

Rancang Bangun Sistem Jemuran Otomatis dan Monitoring Cuaca berbasis Logika Fuzzy

DOI: <http://dx.doi.org/10.35889/jutisi.v15i1.3434>

Creative Commons License 4.0 (CC BY – NC)

**Alexander Ardian Pulung Sasongko**

Teknik Informatika, Universitas Stikubank, Semarang, Indonesia

*e-mail *Corresponding Author*: alexanderardian17@gmail.com

Abstract

Weather uncertainty in tropical regions often hinders the drying process and increases the risk of clothes getting wet again due to rain. This study aims to design a smart clothesline system based on the Internet of Things (IoT) with monitoring and adaptive automation capabilities. The methodology applied is prototyping using an ESP32 microcontroller integrated with BME280, BH1750, YL-83, and clothing dryness sensors, as well as utilizing Fuzzy Logic for decision making on the position of the drying rack poles. The main findings show that the environmental sensors have high accuracy with an average error of less than 1%. The Fuzzy algorithm proved effective in executing safety priorities by retracting the drying rack during rain, as well as improving efficiency by automatically retracting the drying rack when clothes are detected as dry by the sensor. It is concluded that this system successfully provides a reliable and efficient autonomous solution for drying management amid dynamic weather conditions.

Keywords: Automatic Clothesline; ESP32; Fuzzy Logic; Internet of Things; Weather Monitoring.

Abstrak

Ketidakpastian cuaca di wilayah tropis sering menghambat proses pengeringan dan meningkatkan risiko pakaian basah kembali akibat hujan. Penelitian ini bertujuan merancang sistem jemuran cerdas berbasis *Internet of Things* (IoT) dengan kemampuan monitoring dan otomasi adaptif. Metodologi yang diterapkan adalah prototyping menggunakan mikrokontroler ESP32 yang terintegrasi dengan sensor BME280, BH1750, YL-83, dan sensor kekeringan pakaian, serta memanfaatkan Logika Fuzzy untuk pengambilan keputusan posisi tiang jemuran. Temuan utama menunjukkan bahwa sensor lingkungan memiliki akurasi tinggi dengan rata-rata error di bawah 1%. Algoritma Fuzzy terbukti efektif menjalankan prioritas keamanan dengan menarik tiang jemuran saat hujan, serta meningkatkan efisiensi dengan menarik jemuran secara otomatis saat pakaian terdeteksi kering oleh sensor. Disimpulkan bahwa sistem ini berhasil memberikan solusi otonom yang andal dan efisien untuk manajemen penjemuran di tengah kondisi cuaca dinamis.

Kata kunci: ESP32; Internet of Things; Jemuran Otomatis; Logika Fuzzy; Monitoring Cuaca

1. Pendahuluan

Pemanfaatan teknologi *Internet of Things* (IoT) telah menjadi pendorong utama dalam evolusi perangkat rumah tangga modern menuju konsep *smart home*. Integrasi teknologi ini menawarkan potensi besar untuk meningkatkan efisiensi aktivitas domestik dengan mengubah perangkat konvensional menjadi sistem cerdas yang mampu berinteraksi dengan lingkungannya[1], [2]. Dalam konteks kehidupan masyarakat modern, efisiensi waktu dan otomatisasi menjadi kebutuhan krusial, terutama untuk menangani pekerjaan rumah tangga yang bersifat repetitif. Salah satu bagian yang mendapat perhatian serius adalah manajemen perawatan pakaian. Penerapan IoT dalam bagian ini tidak hanya menawarkan kenyamanan, tetapi juga solusi atas keterbatasan pemantauan manusia yang tidak dapat dilakukan terus-menerus selama 24 jam. Oleh karena itu, pengembangan sistem otomasi pada perangkat rumah tangga menjadi tema penelitian yang urgen untuk menjawab tantangan efisiensi energi dan produktivitas manusia di era digital saat ini.

Namun, di wilayah beriklim tropis seperti Indonesia, aktivitas menjemur pakaian menghadapi tantangan spesifik berupa ketidakpastian cuaca yang tinggi. Perubahan kondisi atmosfer yang cepat antara panas terik dan hujan tiba-tiba seringkali menghambat proses pengeringan dan meningkatkan risiko pakaian kembali basah atau berbau apek[3]. Meskipun berbagai penelitian sebelumnya telah mengembangkan jemuran otomatis, mayoritas solusi yang ada masih memiliki keterbatasan fundamental. Penelitian-penelitian terdahulu umumnya hanya bersifat reaktif dengan mengandalkan sensor hujan dan cahaya menggunakan logika biner sederhana (buka/tutup)[4], [5]. Pendekatan ini dinilai kurang efektif karena tidak mempertimbangkan kondisi "abu-abu" seperti cuaca mendung namun tidak hujan, serta mengabaikan variabel terpenting yaitu kondisi kekeringan pakaian itu sendiri. Akibatnya, sistem seringkali bekerja kurang efisien atau menarik jemuran terlalu cepat padahal pakaian belum kering, atau sebaliknya, membiarkan jemuran terpapar debu saat sudah kering.

Untuk mengatasi permasalahan tersebut, penelitian ini menawarkan solusi berupa sistem jemuran cerdas yang mengintegrasikan mikrokontroler ESP32 dengan metode kendali Logika *Fuzzy*. Konsep solusi ini dipilih dengan rasionalisasi bahwa Logika *Fuzzy* memiliki kemampuan meniru cara berpikir manusia dalam menangani ketidakpastian data, sehingga mampu menerjemahkan kondisi cuaca yang ambigu (seperti mendung tipis atau gerimis ringan) menjadi keputusan mekanis yang presisi, tidak sekadar *on* atau *off*[6], [7].

Berdasarkan paparan konsep solusi di atas, penelitian ini bertujuan untuk merancang dan membangun prototipe sistem jemuran cerdas berbasis IoT yang mampu melakukan *monitoring* kondisi cuaca secara *real-time* serta mengendalikan tiang jemuran secara otonom menggunakan algoritma *Fuzzy*. Manfaat yang diharapkan dari penelitian ini adalah tersedianya sistem yang dapat meningkatkan efisiensi proses pengeringan pakaian, memberikan perlindungan otomatis terhadap perubahan cuaca, serta menyediakan fitur pemantauan dan notifikasi jarak jauh melalui platform seluler. Dengan demikian, sistem ini diharapkan dapat memberikan solusi praktis bagi permasalahan rumah tangga serta berkontribusi pada pengembangan literatur mengenai penerapan kecerdasan buatan pada perangkat IoT berbiaya rendah.

2. Tinjauan Pustaka

Berbagai penelitian telah dilakukan untuk mengembangkan sistem jemuran otomatis guna mengatasi permasalahan cuaca tak menentu. Fokus utama dari penelitian-penelitian terdahulu umumnya terletak pada aspek konektivitas dan responsivitas terhadap hujan. Pengembangan alat jemuran otomatis berbasis NodeMCU ESP8266 yang memanfaatkan sensor hujan (*rain sensor*) sebagai pemicu utama dengan model yang berfokus pada mekanisme perlindungan sederhana di mana motor DC yang dikaitkan dengan tali akan menarik jemuran saat mendeteksi air, dengan fitur pemantauan tambahan melalui aplikasi Blynk[5].

Pendekatan serupa dilakukan dengan merancang sistem jemuran otomatis menggunakan ESP32 yang berfokus pada integrasi notifikasi *real-time* melalui Telegram untuk memberikan peringatan dini kepada pengguna, namun logika pengendalian yang digunakan masih bersifat reaktif dengan ambang batas biner[3].

Upaya untuk meningkatkan kecerdasan sistem juga dilakukan dengan menerapkan metode Logika Fuzzy untuk mengolah data suhu (DHT22), cahaya (LDR), dan hujan guna mengontrol tiang secara lebih halus[8]. Sementara itu, pendekatan berbeda ditawarkan dengan mengintegrasikan API cuaca (*Open Weather*) ke dalam sistem berbasis ESP32 dengan konsep peringatan dini (*early warning*) berdasarkan prakiraan cuaca digital, sehingga sistem dapat mengantisipasi hujan sebelum benar-benar terjadi, meskipun memiliki ketergantungan tinggi pada koneksi internet[4].

Adapula yang menambahkan fitur notifikasi suara dan email, namun belum menyentuh aspek efisiensi pengeringan berdasarkan kondisi pakaian itu sendiri[9]. Pengembangan sistem berbasis Arduino Uno yang menambahkan sensor kelembapan dan kipas pengering juga telah dilakukan, namun masih menggunakan mikrokontroler dengan performa terbatas dan logika *threshold* sederhana[10].

Meskipun berbagai prototipe telah dikembangkan, terdapat kelemahan mendasar pada mayoritas penelitian terdahulu yang membatasi efektivitas sistem dalam skenario nyata. Sebagian besar sistem yang ada hanya berfokus pada perlindungan reaktif terhadap hujan dengan menggunakan sensor tunggal atau kombinasi sensor lingkungan yang terbatas, tanpa memperhitungkan kondisi objek yang dijemur[6], [7], [10], [11], [12], [13], [14].

Kebaruan dari penelitian ini terletak pada integrasi parameter kondisi internal objek (pakaian) ke dalam algoritma pengambilan keputusan, yang belum diakomodasi secara komprehensif oleh penelitian-penelitian terdahulu. Jika mayoritas riset sebelumnya hanya berfokus pada perlindungan reaktif terhadap faktor eksternal (hujan) menggunakan logika biner, dan menggunakan sensor analog standar, penelitian ini mengajukan konsep sistem yang proaktif dengan kebaruan (*novelty*) pada tiga aspek utama. Pertama, penambahan *Water Sensor* khusus untuk mendeteksi tingkat kekeringan pakaian secara *real-time*, memungkinkan sistem untuk menghentikan proses penjemuran secara otonom demi efisiensi energi dan menjaga kualitas kain. Kedua, penggunaan sensor lingkungan presisi tinggi (BME280 dan BH1750 digital) untuk menggantikan sensor analog konvensional. Ketiga, implementasi Logika *Fuzzy* yang mengolah korelasi antara kondisi cuaca eksternal dan tingkat kekeringan internal pakaian untuk menghasilkan keputusan kendali yang presisi. Model yang diusulkan ini menawarkan solusi manajemen penjemuran yang lebih terintegrasi, efisien, dan cerdas dibandingkan model-model sebelumnya.

3. Metodologi

Penelitian ini menggunakan metode pengembangan prototyping yang terdiri dari tahapan identifikasi kebutuhan, perancangan sistem, implementasi, dan pengujian. Pendekatan ini memungkinkan evaluasi berkelanjutan pada setiap iterasi untuk memastikan sistem bekerja secara adaptif terhadap kondisi lingkungan yang dinamis.

1) Penelitian Awal

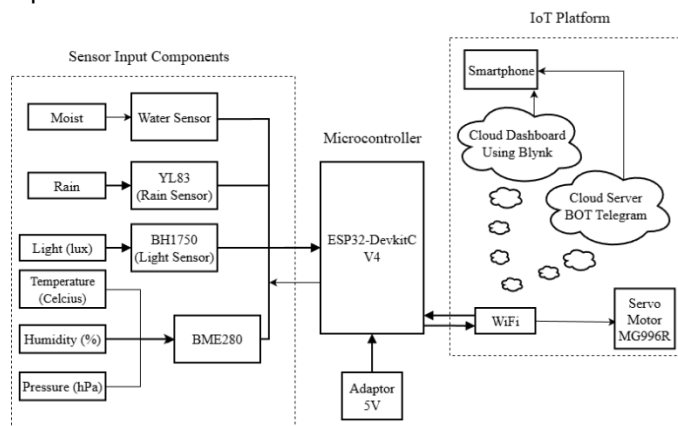
Tahap awal penelitian difokuskan pada identifikasi kebutuhan sistem untuk mengatasi permasalahan ketidakpastian cuaca. Sistem dirancang untuk tidak hanya memantau parameter lingkungan, tetapi juga melakukan aksi otomatis berdasarkan kondisi tersebut. Kebutuhan fungsional utama mencakup kemampuan akuisisi data multi-sensor (suhu, kelembapan, hujan, cahaya, dan kekeringan pakaian), pengolahan data menggunakan algoritma cerdas, serta penyediaan antarmuka pemantauan jarak jauh. Berikut adalah hasil identifikasi kebutuhan fungsional sistem:

Tabel 1. Tabel Kebutuhan Fungsional Sistem

Kategori Pengujian	Skenario Pengujian
Stabilitas Koneksi	Sistem mampu mempertahankan koneksi WiFi/Internet agar proses monitoring tetap stabil
Akuisisi Data Lingkungan	Sistem mampu membaca parameter cuaca (suhu, kelembapan, tekanan, cahaya, dan hujan) secara <i>real-time</i> dari rangkaian sensor.
Deteksi Kondisi Pakaian	Sistem mampu mengidentifikasi tingkat kekeringan pakaian (Basah/Lembap/Kering) menggunakan <i>water sensor</i> .
Logika Kendali Cerdas	Sistem mampu mengolah data multi-sensor menggunakan algoritma <i>Fuzzy</i> untuk menghasilkan keputusan gerak yang adaptif.
Otomasi Mekanis	Sistem mampu menggerakkan aktuator motor servo ke posisi aman (0°), siaga (90°), atau jemur (180°) sesuai keputusan logika.
Pemantauan Jarak Jauh	Sistem menyediakan antarmuka visual (<i>dashboard</i>) yang dapat diakses melalui <i>smartphone</i> (aplikasi Blynk) untuk memantau status sistem.
Notifikasi Peringatan	Sistem mengirimkan pesan peringatan dini (<i>early warning</i>) via Telegram saat mendeteksi hujan atau saat pakaian telah kering.
Kendali Manual (Override)	Sistem memfasilitasi pengguna untuk mengambil alih kendali tiang secara manual jika diperlukan, mengabaikan logika otomatis.

2) Perancangan Sistem

Perancangan sistem difokuskan pada integrasi komponen *Input-Process-Output* (IPO) yang digambarkan dalam diagram blok sistem. Seperti terlihat pada Gambar 1, sistem menggunakan mikrokontroler ESP32 sebagai unit pemrosesan pusat yang mengelola aliran data dari sisi input menuju output.



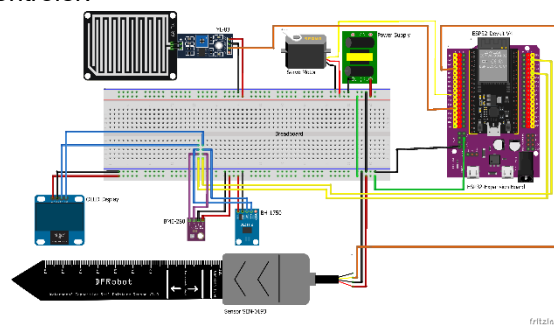
Gambar 1. Block Diagram

Pada sisi input, sistem mengakuisisi data lingkungan melalui sensor BME280 (suhu, kelembapan, tekanan) dan BH1750 (cahaya) menggunakan protokol komunikasi digital, serta sensor YL-83 (hujan) dan Water Sensor (kelembapan pakaian) melalui jalur analog. Data ini diproses oleh ESP32 menggunakan algoritma Fuzzy untuk menghasilkan keputusan kontrol.

Pada sisi output, keputusan sistem dieksekusi oleh motor servo MG996R yang berfungsi sebagai aktuator penggerak tiang jemuran. Selain aksi mekanis, sistem juga mendistribusikan data telemetri ke antarmuka pengguna, yaitu layar OLED untuk pemantauan lokal, serta platform Blynk IoT dan Bot Telegram untuk pemantauan jarak jauh dan notifikasi peringatan dini.

3) Desain Rangkaian Elektrik

Implementasi perangkat keras mengacu pada desain skema pengkabelan (wiring diagram) yang ditunjukkan pada Gambar 2. Desain ini menerapkan strategi pemisahan jalur daya dan efisiensi pin mikrokontroler.



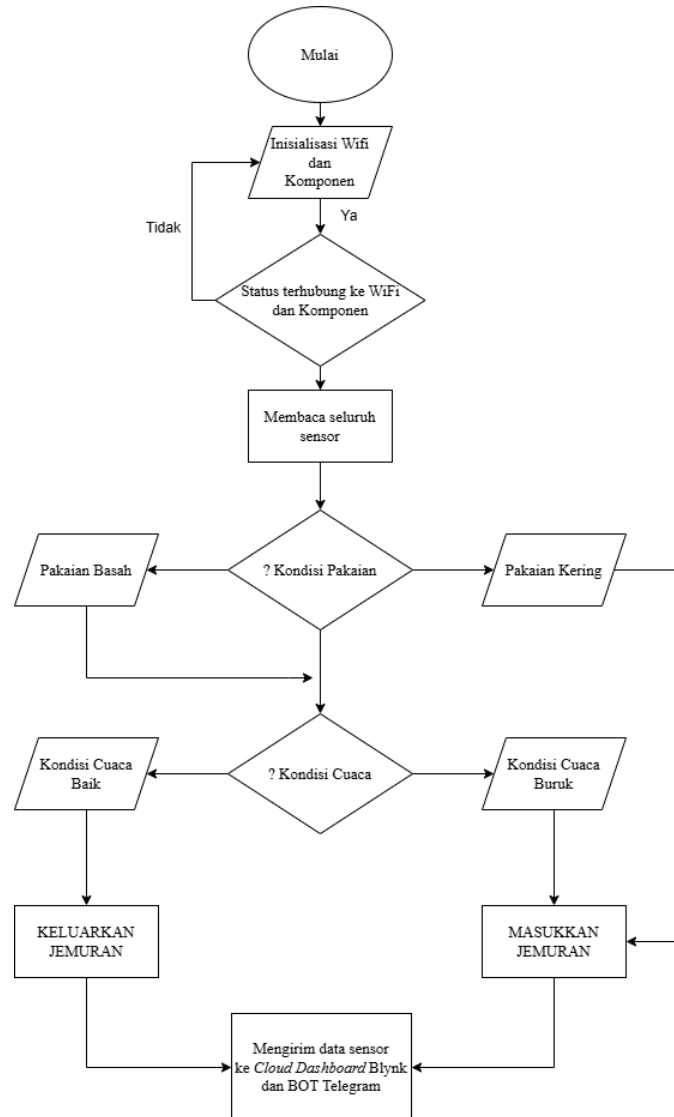
Gambar 2. Wiring Diagram

Untuk menjaga stabilitas tegangan, sistem menggunakan *power supply* ganda. Jalur 3.3V dari regulator ESP32 digunakan untuk menyuplai sensor berdaya rendah (BME280, BH1750, OLED), sedangkan jalur eksternal 5V digunakan khusus untuk menyuplai motor servo MG996R. Pemisahan ini mencegah terjadinya *brownout* atau *reset* pada mikrokontroler akibat lonjakan arus saat servo bergerak.

Komunikasi data dioptimalkan menggunakan jalur I2C (*Inter-Integrated Circuit*) secara paralel pada pin GPIO 21 (SDA) dan GPIO 22 (SCL) untuk sensor BME280, BH1750, dan OLED. Sementara itu, sensor analog dihubungkan ke pin ADC (*Analog-to-Digital Converter*) GPIO 34 dan 35 yang memiliki resolusi pembacaan 12-bit, memberikan presisi tinggi dalam mendeteksi perubahan kelembapan kain dan intensitas hujan.

4) Pengembangan Kode Program

Pengembangan perangkat lunak (firmware) merupakan implementasi logika kontrol yang ditanamkan ke dalam mikrokontroler ESP32. Program dibangun menggunakan bahasa C++ pada lingkungan Arduino IDE dengan struktur non-blocking menggunakan fungsi pewaktu internal (millis) untuk memastikan sistem tetap responsif terhadap interupsi eksternal, seperti perintah dari Bot Telegram, meskipun sedang menjalankan siklus pembacaan sensor. Alur logika program dirancang dalam siklus tertutup (*closed-loop*) yang diilustrasikan pada Gambar 3.



Gambar 3. Flowchart Diagram

Berdasarkan diagram alir di atas, mekanisme kerja kode program dapat diuraikan ke dalam beberapa tahapan sekuensial sebagai berikut:

1) Tahap Inisialisasi:

Saat perangkat pertama kali dinyalakan, program menjalankan prosedur `setup()` untuk menginisialisasi modul komunikasi (WiFi dan I2C) serta melakukan konfigurasi pin untuk sensor dan aktuator. Sistem akan melakukan upaya koneksi ke jaringan internet dan server Blynk. Jika koneksi gagal, program akan masuk ke sub-rutin reconnect hingga status terhubung tercapai, memastikan sistem tidak berjalan tanpa konektivitas.

2) Siklus Akuisisi Data:

Setelah inisialisasi berhasil, program memasuki siklus utama (*loop*). Langkah pertama dalam siklus ini adalah membaca data mentah dari seluruh sensor secara simultan: suhu/kelembapan (BME280), intensitas cahaya (BH1750), curah hujan (YL-83), dan kelembapan pakaian (*Water Sensor*). Data analog dari sensor hujan dan pakaian dikonversi menjadi nilai persentase atau kategori linguistik melalui proses fuzzifikasi.

3) Logika Pengambilan Keputusan (*Decision Making*):

Tahap ini merupakan implementasi dari blok keputusan (belah ketupat) pada flowchart, yang dieksekusi menggunakan algoritma Logika Fuzzy. Program memprioritaskan evaluasi kondisi dengan urutan hierarki:

- a. Evaluasi Kondisi Pakaian: Program pertama kali memeriksa nilai keanggotaan fuzzy dari sensor pakaian. Jika terdeteksi kondisi "Kering", program akan mengabaikan kondisi cuaca dan langsung menetapkan target aksi untuk menarik jemuran masuk. Ini adalah fitur efisiensi energi untuk menghentikan proses jika tujuan pengeringan sudah tercapai.
- b. Evaluasi Kondisi Cuaca: Jika pakaian masih terdeteksi basah atau lembap, program melanjutkan evaluasi ke parameter cuaca. Logika kontrol akan memeriksa variabel hujan dan cahaya. Jika terdeteksi hujan atau kondisi cahaya gelap, program menetapkan aksi "Masukan Jemuran". Sebaliknya, jika tidak hujan dan cahaya cukup (Cerah/Berawan), program menetapkan aksi "Keluarkan Jemuran" ke posisi sudut servo yang sesuai (90° atau 180°).

4) Eksekusi dan Telemetry:

Hasil keputusan (sudut target) dikirimkan ke motor servo untuk menggerakkan mekanisme tiang secara fisik. Secara paralel, program mengemas seluruh data sensor dan status sistem saat ini untuk dikirimkan (push) ke *Cloud Dashboard Blynk* sebagai visualisasi grafik, serta mengirimkan notifikasi ke Bot Telegram jika terjadi perubahan status kritis (misalnya: "Hujan Terdeteksi" atau "Jemuran Kering"). Siklus ini kemudian berulang kembali untuk memantau perubahan kondisi secara real-time.

5) Desain Sistem

Perancangan sistem mengadopsi arsitektur *Internet of Things* (IoT) dengan mikrokontroler ESP32 sebagai unit pemrosesan pusat. Desain sistem dibagi menjadi blok *Input*, *Process*, dan *Output*.

- a. **Blok Input:** Terdiri dari rangkaian sensor yang terhubung melalui antarmuka komunikasi spesifik. Sensor BME280 (suhu, kelembapan, tekanan) dan BH1750 (cahaya) terhubung melalui jalur I2C untuk presisi data digital. Sensor YL-83 (hujan) dan *Water Sensor* (kelembapan kain) terhubung melalui pin ADC (*Analog-to-Digital Converter*) untuk mendeteksi perubahan nilai resistansi dan kapasitansi secara kontinu.
- b. **Blok Process:** ESP32 menjalankan algoritma Logika Fuzzy. Metode ini dirancang untuk memetakan input yang bersifat ambigu (seperti "agak mendung" atau "gerimis ringan") menjadi keputusan tegas. Logika sistem memprioritaskan faktor keamanan (hujan) di atas efisiensi (panas matahari).
- c. **Blok Output:** Terdiri dari Motor Servo MG996R sebagai aktuator penggerak tiang jemuran, layar OLED untuk visualisasi lokal, serta modul Wi-Fi untuk transmisi data ke platform *Cloud* (Blynk dan Telegram).

6) Objek Penelitian dan Teknik Pengujian

Untuk mengukur keberhasilan sistem, ditetapkan variabel pengamatan yang dikategorikan sebagai variabel input dan output.

- a. **Variabel Input (Bebas):** Meliputi intensitas cahaya (Lux), persentase deteksi air hujan (%), nilai analog kelembapan kain, serta parameter suhu ($^\circ\text{C}$).
- b. **Variabel Output (Terikat):** Meliputi sudut pergerakan motor servo (0° untuk posisi aman, 90° untuk posisi siaga, 180° untuk posisi menjemur) dan latensi notifikasi pada aplikasi.

Teknik pengujian sistem dilakukan menggunakan metode *Blackbox Testing* dan Uji Validasi Sensor. *Blackbox testing* digunakan untuk memverifikasi kesesuaian antara fungsi logika

otomasi dengan skenario cuaca yang disimulasikan (misalnya: simulasi hujan buatan atau penggelapan sensor cahaya). Sedangkan uji validasi sensor dilakukan dengan membandingkan nilai pembacaan sensor sistem terhadap alat ukur standar terkalibrasi (*Thermometer*, *Hygrometer*, dan *Lux Meter* digital) untuk menghitung persentase *error* dan akurasi data.

4. Hasil dan Pembahasan

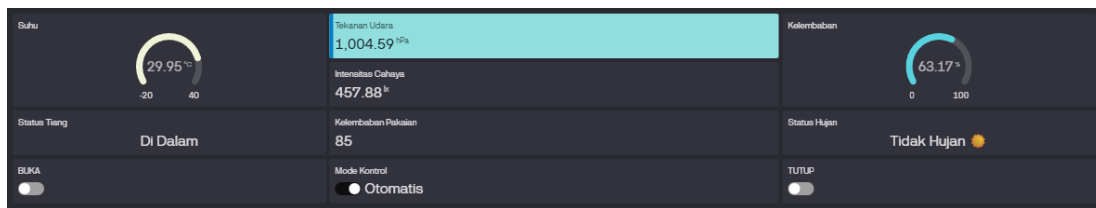
Pemaparan realisasi sistem yang telah dibangun serta analisis kinerjanya. Pembahasan meliputi perancangan arsitektur, implementasi rangkaian elektrik, pengembangan logika program, serta validasi fungsionalitas sistem melalui pengujian *blackbox* dan evaluasi kinerja keseluruhan. Dalam kasus tertentu, pembahasan juga mesti mengaitkan sejauh mana relevansi antara hasil temuan penelitian/kajian saat ini terhadap hasil-hasil temuan penelitian terdahulu yang relevan, apakah saling menguatkan, saling mempertentangkan, atau merupakan temuan baru.

1) Implementasi Sistem

Sisi perangkat lunak difokuskan pada pengembangan antarmuka pemantauan dan pengendalian jarak jauh berbasis IoT. Sistem ini menggunakan dua platform utama, yaitu Blynk IoT sebagai *dashboard* visual dan Telegram Bot sebagai sistem notifikasi berbasis kejadian (*event-based*).

a. Dashboard Blynk

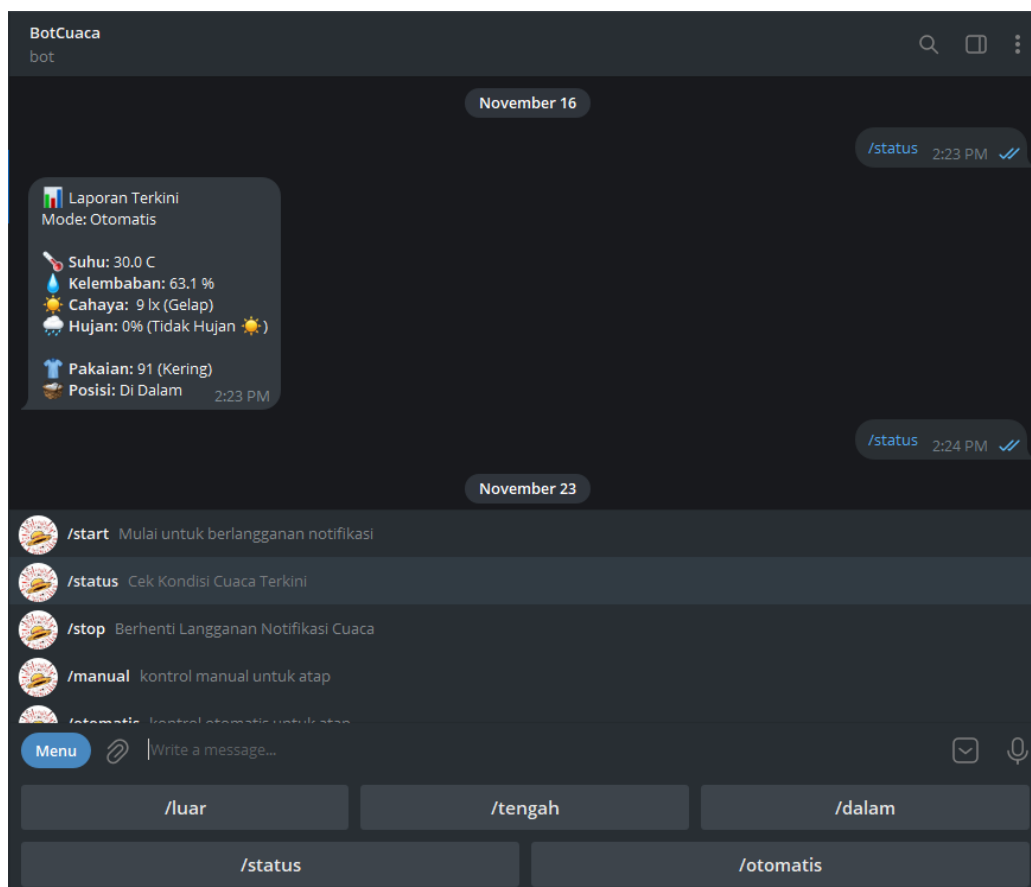
Implementasi pada platform Blynk menyajikan visualisasi data sensor secara real-time dalam format grafis yang intuitif. Widget Gauge digunakan untuk menampilkan suhu dan kelembapan udara, sementara Label Value menampilkan intensitas cahaya (Lux), status hujan, status tiang dan kelembapan pakaian. Selain fungsi monitoring, tersedia juga tombol Switch virtual untuk mode "Manual/Otomatis" dan kontrol posisi tiang, yang memungkinkan pengguna mengambil alih kendali sistem kapan saja.



Gambar 4. Dashboard Blynk via Web

b. Bot Telegram

Bot Telegram dikonfigurasi sebagai kanal komunikasi dua arah. Fungsi utamanya adalah mengirimkan pesan peringatan dini (*early warning*) secara otomatis kepada pengguna saat terjadi kondisi kritis, seperti "Hujan Turun". Selain itu, pengguna dapat mengirimkan perintah teks seperti `/status` untuk meminta laporan kondisi terkini, atau `/buka` dan `/tutup` untuk mengendalikan tiang secara langsung melalui aplikasi chatting.



Gambar 5. Tampilan Bot Telegram

2) Pengujian Sistem

Validasi fungsionalitas sistem dilakukan menggunakan metode *Blackbox Testing*. Pengujian ini bertujuan memverifikasi apakah sistem mampu menghasilkan keluaran (output) yang sesuai dengan skenario masukan (input) yang diberikan, tanpa melihat struktur kode internal[15].

Tabel 2 merangkum hasil pengujian terhadap fitur-fitur kritis, meliputi konektivitas, pembacaan sensor, antarmuka, logika otomasi fuzzy, dan kontrol manual.

Tabel 2. Tabel Hasil Pengujian Sistem

Skenario Pengujian	Input	Output yang diharapkan	Hasil Pengamatan	Status
Konektivitas & Inisialisasi	Menghubungkan daya sistem dan memulai proses <i>booting</i> .	1. Layar OLED menampilkan status awal. 2. Mikrokontroler terhubung ke WiFi & Server (Blynk/Telegram).	Sistem berhasil <i>online</i> dan siap menerima perintah.	Berhasil
Uji Akurasi Sensor Lingkungan	Memberikan cahaya lampu senter (>10.000 Lux) dan uap panas pada sensor.	Nilai Lux pada Dashboard naik >10.000 dan Suhu terbaca naik (30°C+).	Data pada OLED dan Dashboard berubah sesuai stimulus.	Berhasil

Skenario Pengujian	Input	Output yang diharapkan	Hasil Pengamatan	Status
Uji Deteksi Kelembapan Kain	Menempelkan sensor pada kain basah, lalu pada kain kering.	Indikator status berubah dari "Basah" (Analog >2000) menjadi "Kering" (Analog <500).	Jemuran ditarik masuk secara otomatis.	Berhasil
Uji Logika Prioritas (Hujan)	Meneteskan air pada sensor hujan saat tiang terbuka (180°).	Motor servo langsung berputar menutup tiang ke sudut 0° dalam < 2 detik.	Jemuran ditarik masuk dan notifikasi diterima pengguna.	Berhasil
Uji Logika Adaptif (Mendung)	Kondisi tidak hujan, kain basah, cahaya redup (~5000 Lux).	Motor servo bergerak ke posisi antisipasi 90° (setengah terbuka).	Jemuran berada di posisi setengah keluar.	Berhasil
Uji Otomasi Efisiensi (Kering)	Melepas sensor dari kain basah (Simulasi pakaian sudah kering).	Motor servo menutup tiang ke sudut 0° dan sistem berhenti menjemur.	Jemuran dimasukkan penuh.	Berhasil
Uji Sinkronisasi IoT	Membandingkan pembacaan Serial Monitor dengan App Blynk dan Pengujian Kendali Manual.	Data angka pada Aplikasi Blynk tampil sama persis dengan data pada Serial Monitor Arduino IDE serta dapat melakukan <i>override</i> pada kendali otomatis supaya berubah menjadi kendali manual.	Perubahan nilai dan data yang ditampilkan di Blynk tampil sama persis. Kendali otomatis berubah menjadi kendali manual ketika melakukan proses <i>override</i>	Berhasil
Uji Latensi Notifikasi	Memicu kondisi hujan secara tiba-tiba.	Pesan "Peringatan Hujan!" masuk ke Telegram dengan jeda waktu 2 detik.	Pesan peringatan diterima di Telegram dengan <i>delay</i> < 3 detik dan Jemuran masuk.	Berhasil

3) Pembahasan

Berdasarkan data eksperimen yang tersaji pada Tabel 2, pembahasan ini difokuskan pada analisis efektivitas solusi terhadap penyelesaian masalah serta komparasi kontribusi penelitian terhadap studi-studi terdahulu. Tujuan utama penelitian ini adalah mengatasi ketidakpastian cuaca yang menghambat proses pengeringan dan meningkatkan efisiensi melalui otomasi. Hasil pengujian menunjukkan bahwa integrasi algoritma *Fuzzy* berhasil menjawab tantangan tersebut melalui mekanisme proteksi adaptif dan efisiensi berbasis objek.

Pada Skenario 2 (Uji Akurasi Sensor Lingkungan), pengujian dilakukan dengan memberikan stimulus intensitas cahaya (>10.000 Lux) menggunakan senter serta stimulus udara panas. Skenario ini mensimulasikan kondisi cuaca siang hari yang terik dan cerah, di mana matahari bersinar optimal untuk proses pengeringan. Sebaliknya, pemberian tetesan air pada

sensor YL-83 secara langsung merepresentasikan kondisi turunnya hujan secara fisik di lingkungan sekitar.

Pada Skenario 3 (Uji Deteksi Kelembapan Kain), pengujian menggunakan medium kain basah dan kering secara langsung untuk mensimulasikan kondisi fisik pakaian yang sedang dijemur. Nilai analog tinggi (di atas 2000) merepresentasikan pakaian yang baru saja dicuci dan masih menahan banyak molekul air (basah kuyup). Sebaliknya, saat sensor dilepas atau kain telah kering sepenuhnya, nilai sensor turun di bawah ambang batas (<500). Angka inilah yang merepresentasikan kondisi bahwa pakaian telah kering sempurna dan siap untuk diangkat oleh pengguna.

Kecerdasan sistem divalidasi melalui Skenario 4, 5 dan 6 (Logika Kendali) yang terbagi ke dalam tiga simulasi cuaca utama. Simulasi Hujan Tiba-tiba (Skenario 4) mensimulasikan kejadian di mana pagi hari sangat cerah namun siang hari mendadak turun hujan. Ketika sistem mendeteksi tetesan air sistem langsung menarik tiang(0°), yang menunjukkan bahwa keselamatan pakaian dari air hujan menjadi prioritas mutlak, mengesampingkan variabel lainnya.

Selanjutnya, Simulasi Pakaian Telah Kering (Skenario 6) mensimulasikan kondisi di mana target pengeringan telah tercapai meskipun matahari di luar masih bersinar terik. Pergerakan motor servo untuk menarik tiang pada kondisi ini merepresentasikan tindakan perlindungan untuk mencegah pakaian dari paparan sinar UV berlebih dan debu yang dapat merusak serat kain.

Terakhir, Simulasi Cuaca Mendung (Skenario 5) mensimulasikan cuaca yang ambigu, yaitu kondisi langit redup (~5000 Lux) tanpa tetesan air hujan, sementara pakaian masih basah. Sistem merespons dengan memposisikan tiang jemuran setengah terbuka (90°). Respons antisipatif ini berarti sistem siap menutup cepat jika hujan tiba-tiba turun, namun tetap memberikan ruang bagi sirkulasi angin untuk membantu proses pengeringan pakaian karena tidak adanya sinar matahari.

Dari aspek interaksi pengguna, Skenario 7 dan 8 (Fitur IoT) merepresentasikan aktivitas keseharian pengguna modern yang memiliki mobilitas tinggi dan sedang berada jauh dari rumah (bekerja/bepergian). Uji latensi notifikasi hujan dan pakaian kering (Skenario 8) yang hanya berkisar 2 detik mensimulasikan fungsi Early Warning System (Sistem Peringatan Dini) yang sangat responsif, sehingga pengguna tidak perlu cemas memikirkan jemurannya. Sementara itu, uji mode manual merepresentasikan otoritas pengguna untuk mengambil alih kendali sistem secara paksa (*override*) ketika terjadi kondisi di luar prediksi algoritma.

Temuan dalam penelitian ini memberikan implikasi teoretis dan praktis yang memperkuat sekaligus melengkapi literatur sistem *smart home* yang ada. Sejalan dengan penelitian terdahulu milik Hidayatulloh dan Aryanto[8], penelitian ini mengonfirmasi bahwa metode Logika *Fuzzy* jauh lebih andal dibandingkan logika *threshold* sederhana dalam menangani parameter lingkungan yang fluktuatif. Namun, penelitian ini memberikan kontribusi kebaruan (*novelty*) yang mengisi celah pada penelitian [3], [5], dan [16], di mana sistem mereka hanya bersifat reaktif (menunggu hujan turun baru menutup). Sebaliknya, sistem yang diusulkan dalam riset ini bekerja secara proaktif dan terintegrasi dengan menggabungkan variabel lingkungan (eksternal) dan variabel kondisi pakaian (internal), merepresentasikan perkembangan dari *Automated Clothesline* menjadi *Smart Drying System* yang layak diadopsi sebagai standar baru perangkat IoT rumah tangga.

5. Simpulan

Penelitian ini berhasil merancang dan mengimplementasikan sistem jemuran cerdas berbasis IoT yang terintegrasi dengan metode Logika *Fuzzy*. Berdasarkan hasil pengujian eksperimental, sistem menunjukkan kinerja yang sangat baik dengan tingkat keberhasilan fungsional mencapai 100% pada seluruh skenario uji.

Implementasi algoritma *fuzzy* terbukti efektif dalam menangani ambiguitas cuaca, di mana sistem mampu merespons kondisi hujan dengan waktu reaksi di bawah 2 detik untuk menarik tiang (proteksi) dan secara akurat menghentikan proses penjemuran saat sensor mendeteksi nilai kelembapan kain di bawah ambang batas 500 (efisiensi). Dari segi akurasi perangkat keras, sensor lingkungan (BME280 dan BH1750) memiliki presisi tinggi dengan rata-rata *error* pengukuran di bawah 1% dibandingkan alat ukur standar. Selain itu, fitur notifikasi peringatan dini melalui Telegram berjalan stabil dengan rata-rata latensi pengiriman pesan sebesar 2-3 detik. Temuan ini mengukuhkan bahwa integrasi sensor kelembapan pakaian dan logika *fuzzy* mampu meningkatkan otonomi sistem dari sekadar alat pelindung hujan reaktif

menjadi sistem pengeringan cerdas yang proaktif. Untuk pengembangan selanjutnya, disarankan penggunaan sensor berat (*load cell*) guna meningkatkan presisi deteksi kekeringan pada kapasitas jemuran yang lebih besar serta penguatan struktur mekanik untuk ketahanan angin di lingkungan terbuka.

Daftar Referensi

- [1] K. Baras, L. Brito, and Q. F. Hassan, "Internet of Things," in *Advances in the Internet of Things*, Boca Raton: CRC Press, 2025, pp. 3–35. doi: 10.1201/9781003506638-2.
- [2] M. Soori, B. Arezoo, and R. Dastres, "Internet of things for smart factories in industry 4.0, a review," *Internet Things Cyber-Physical Syst.*, vol. 3, no. March, pp. 192–204, 2023, doi: 10.1016/j.iotcps.2023.04.006.
- [3] I. K. Wijayanti, Nurchim, and J. Maulindar, "Perancangan Smart Home Jemuran Otomatis Berbasis Internet of Things," *INFOTECH J.*, vol. 9, no. 1, pp. 183–189, 2023, doi: 10.31949/infotech.v9i1.5344.
- [4] D. Laksmiati and I. F. Amin, "Perancangan Jemuran Otomatis Berbasis Iot Menggunakan ESP32 Dan API," *J. Ilm. Giga*, vol. 27, no. 1, pp. 23–32, 2024, doi: 10.47313/jig.v27i1.3710.
- [5] A. Syam and A. M. Asmidun, "Alat Jemuran Otomatis Menggunakan Rain Sensor Dan Internet Of Things (IoT)," *J. Mediat.*, vol. 6, no. 1, pp. 1–5, 2024, doi: 10.59562/mediatik.v6i1.1352.
- [6] A. Azis and A. Chusyairi, "Prototype Alat Jemuran Pakaian Otomatis Menggunakan Arduino Berbasis Android," *Infotech J. Technol. Inf.*, vol. 7, no. 2, pp. 63–70, 2022, doi: 10.37365/jti.v7i2.112.
- [7] M. Muhandi, W. Sari, and Y. Irawan, "Prototype Jemuran Otomatis Menggunakan Sensor Raindrop Dan Sensor Ldr Berbasis Arduino Nano," *J. Ilmu Komput.*, vol. 10, no. 2, pp. 102–106, 2021, doi: 10.33060/jik/2021/vol10.iss2.222.
- [8] S. Hidayatulloh and J. Aryanto, "Sistem Pengendalian Jemuran Otomatis berbasis IoT dengan Logika Fuzzy untuk Pengkondisian Cuaca," *Edumatic J. Pendidik. Inform.*, vol. 7, no. 2, pp. 287–296, 2023, doi: 10.29408/edumatic.v7i2.21515.
- [9] A. P. Ramadhan, Jufrizel, Putut Son Maria, and Hilman Zarory, "Prototipe Jemuran Pakaian Otomatis Menggunakan Wemos D1R2 Dengan Notifikasi Suara dan Email Dari Thinger.io," *J. Sist. Cerdas*, vol. 5, no. 3, pp. 171–181, 2022, doi: 10.37396/jsc.v5i3.263.
- [10] Y. Hendrian, Y. P. Yudatama, and V. S. Pratama, "Jemuran Otomatis Menggunakan Sensor LDR, Sensor Hujan Dan Sensor Kelembaban Berbasis Arduino Uno," *J. Tek. Komput.*, vol. 6, no. 1, pp. 21–30, Jan. 2020, doi: 10.31294/jtk.v6i1.6683.
- [11] P. Y.M Bate, A. S. Wiguna, and D. A. Nugraha, "Sistem Penjemuran Otomatis Menggunakan Arduino Uno R3 Dengan Pendekatan Metode Fuzzy," *Kurawal - J. Teknol. Inf. dan Ind.*, vol. 3, no. 1, pp. 81–92, 2020, doi: 10.33479/kurawal.v3i1.306.
- [12] Sumarno, M. T. Banjarnahor, B. E. Damanik, I. Gunawan, and I. O. Kirana, "Jemuran_Pintar_Dengan_Sensor_Ldr_Sensor," vol. 1, no. 2, pp. 75–81, 2019.
- [13] N. A. Harahap, "Perancangan Prototype Jemuran Otomatis Menggunakan Sensor Air Dan Sensor Ldr Berbasis Mikrokontroler Arduino Uno Dengan Metode Flc," *J. Media Inform. Budidarma*, vol. 2, no. 1, pp. 15–25, 2018, doi: 10.30865/mib.v2i1.814.
- [14] N. Laila and Taufiq, "Rancang Bangun Kendali Atap Jemuran Otomatis Berbasis ATMEGA 328," *J. Tek. Inform.*, vol. 8, no. 1, pp. 45–52, 2019.
- [15] A.N. Fathoni and U.Y. Oktiawati, "Blackbox Testing terhadap Prototipe Sistem Monitoring Kualitas Air Berbasis IoT," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 10, no. 4, pp. 362–368, 2021, doi: 10.22146/jnteti.v10i4.2095.
- [16] M.D. Supiannor, F. Fitriyadi, & N. Rosmawanti, "Model Atap Jemuran Gabah Otomatis Berbasis Mikrokontroler Atmega328. *Progresif: Jurnal Ilmiah Komputer*, vol. 18, no. 1, pp. 43–54, 2022.