian hari, serta rekomendasi perbaikan pada penelitian selanjutnya.

**3. HASIL DAN PEMBAHASAN**

**3.1 Implementasi Sistem**

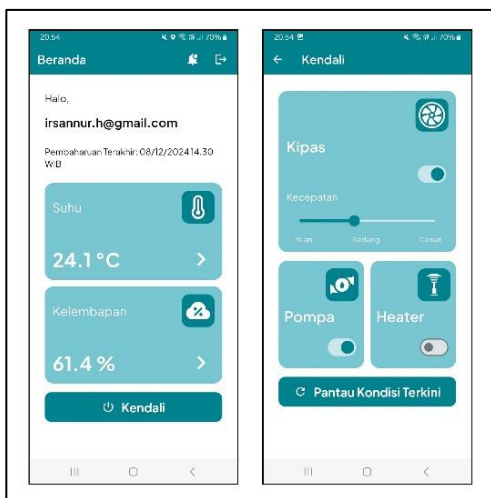
Implementasi sistem berupa wadah (*box*) yang digunakan sebagai sistem pemantauan dan pengendalian pada budidaya jamur tiram berbasis IoT menggunakan logika *fuzzy*. Wadah ini dapat dilihat pada Gambar 16 dan 17.



Gambar 16. Implementasi Sistem (Tampak Luar)



Gambar 17. Implementasi Sistem (Tampak Dalam)



Gambar 18. Implementasi Aplikasi Mobile

Triplek berketebalan 12 milimeter berukuran 47 x 40 x 40 sentimeter digunakan sebagai bahan untuk implementasi sistem. Adapun di dalam *box* yang dibuat, aluminium *foil* digunakan untuk melapisi

triplek. Bahan ini dipilih karena sifatnya yang mampu meredam panas. Penggunaan aluminium *foil* diharapkan membuat suhu di dalam *box* sistem akan tetap stabil.

Selain itu, implementasi juga dilakukan pada aplikasi *mobile*. Aplikasi *mobile* ini dibuat dengan memanfaatkan *Flutter SDK*. Implementasi aplikasi *mobile* ini sendiri dapat dilihat pada Gambar 18.

Dengan aplikasi *mobile* ini, petani selaku pengguna sistem ini dapat melakukan proses pemantauan kondisi suhu serta kelembapan pada kumbungnya secara berkala. Selain itu, petani juga dapat mengendalikan aktuator-aktuator yang digunakan dalam sistem ini.

**3.2 Pengujian Sistem**

**3.2.1 Pengujian Koneksi**

Hasil pengujian koneksi mikrokontroler terhadap *WiFi* yang telah dilaksanakan dapat dilihat pada Tabel 4.

Tabel 4. Hasil Pengujian Koneksi

Pengujian	Pengujian Ke-	Waktu (detik)	Rata-Rata (detik)	Standar Deviasi (detik)
0 – 1 meter	1	0,592	1,191	0,547
	2	1,592		
	3	1,591		
	4	1,591		
	5	0,592		
1 – 5 meter	1	0,591	1,805	1,098
	2	3,605		
	3	1,652		
	4	1,592		
	5	1,592		
6 – 10 meter	1	0,592	1,910	1,298
	2	1,608		
	3	1,654		
	4	4,091		
	5	1,608		
11 – 15 meter	1	0,652	1,206	0,543
	2	1,607		
	3	1,592		
	4	0,592		
	5	1,591		
16 – 20 meter	1	0,642	2,504	2,345
	2	1,591		
	3	6,593		
	4	2,107		
	5	1,591		

Berdasarkan hasil pengujian yang telah dilaksanakan, didapatkan temuan bahwa mikrokontroler dapat terkoneksi dengan *WiFi* dalam waktu kurang dari 30 detik, di mana mikrokontroler rata-rata bahkan dapat terkoneksi dalam waktu kurang dari 5 detik.

**3.2.2 Pengujian Pengiriman Data**

Hasil pengujian pengiriman data yang telah dilakukan dapat dilihat pada Tabel 5.

Dari hasil pengujian yang telah dilaksanakan, menunjukkan bahwa mikrokontroler mampu mengirimkan data menuju *Firebase Realtime*

Database dalam waktu kurang dari 20 detik, dengan keseluruhan rata-rata hasil pengujian pada setiap rentangnya bahkan masih berada di bawah satu detik.

Tabel 5. Hasil Pengujian Pengiriman Data

Pengujian	Pengujian Ke-	Waktu (detik)	Rata-Rata (detik)	Standar Deviasi (detik)
0 – 1 meter	1	0,447	0,315	0,137
	2	0,298		
	3	0,372		
	4	0,365		
	5	0,093		
1 – 5 meter	1	0,653	0,416	0,157
	2	0,277		
	3	0,432		
	4	0,440		
	5	0,279		
6 – 10 meter	1	0,775	0,556	0,295
	2	0,125		
	3	0,421		
	4	0,664		
	5	0,795		
11 – 15 meter	1	1,164	0,813	0,399
	2	0,377		
	3	0,414		
	4	1,006		
	5	1,107		
16 – 20 meter	1	0,925	0,684	0,194
	2	0,605		
	3	0,768		
	4	0,715		
	5	0,405		

### 3.2.3 Pengujian Respons Sistem

Hasil pengujian respons sistem terhadap perintah yang diberikan melalui aplikasi *mobile* terhadap masing-masing aktuatur dapat dilihat pada Tabel 6.

Hasil pengujian menunjukkan bahwa sistem mampu memberikan respons terhadap perintah yang diberikan dalam waktu kurang dari 20 detik untuk menyalakan aktuatur-aktuatur yang digunakan, yaitu kipas, pompa, serta *heater*. Secara umum sistem memang tidak dapat merespon dalam waktu di bawah satu detik, karena mikrokontroler harus menjalankan kode programnya satu persatu, atau dengan kata lain pemrogramannya memang belum dibuat secara konkuren.

Tabel 6. Hasil Pengujian Respons Sistem

Pengujian ke-	Respon Aktuatur (detik)		
	Kipas	Pompa	Heater
1	3,250	3,380	6,560
2	3,600	2,850	7,220
3	2,750	2,920	4,170
4	5,140	2,060	6,430
5	4,300	6,550	1,060
<b>Rata-Rata (detik)</b>	3,800	3,550	5,080
<b>Standar Deviasi (detik)</b>	0,934	1,742	2,529

### 3.2.4 Pengujian Akurasi Penghitungan Logika Fuzzy

Hasil pengujian akurasi penghitungan logika *fuzzy* yang telah dilakukan untuk seluruh *output*, yaitu

PWM Kipas, Durasi Pompa, dan Durasi *Heater* dapat dilihat pada Tabel 7, 8, dan 9.

Tabel 7. Hasil Pengujian Akurasi Penghitungan PWM Kipas

No	Input		Output (PWM)		Hasil
	Suhu (°C)	Kelembapan (%)	ESP 32	Python	
1	27,30	83,00	168	168	Akurat
2	27,20	85,00	172	172	Akurat
3	27,40	85,00	164	164	Akurat
4	27,50	85,00	160	160	Akurat
5	26,80	85,00	140	140	Akurat
6	27,30	86,00	168	168	Akurat
7	27,20	88,00	172	172	Akurat
8	27,00	85,00	140	140	Akurat
9	27,00	84,00	140	140	Akurat
10	26,80	80,00	140	140	Akurat
11	27,60	84,00	156	156	Akurat
12	27,40	79,00	154	154	Akurat
13	27,30	77,00	142	142	Akurat
14	26,10	75,00	140	140	Akurat
15	25,50	81,00	140	140	Akurat

Tabel 8. Hasil Pengujian Akurasi Penghitungan Durasi Pompa

No	Input		Output (detik)		Hasil
	Suhu (°C)	Kelembapan (%)	ESP32	Python	
1	27,30	83,00	252,75	252,75	Akurat
2	27,20	85,00	253,5	253,5	Akurat
3	27,40	85,00	252	252	Akurat
4	27,50	85,00	251,25	251,25	Akurat
5	26,80	85,00	90	90	Akurat
6	27,30	86,00	252,75	252,75	Akurat
7	27,20	88,00	253,5	253,5	Akurat
8	27,00	85,00	90	90	Akurat
9	27,00	84,00	90	90	Akurat
10	26,80	80,00	90	90	Akurat
11	27,60	84,00	250,5	250,5	Akurat
12	27,40	79,00	278,4	278,4	Akurat
13	27,30	77,00	357,6	357,6	Akurat
14	26,10	75,00	420	420	Akurat
15	25,50	81,00	90	90	Akurat

Tabel 9. Hasil Pengujian Akurasi Penghitungan Durasi Heater

No	Input		Output (detik)		Hasil
	Suhu (°C)	Kelembapan (%)	ESP32	Python	
1	27,30	83,00	31,5	31,5	Akurat
2	27,20	85,00	31	31	Akurat
3	27,40	85,00	32	32	Akurat
4	27,50	85,00	32,5	32,5	Akurat
5	26,80	85,00	30	30	Akurat
6	27,30	86,00	31,5	31,5	Akurat
7	27,20	88,00	31	31	Akurat
8	27,00	85,00	30	30	Akurat
9	27,00	84,00	30	30	Akurat
10	26,80	80,00	30	30	Akurat
11	27,60	84,00	33	33	Akurat
12	27,40	79,00	33,2	33,2	Akurat
13	27,30	77,00	34	34	Akurat
14	26,10	75,00	30	30	Akurat
15	25,50	81,00	30	30	Akurat

Dari hasil pengujian ini, akurasinya adalah sebagai berikut

Akurasi PWM Kipas

$$= \frac{\sum \text{Data Akurat}}{\sum \text{Seluruh Data}} \times 100\% \quad (26)$$

$$\text{Akurasi PWM Kipas} = \frac{15}{15} \times 100\% = 100\%$$

Akurasi Durasi Pompa

$$= \frac{\sum \text{Data Akurat}}{\sum \text{Seluruh Data}} \times 100\% \quad (27)$$

Akurasi Durasi Pompa

$$= \frac{15}{15} \times 100\% = 100\%$$

Akurasi Durasi Heater

$$= \frac{\sum \text{Data Akurat}}{\sum \text{Seluruh Data}} \times 100\% \quad (28)$$

Akurasi Durasi Heater

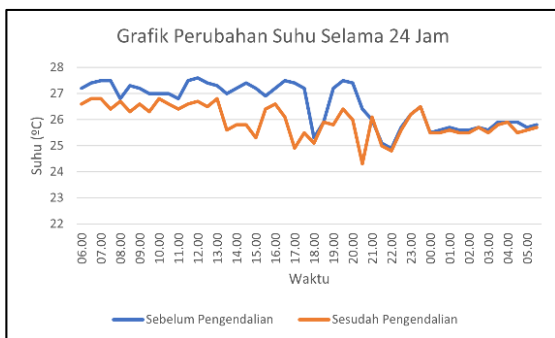
$$= \frac{15}{15} \times 100\% = 100\%$$

Dari keseluruhan pengujian akurasi ini, didapatkan temuan bahwa akurasi penghitungan logika *fuzzy* di antara yang dilakukan oleh mikrokontroler dengan program *python* sebagai referensi sangat tinggi, yaitu 100% untuk masing-masing *output*.

3.2.4 Pengujian Efektivitas Sistem

Hasil pengujian efektivitas sistem dalam menjaga kondisi ideal suhu dan kelembapan selama 24 jam penuh dapat dilihat pada Tabel 10.

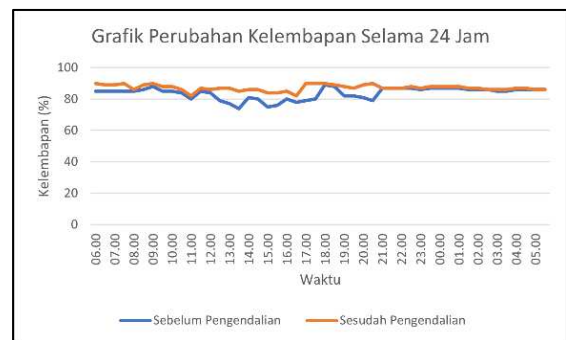
Adapun grafik perubahan suhu dan kelembapan selama 24 jam pengoperasian sistem ini dapat dilihat pada Gambar 19 dan 20.



Gambar 19. Grafik Perubahan Suhu Selama 24 Jam

Tabel 10. Hasil Pengujian Efektivitas Sistem

Waktu	Sebelum Pengendalian		Setelah Pengendalian	
	Suhu (°C)	Kelembapan (%)	Suhu (°C)	Kelembapan (%)
06.00	27,20	85,00	26,60	90,00
07.00	27,50	85,00	26,80	89,00
08.00	26,80	85,00	26,70	86,00
09.00	27,20	88,00	26,60	90,00
10.00	27,00	85,00	26,80	88,00
11.00	26,80	80,00	26,40	82,00
12.00	27,60	84,00	26,70	86,00
13.00	27,30	77,00	26,80	87,00
14.00	27,20	81,00	25,80	86,00
15.00	27,20	75,00	25,30	84,00
16.00	27,20	80,00	26,60	85,00
17.00	27,40	79,00	24,90	90,00
18.00	25,30	89,00	25,10	90,00
19.00	27,20	82,00	25,80	88,00
20.00	27,40	81,00	26,00	89,00
21.00	26,00	87,00	26,10	87,00
22.00	24,90	87,00	24,80	87,00
23.00	26,20	86,00	26,20	87,00
00.00	25,50	87,00	25,50	88,00
01.00	25,70	87,00	25,60	88,00
02.00	25,60	86,00	25,50	87,00
03.00	25,60	85,00	25,50	86,00
04.00	25,90	86,00	25,90	87,00
05.00	25,70	86,00	25,50	86,00
<b>Rata-Rata</b>	26,99	83,16	26,35	86,33
<b>Standar Deviasi</b>	0,70	3,87	0,98	2,12



Gambar 20. Grafik Perubahan Kelembapan Selama 24 Jam

Dari hasil pengujian keempat ini, didapatkan temuan bahwa sistem mampu mengawal kondisi suhu serta kelembapan dalam kumbung jamur tiram supaya tetap berada dalam kondisi yang ideal, yaitu 24 – 27 derajat celsius dan 80 – 90 persen.

3.2.5 Pengujian Penghitungan Konsumsi Listrik dan Air

Untuk melakukan penghitungan konsumsi listrik, diperlukan data konsumsi daya komponen-komponen yang digunakan. Dalam hal ini, data konsumsi daya dikumpulkan dari spesifikasi komponen yang terdapat dalam *datasheet* ataupun yang tertera dalam platform *e-commerce*. Selain itu, data lama aktifnya masing-masing komponen juga diperlukan untuk dapat menghitung nilai konsumsi listrik yang digunakan. Adapun lama aktifnya masing-masing komponen dalam satu harinya

didapatkan dari hasil pengujian efektivitas sistem. Hasil pengujian penghitungan konsumsi listrik sendiri dapat dilihat pada Tabel 11.

Tabel 11. Konsumsi Daya Sistem Dalam Sehari

No.	Komponen	Daya (Watt)	Lama Aktif (jam)	Energi (Wh)
1.	NodeMCU ESP32	1,65	24	39,6
2.	LCD I2C 20 x 4	1,25	24	30
3.	Sensor DHT22	0,00495	0,27	0,00134
4.	Modul RTC DS3231	0,00099	24	0,0238
5.	Relay 5 V (2 buah)	0,50	1,87	0,939
6.	Kipas DC 12 V (2 buah, rata-rata 155 PWM)	14,6	1,87	27,4
7.	Pompa DC 12 V	24	1,87	45,1
8.	Heater	25	0	0
<b>Total (Wh)</b>				143,06
<b>Total (kWh)</b>				0,143

Berdasarkan tabel di atas, didapatkan temuan bahwa sistem ini hanya mengonsumsi daya sebesar 0,143 kWh saja dalam satu hari. Dalam hal ini, tidak semua komponen aktif selama 24 jam penuh, seperti Sensor DHT22 yang hanya aktif selama 0,27 jam (16 menit), dan akuator (kipas, pompa, serta relay) yang hanya aktif selama 1,87 jam (1 jam 53 menit) saja. Artinya, energi listrik yang diperlukan sangatlah kecil untuk dapat mengoperasikan sistem ini.

Selain listrik, konsumsi air juga perlu dihitung. Untuk menghitung konsumsi air sendiri, data-data yang diperlukan adalah debit pompa sesuai spesifikasi yang tercantum pada platform *e-commerce* tempat pembelian pompa, serta lama aktifnya pompa dalam satu hari.

Dalam spesifikasi yang tercantum, debit pompa yang digunakan adalah 1,4 liter/menit, sedangkan lama aktifnya sendiri adalah 1 jam 53 menit (113 menit). Dari data-data tersebut, konsumsi airnya adalah sebagai berikut.

Volume

$$= \frac{\text{Debit} \left( \frac{\text{liter}}{\text{menit}} \right) \times \text{Waktu (menit)}}{1000} \quad (29)$$

$$\text{Volume} = \frac{1,4 \times 113}{1000} = 0,158$$

Berdasarkan hasil penghitungan di atas, didapatkan dalam satu hari adalah 0,158 m<sup>3</sup>. Terlihat cukup besar, tetapi, ini menjadi beralasan karena perlunya kumbung jamur tiram untuk senantiasa dalam kondisi kelembapan yang tinggi.

Hasil pengujian ini tidak menggambarkan kebutuhan harian secara pasti. Konsumsi listrik dan air bisa lebih rendah lagi jika kondisi suhu dan kelembapan kumbung telah stabil.

### 3.2.6 Pengujian Perbandingan Hasil Panen Dengan Sistem Manual

Dalam pengujian perbandingan ini, baik sistem manual maupun sistem yang dilengkapi dengan IoT dan logika *fuzzy* sama-sama menggunakan empat buah *baglog*. Adapun kumbung manual perbandingan ini sendiri dapat dilihat pada Gambar 21.



Gambar 21. Kumbung Manual Sebagai Pembanding

Untuk hasil pengujiannya sendiri dapat dilihat pada Tabel 12.

Tabel 12. Hasil Pengujian Perbandingan Hasil Panen Dengan Sistem Manual

Baglog ke-	Hasil (gram)	
	Sistem IoT dan Fuzzy	Sistem Manual
1	150	0
2	150	0
3	50	0
4	150	0
Jumlah	500	0

Hasil pengujian yang dilaksanakan menunjukkan bahwa sistem hasil pengembangan dari penelitian ini telah efektif dalam mendukung keberhasilan budidaya jamur tiram, dengan hasil sebanyak 500 gram dari seluruh *baglog* yang digunakan. Di saat yang bersamaan, *baglog* yang dibudidayakan secara manual justru mengalami kegagalan panen.

Hasil pengujian ini bisa dikatakan sejalan, jika dikomparasikan dengan penelitian lainnya oleh Dani et al. (2022), di mana pertumbuhan jamur tiram yang dibudidayakan dengan menggunakan kendali logika *fuzzy* juga jauh lebih baik dibandingkan dengan jamur tiram yang dibudidayakan secara manual. Meskipun demikian, dalam penelitian tersebut, kendali pada sistem pembanding masih dijalankan secara otomatis tetapi tanpa menggunakan logika *fuzzy*. Selain itu, frekuensi penyiraman juga lebih banyak (empat kali), jika dikomparasikan dengan penelitian ini yang hanya tiga kali dan dilakukan secara manual oleh manusia. Hal tersebut memperlihatkan adanya jurang perbedaan yang besar di antara hasil pembudidayaan secara manual, pembudidayaan secara otomatis, dan pembudidayaan secara otomatis dengan logika *fuzzy*.

### 3.3 Analisis Keekonomian Sistem

Dalam membuat sebuah sistem, termasuk sistem ini, ada biaya investasi yang harus dikeluarkan. Adapun biaya yang dikeluarkan mencakup biaya awal serta biaya operasional. Rincian biaya investasi yang mesti dikeluarkan sendiri dapat dilihat pada Tabel 13.

Tabel 13. Biaya Investasi Sistem Berbasis IoT dan Logika Fuzzy

No.	Komponen	Biaya
<b>Biaya Awal</b>		
1.	Mikrokontroler, Sensor, dan Aktuator	Rp398.750,00
2.	Kabel, Breadboard, Stop Kontak, Power Supply, dan Adaptor	Rp242.250,00
3.	Triplek, Aluminium foil, Skotlet, dan Baut	Rp271.160,00
4.	Perakitan Kumbung	Rp100.000,00
<b>Total Biaya Awal</b>		<b>Rp1.012.160,00</b>
<b>Biaya Operasional</b>		
1.	Pembelian Baglog (4 buah)	Rp32.000,00
2.	Biaya Air (0,158 m <sup>3</sup> x 30 hari = 4,74 m <sup>3</sup> , 1 m <sup>3</sup> = Rp4.900,00)	Rp23.226,00
3.	Biaya Listrik (0,143 kWh x 30 hari = 4,29 kWh, 1 kWh = Rp1.444,70)	Rp6.197,76
4.	Biaya Internet	Rp50.000,00
<b>Total Biaya Operasional</b>		<b>Rp111.423,76</b>
<b>TOTAL</b>		<b>Rp1.123.683,76</b>

Biaya investasi di atas tetap memiliki potensi fluktuatif. Biaya air dan listrik bisa lebih rendah lagi jika kondisi suhu dan kelembapan kumbung sudah stabil. Selain itu biaya pembelian baglog bukan merupakan biaya yang harus dikeluarkan setiap bulan, karena baglog secara umum dapat digunakan hingga tiga bulan. Hal ini menandakan bahwa sistem mendukung keberlanjutan dengan adanya efisiensi biaya operasional untuk jangka panjang.

Adapun untuk biaya investasi yang dikeluarkan untuk sebuah sistem budidaya berbasis manual, rinciannya dapat dilihat melalui Tabel 14.

Tabel 14. Biaya Investasi Sistem Manual

No.	Komponen	Biaya
<b>Biaya Awal</b>		
1.	Triplek, Aluminium foil, Skotlet, dan Baut	Rp271.160,00
2.	Penyiram Tanaman 1 liter	Rp23.940,00
3.	Perakitan Kumbung	Rp100.000,00
<b>Total Biaya Awal</b>		<b>Rp395.100,00</b>
<b>Biaya Operasional</b>		
1.	Pembelian Baglog (4 buah)	Rp32.000,00
2.	Biaya Air (0,001 m <sup>3</sup> x 30 hari = 0,03 m <sup>3</sup> , 1 m <sup>3</sup> = Rp4.900,00)	Rp441,00
3.	Biaya Tenaga Kerja (1 orang)	Rp1.200.000,00
<b>Total Biaya Operasional</b>		<b>Rp1.232.441,00</b>
<b>TOTAL</b>		<b>Rp1.627.541,00</b>

Dari perbandingan biaya yang dikeluarkan dan hasil panen di antara kedua sistem, menunjukkan bahwa sistem yang dikembangkan dalam penelitian ini lebih efektif secara teknis. Namun, dengan kapasitas yang ada sekarang, keuntungan finansial tidak akan diperoleh dalam waktu singkat. Meskipun demikian, adanya peningkatan hasil produksi jamur tiram menunjukkan bahwa potensi skalabilitas

sistem ini masih memungkinkan. Peningkatan perangkat terhubung, kapasitas baglog, serta kapasitas basis penyimpanan data yang digunakan akan meningkatkan potensi tercapainya keuntungan finansial yang lebih cepat. Walaupun begitu, jika peningkatan kapasitas ini dilakukan maka akan berdampak pada kenaikan biaya investasi yang harus diperlukan.

## 4. KESIMPULAN DAN SARAN

Dari keseluruhan proses penelitian, dapat ditarik kesimpulan bahwa sistem berhasil memenuhi kebutuhan yang ada untuk membuat proses budidaya menjadi lebih optimal, seperti memantau kondisi lingkungan (suhu dan kelembapan) serta menyampaikannya kepada pengguna. Selain itu, sistem juga dapat melakukan pengendalian otomatis dan akurat terhadap kondisi lingkungan yang ada. Sistem mampu beroperasi untuk menjaga kondisi kumbung selama 24 jam penuh untuk menjaga kondisi kumbung tetap ideal. Penggunaan sistem ini juga berdampak terhadap hasil panen jamur tiram yang lebih maksimal.

Penelitian ini masih memiliki keterbatasan. Maka diperlukan sebuah pengembangan lebih lanjut. Saran yang dapat diberikan untuk pengembangan selanjutnya adalah perlunya meningkatkan kapasitas sistem sehingga jumlah baglog yang digunakan dapat lebih banyak lagi. Dengan pengembangan tersebut, diharapkan sistem ini dapat lebih efektif terhadap optimalisasi budidaya jamur tiram sehingga berdampak terhadap keuntungan yang lebih signifikan dalam periode waktu yang lebih singkat.

## DAFTAR PUSTAKA

- AWI, F.B., RABI, A. dan DIRGANTARA, W., 2022. Pengimplementasian Metode Fuzzy Logic pada Kontrol Rumah Jamur Otomatis Berbasis Node-RED. Jurnal Teknik Elektro dan Komputer TRIAC, 9(3), pp.130–134. <https://doi.org/10.21107/triac.v9i3.17278>.
- DANI, A.W., SIAHAAN, D.Y., YULIZA, Y., SIRAIT, F. dan SUPEGINA, F., 2022. Rancang Bangun Smart Garden Untuk Budidaya Jamur Tiram Dengan Metode Sistem Fuzzy Mamdani Berbasis Internet Of Things (IoT). Jurnal Teknologi Elektro, 13(2), p.108. <https://doi.org/10.22441/jte.2022.v13i2.008>.
- GIGNAC, G.E. dan SZODORAI, E.T., 2024. Defining intelligence: Bridging the gap between human and artificial perspectives. Intelligence, 104, p.101832. <https://doi.org/10.1016/j.intell.2024.101832>.
- HAERUDDIN, H., CANDRA, B., LEE, D., ADIYASA, F., HADI, H., SEPBIANTO, S. dan RICHTAN, J., 2022. Rancangan Sistem Keamanan Rumah Berbasis IoT dengan Cisco Packet Tracer. Telcomatics, 7(1), p.30.

- <https://doi.org/10.37253/telcomatics.v7i1.6767>
- HERIYANI, N. dan ERNAWATI, S., 2024. Pemanfaatan Teknologi IoT Berbasis Mobile Dalam Upaya Monitoring Kualitas Air Pada Tanaman Hidroponik. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(3), pp.2807–2813. <https://doi.org/10.36040/jati.v8i3.9273>.
- IRWANTO, F., HASAN, U., LAYS, E.S., DE LA CROIX, N.J., MUKANYILIGIRA, D., SIBOMANA, L. dan AHMAD, T., 2024. IoT and fuzzy logic integration for improved substrate environment management in mushroom cultivation. *Smart Agricultural Technology*, 7, p.100427. <https://doi.org/10.1016/j.atech.2024.100427>.
- KUSWANTO, J., MAULANA, B., VERNANDO, R. dan BERTA, S., 2023. Penerapan Metode Fuzzy Tsukamoto dalam Penilaian Kinerja Karyawan di Perusahaan Air Minum Kabupaten OKU. *Bulletin of Computer Science Research*, 4(1), pp.84–90. <https://doi.org/10.47065/bulletincsr.v4i1.312>.
- LASUT, R.C.I., OSLAN, Y. dan WIBOWO, A., 2024. Program Bantu Perkiraan Hasil Panen Tanaman Jagung Berbasis Web Studi Kasus: Kelompok Tani Mutiara Desa Maunggal Karya. *Jurnal Teknologi Dan Sistem Informasi Bisnis*, 6(4), pp.609–620. <https://doi.org/10.47233/jteksis.v6i4.1253>.
- MUHAMMAD, D., AHVAR, E., AHVAR, S., TROCAN, M., MONTPETIT, M.-J. dan EHSANI, R., 2024. Artificial Intelligence of Things (AIoT) for smart agriculture: A review of architectures, technologies and solutions. *Journal of Network and Computer Applications*, 228, p.103905. <https://doi.org/10.1016/j.jnca.2024.103905>.
- PUJIANTO, S. dan FADLILAH, U., 2023. Pemanfaatan ESP8266 Dan Aplikasi Blynk Untuk Monitoring Suhu Dan Kelembapan Kumbung Jamur. [Undergraduate Thesis] Universitas Muhammadiyah Surakarta. Available at: <<https://eprints.ums.ac.id/114353/>> [Accessed 1 January 2025].
- SETIYAWAN, D., ARBANSYAH, A. dan LATIPAH, A.J., 2023. Fuzzy Inference System Metode Tsukamoto Untuk Penentuan Program Studi Fakultas Sains Dan Teknologi Di Universitas Muhammadiyah Kalimantan Timur. *JIKO (Jurnal Informatika dan Komputer)*, 7(1), p.23. <https://doi.org/10.26798/jiko.v7i1.657>.
- SHAHARA, M.A., GRASBERGER, J., ABDELKHALIK, O. dan TOM, N., 2024. Fuzzy Logic Control of Wave Energy Converters. *IFAC-PapersOnLine*, 58(20), pp.404–409. <https://doi.org/10.1016/j.ifacol.2024.10.087>.
- SUNARDI, YUDHANA, A. dan FURIZAL, 2023. Tsukamoto Fuzzy Inference System on Internet of Things-Based for Room Temperature and Humidity Control. *IEEE Access*, 11, pp.6209–6227. <https://doi.org/10.1109/ACCESS.2023.3236183>.
- TRIONO, E., 2020. Budidaya Jamur Tiram dan Pengolahannya Sebagai Upaya Meningkatkan Ekonomi Kreatif Desa Kaulon. *Jurnal Karinov*, 3(2), pp.64–68.
- UBAIDILLAH, F.I., ISTIADI, I. dan MUKHSIM, M., 2020. Sistem Pemantauan Dan Pengendalian Rumah Jamur Dengan Metode Fuzzy Secara Wireless. *Simetris: Jurnal Teknik Mesin, Elektro dan Ilmu Komputer*, 11(1), pp.223–232. <https://doi.org/10.24176/simet.v11i1.3975>.
- WU, M. dan CHEN, X., 2024. Application of Internet of Things and embedded technology in electronic communication. *Measurement: Sensors*, 34, p.101246. <https://doi.org/10.1016/j.measen.2024.101246>.

*Halaman ini sengaja dikosongkan*