

Rancang Bangun *Internet of Things* Purwarupa Jemuran Otomatis Berbasis Website

Nathanael Sharron¹

¹Program Studi Teknik Informatika, Fakultas Teknik, Universitas Kristen Surakarta

E-mail : nathanaeldharron@gmail.com

*Koresponden email: nathanaeldharron@gmail.com

Diterima : 17 Juni 2026

Disetujui:24 Juli 2026

ABSTRAK

Penelitian ini bertujuan untuk membangun aplikasi monitoring dan kendali purwarupa jemuran otomatis berbasis *Internet of Things* (IoT) sebagai solusi mengatasi ketidakpraktisan proses penjemuran pakaian akibat perubahan cuaca mendadak. Permasalahan utama yang dihadapi oleh masyarakat dengan mobilitas tinggi, seperti mahasiswa dan pekerja, adalah risiko pakaian kembali basah dan apek ketika ditinggal karena sistem penjemuran yang masih konvensional. Metode penelitian yang digunakan meliputi analisis kebutuhan fungsional dan non-fungsional, perancangan sistem menggunakan *flowchart* dan DFD, serta implementasi perangkat keras menggunakan mikrokontroler ESP32 yang terintegrasi dengan sensor *Raindrops* dan DHT11. Antarmuka pemantauan (*dashboard*) dibangun menggunakan teknologi HTML, CSS, dan JavaScript yang terhubung secara *real-time* dengan *Firebase Realtime Database*. Hasil penelitian menunjukkan bahwa aplikasi yang dikembangkan sukses memantau parameter lingkungan (suhu, kelembapan, cuaca) dan mengendalikan pergerakan jemuran secara otomatis maupun manual. Berdasarkan pengujian *User Acceptance Test* (UAT), sistem ini memperoleh indeks kelayakan sebesar 93,61%, sehingga masuk ke dalam kategori "Sangat Layak" untuk mempermudah monitoring jemuran jarak jauh.

Key word : *Dashboard website, ESP32, Firebase, IoT, Jemuran otomatis*

ABSTRACT

This study aims to design and develop a prototype of Internet of Things (IoT)-based automatic clothing clothesline monitoring and control application to address the impracticality of drying clothes due to sudden weather changes. The main problem faced by high-mobility individuals, such as students and workers, is the risk of clothes getting wet and musty when left behind due to conventional drying systems. The research method includes functional and non-functional requirements analysis, system design using flowcharts and DFDs, and hardware implementation using an ESP32 microcontroller integrated with Raindrops and DHT11 sensors. The monitoring interface (*dashboard*) was built using HTML, CSS, and JavaScript technologies, which are connected in real-time to the *Firebase Realtime Database*. The results show that the developed application successfully monitors environmental parameters (temperature, humidity, weather) and controls the movement of the clothesline both automatically and manually. Based on User Acceptance Test (UAT) testing, the system obtained a feasibility index of 93.61%, placing it in the "Highly Feasible" category for facilitating remote clothesline monitoring.

Key word: Automatic clothesline, ESP32, Firebase, IoT, Website dashboard

1. Pendahuluan

Aktivitas menjemur pakaian merupakan kegiatan rumah tangga rutin yang sangat bergantung pada kondisi cuaca alami, khususnya radiasi panas matahari. Namun, tantangan utama yang sering dihadapi oleh masyarakat dengan

mobilitas tinggi adalah perubahan cuaca yang tidak menentu dan dapat terjadi secara mendadak. Masalah muncul ketika jemuran ditinggalkan tanpa pengawasan; cuaca yang tiba-tiba berubah dari terik menjadi hujan dapat menyebabkan pakaian yang hampir kering menjadi basah kembali dan memicu bau apek akibat kelembapan tinggi. Proses pemantauan dan penarikan jemuran yang masih bersifat konvensional dinilai tidak lagi efektif bagi individu yang harus beraktivitas di luar rumah sepanjang hari. Oleh karena itu, diperlukan sebuah inovasi teknologi yang mampu memantau kondisi lingkungan sekitar area penjemuran secara akurat sekaligus mengendalikannya dari jarak jauh.

Seiring dengan berkembangnya era digital, teknologi *Internet of Things* (IoT) hadir sebagai solusi modern untuk otomatisasi berbagai perangkat rumah tangga dengan prinsip menghubungkan perangkat keras ke jaringan internet demi pertukaran data secara *real-time* [6]. Dalam beberapa tahun terakhir, penelitian mengenai purwarupa deteksi cuaca dan pemantauan parameter lingkungan telah banyak dikembangkan, seperti penggunaan sensor air untuk mendeteksi datangnya hujan lokal [1]. Selain itu, pemanfaatan sensor DHT11 juga sering diimplementasikan untuk membaca fluktuasi nilai suhu dan kelembapan udara di area sekitar objek secara berkala [2]. Integrasi sensor-sensor pemantau lingkungan ini terbukti sangat krusial, tidak hanya untuk kebutuhan rumah tangga tetapi juga untuk menjaga stabilitas parameter ruang penyimpanan yang sensitif terhadap perubahan termal [3].

Meskipun sistem pemantauan lingkungan berbasis mikrokontroler telah banyak diteliti, sebagian besar sistem pemantauan cuaca terdahulu masih menggunakan komunikasi jarak pendek atau infrastruktur jaringan lokal (LAN). Untuk memperluas jangkauan transmisi data, penggunaan mikrokontroler modern seperti ESP32 yang memiliki performa komputasi tinggi dan konektivitas nirkabel jarak jauh menjadi sangat penting dalam mendukung efisiensi pengiriman data sensor [4]. Di sisi lain, agar data yang ditangkap oleh sensor di lapangan dapat diakses oleh pengguna kapan saja dan dari mana saja, diperlukan sebuah sistem manajemen basis data yang andal. Pemanfaatan *Firestore Realtime Database* menawarkan solusi penyimpanan data yang dinamis, ringan, dan mampu melakukan sinkronisasi data secara instan tanpa ada jeda waktu (*latency*) yang berarti antara perangkat keras dan aplikasi pengguna [5].

Untuk mengeksekusi rancangan tersebut secara terstruktur dan terukur, penelitian ini dijalankan dengan mengadopsi panduan metodologi penelitian praktis guna memastikan setiap tahapan pengembangan sistem berjalan secara efektif dan sesuai dengan kebutuhan pengguna [7]. Sebagai interaksi utama, sebuah aplikasi *dashboard* monitoring berbasis website dibangun menggunakan kombinasi HTML, CSS, dan JavaScript. Melalui platform website ini, pengguna dapat memantau status cuaca, grafik suhu, dan kelembapan secara *real-time*. Selain itu, sistem ini menyediakan dua mode kendali: "Mode Otomatis" di mana motor penggerak jemuran akan merespons langsung hasil pembacaan sensor hujan, serta "Mode Manual" yang memberikan hak akses penuh kepada pengguna untuk menggerakkan jemuran melalui tombol virtual pada halaman website. Melalui implementasi sistem ini, diharapkan proses penjemuran pakaian dapat berjalan lebih optimal, aman, serta memberikan efisiensi waktu yang signifikan bagi masyarakat modern.

2. Metodologi Penelitian

Mengikuti panduan metodologi penelitian praktis yang efektif guna memastikan setiap tahapan pengembangan berjalan secara terstruktur [7], penelitian ini dilaksanakan melalui beberapa tahapan sistematis. Tahapan tersebut meliputi analisis kebutuhan sistem (perangkat keras dan perangkat lunak), perancangan arsitektur jaringan IoT, serta pemodelan alur kerja sistem menggunakan *flowchart*.

2.1. Analisis kebutuhan sistem

Tahap awal dilakukan dengan memetakan seluruh komponen pendukung agar sistem dapat berfungsi secara optimal dan berkelanjutan. Kebutuhan tersebut dibagi menjadi dua kategori utama, yaitu:

a. Perangkat Keras (Hardware)

Komponen fisik yang digunakan dalam membangun purwarupa jemuran otomatis ini meliputi:

1. Mikrokontroler ESP32: Berperan sebagai pusat pemroses data (*brain*) yang mengolah input dari sensor dan mengendalikan aktuator, serta menyediakan konektivitas Wi-Fi internal untuk mengirim data ke *cloud* [4]
2. Sensor Hujan (Raindrops): Berperan untuk mendeteksi keberadaan dan intensitas butiran air hujan di sekitar area penjemuran [1].
3. Sensor Suhu dan Kelembapan (DHT11): Berperan untuk membaca fluktuasi nilai suhu lingkungan serta kelembapan udara secara berkala [2], [3].

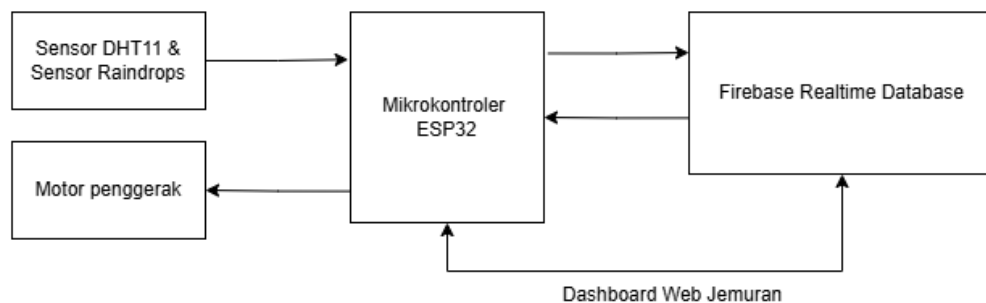
4. Motor Servo : Berperan sebagai aktuator mekanis yang menggerakkan fisik jemuran (menarik masuk atau mengeluarkan jemuran).
- b. Perangkat Lunak (Software)

Komponen sistem perangkat lunak yang digunakan untuk membangun ekosistem monitoring meliputi:

1. Arduino IDE: Digunakan untuk menuliskan dan mengompilasi kode program (*sketch*) berbasis C/C++ ke dalam mikrokontroler ESP32 [8].
2. Firebase Realtime Database: Digunakan sebagai basis data berbasis *cloud* untuk menyimpan dan menyinkronkan status sensor serta kendali motor secara instan (*real-time*) [5].
3. Front-End Website Platform (HTML, CSS, JavaScript): Digunakan untuk membangun antarmuka halaman *dashboard* monitoring yang interaktif dan ringan sehingga mudah diakses oleh pengguna.

2.2. Arsitektur Jaringan IoT

Sistem ini dirancang dengan mengintegrasikan perangkat keras di lapangan dengan antarmuka pengguna melalui jaringan internet menggunakan arsitektur *cloud*. Alur transmisi data pada sistem diilustrasikan pada blok diagram berikut:



Gambar 1. Arsitektur Jaringan IoT

Sensor DHT11 dan Raindrops mengumpulkan data analog/digital dari lingkungan, lalu diteruskan ke ESP32 [1], [2]. ESP32 mengirimkan data tersebut ke Firebase Realtime Database melalui jaringan Wi-Fi lokal [4], [5]. Aplikasi dashboard website yang dibuka oleh pengguna akan secara otomatis membaca perubahan data pada Firebase untuk ditampilkan pada grafik halaman website secara real-time [5]. Sebaliknya, jika pengguna menekan tombol kendali pada halaman website, data instruksi akan dikirim ke Firebase dan langsung diunduh oleh ESP32 untuk menggerakkan motor.

2.3. Perancangan alur kerja sistem

Sistem kendali purwarupa jemuran otomatis ini dirancang memiliki dua mode operasi utama yang dapat dialihkan melalui dashboard website, yaitu Mode Otomatis dan Mode Manual. Alur logika pemrosesan datanya digambarkan sebagai berikut:

- a. Inisialisasi: Sistem dinyalakan, ESP32 melakukan koneksi ke jaringan Wi-Fi dan menghubungkan diri ke server Firebase [4], [5].
- b. Pembacaan Sensor: Sensor DHT11 dan Raindrops mulai membaca parameter cuaca luar ruangan secara kontinu [2]. Data dikirim dan diperbarui ke dalam Firebase [5].
- c. Pengecekan Mode Kendali: Sistem memeriksa status variabel mode kendali yang aktif di Firebase [5]:
 1. Jika Mode Otomatis aktif: Mikrokontroler akan langsung mengambil keputusan berdasarkan input sensor [4]. Jika sensor mendeteksi hujan atau suhu drop (kondisi mendung/malam), maka ESP32 mengirim perintah ke motor untuk menarik jemuran ke dalam tempat teduh [1], [5]. Jika sensor mendeteksi tidak ada hujan dan suhu panas (terik), motor akan bergerak mengeluarkan jemuran [1], [5].
 2. Jika Mode Manual aktif: Logika otomatis sensor akan diabaikan oleh sistem. Pergerakan motor sepenuhnya hanya akan mengeksekusi nilai *switch button* (Buka/Tutup) yang ditekan secara sadar oleh pengguna melalui halaman *dashboard* website.

3. Hasil Dan Pembahasan

3.1. Hasil implementasi perangkat keras

Purwarupa sistem jemuran otomatis ini telah berhasil dirakit dengan mengintegrasikan mikrokontroler ESP32 sebagai pusat pemroses data dengan komponen sensor dan aktuator di lapangan. Sensor DHT11 dipasang pada posisi yang terlindung dari air tetapi tetap terpapar udara bebas untuk membaca suhu dan kelembapan secara akurat. Sensor *Raindrops* diletakkan pada area terbuka dengan sudut kemiringan tertentu agar butiran air hujan dapat dideteksi dengan cepat dan langsung mengalir turun untuk menghindari genangan air yang dapat memanipulasi pembacaan pasca-hujan. Perangkat keras ini terhubung ke jaringan internet menggunakan modul Wi-Fi internal ESP32 untuk mengirimkan data sensor ke *Firebase Realtime Database* secara berkala.

Hasil Purwarupa dapat dilihat pada gambar 2:

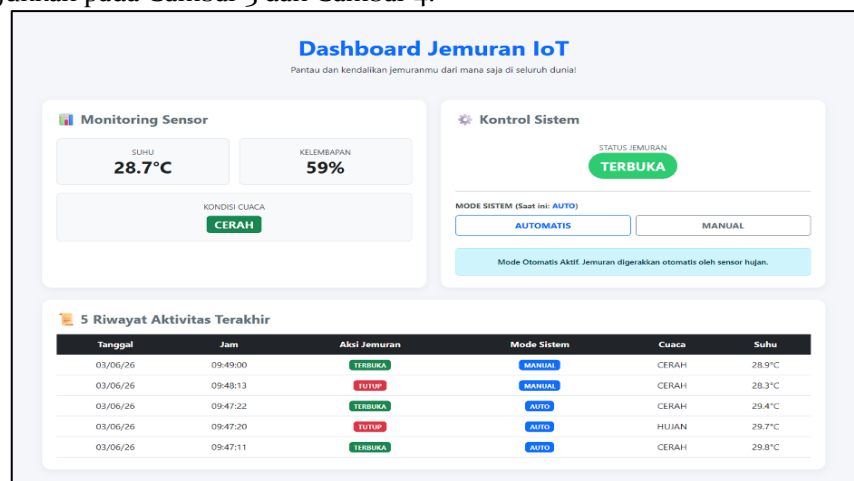


Gambar 2 Implementasi arsitektur *iot* pada purwarupa

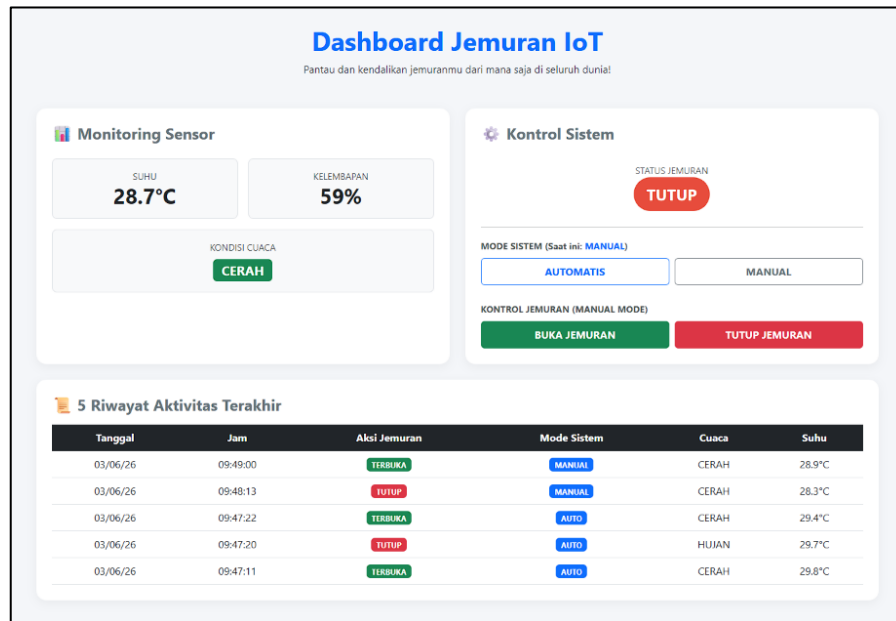
3.2. Hasil implementasi antarmuka *dashboard website*

Halaman *dashboard* website dan kendali purwarupa jemuran otomatis diimplementasikan menggunakan kombinasi HTML, CSS, dan JavaScript yang dirancang secara responsif. Untuk menguji performa sistem secara langsung, halaman *dashboard* website dapat diakses secara publik melalui GitHub Pages dengan kode akses "123456" (<https://zwaneee.github.io/jemuran-iot/skripsijemuraniot>). Berdasarkan hasil perancangan, antarmuka aplikasi menggunakan skema warna dominan biru cerah dan putih untuk memberikan kesan bersih, modern, dan tingkat keterbacaan yang tinggi (*high readability*) bagi pengguna saat memantau sistem.

Secara visual, komponen utama dari halaman *dashboard* website ini dibagi menjadi tiga bagian panel informatif, seperti yang ditunjukkan pada Gambar 3 dan Gambar 4:



Gambar 3 *Dashboard* jemuran *iot* dalam mode otomatis



Gambar 4 Dashboard jemuran iot dalam mode manual

a. Panel monitoring

Panel ini terletak pada bagian atas antarmuka yang berfungsi menyajikan seluruh data parameter lingkungan secara real-time dari lapangan berdasarkan pembacaan sensor. Panel monitoring ini memuat tiga komponen informasi utama:

1. Suhu: Menampilkan parameter temperatur udara saat ini di sekitar maket jemuran dalam satuan derajat Celsius (°C) yang bersumber dari sensor DHT11.
2. Kelembapan: Menampilkan parameter tingkat kebasahan atau kelembapan udara lingkungan dalam bentuk persentase (%) melalui sensor DHT11.
3. Kondisi Cuaca: Menampilkan status cuaca hasil interpretasi sensor Raindrops. Status akan menampilkan teks "Cerah" dengan indikator visual hijau saat kering, dan akan otomatis berubah secara instan apabila sensor mendeteksi adanya air hujan.

b. Panel kontrol sistem

Panel ini merupakan pusat interaksi utama pengguna untuk mengatur operasional perangkat keras dan memantau status fisik jemuran. Di dalam panel kontrol sistem ini terdapat tiga bagian penting:

1. Status Jemuran: Menampilkan informasi visual mengenai kondisi posisi fisik maket jemuran saat ini, apakah dalam status sedang "Terbuka" (berada di luar ruangan) atau "Tertutup" (berada di dalam tempat teduh).
2. Mode Sistem (Otomatis/Manual): Tombol interaktif (switch mode) untuk mengalihkan sistem kerja. Pada Mode Otomatis (Gambar 1), seluruh pergerakan motor digerakkan secara mandiri oleh ESP32 berdasarkan parameter sensor, dan tombol aksi manual akan dinonaktifkan.
3. Tombol Kendali Aksi Manual: Tombol virtual dinamis yang terdiri dari tombol "BUKA JEMURAN" dan "TUTUP JEMURAN". Tombol-tombol ini hanya akan muncul dan aktif ketika pengguna memilih Mode Manual (Gambar 2). Melalui tombol ini, pengguna diberikan hak akses penuh untuk menggerakkan posisi jemuran secara sadar kapan saja dari jarak jauh dengan mengabaikan seluruh hasil pembacaan sensor lingkungan.

c. Panel riwayat

Panel ini terletak pada bagian bawah halaman dashboard yang berfungsi sebagai media pelacakan log aktivitas alat. Panel ini menampilkan tabel data dinamis yang mencatat 5 aktivitas terakhir dari pergerakan sistem jemuran IoT. Informasi detail yang direkam secara berurutan pada panel riwayat ini meliputi:

1. Tanggal dan Jam: Mencatat waktu presisi kapan aktivitas perubahan posisi jemuran tersebut terjadi.
2. Aksi Jemuran: Menampilkan riwayat tindakan mekanis alat, apakah jemuran bergerak membuka atau menutup.
3. Mode Sistem: Menampilkan informasi mode operasional yang sedang aktif saat aksi terjadi (Otomatis atau Manual).
4. Cuaca dan Suhu: Mencatat kondisi parameter lingkungan sekitar yang terekam pada basis data tepat pada saat posisi jemuran tersebut mengalami perubahan.

3.3 Hasil pengujian Sistem dan Analisis Data

Untuk menguji keandalan sistem IoT yang telah dibangun, dilakukan serangkaian pengujian yang meliputi uji fungsionalitas sensor, uji kecepatan transmisi data (*latency*), serta uji kelayakan pengguna.

a. Hasil Pengujian Fungsionalitas Sensor dan Aktuator

Pengujian ini dilakukan untuk memastikan akurasi pembacaan sensor serta ketepatan respons motor penggerak berdasarkan skenario kondisi lingkungan yang berbeda. Hasil pengujian fungsionalitas dirangkum dalam Tabel 1.

Tabel 1 Hasil pengujian

No	Kondisi Lingkungan	Pembacaan Sensor	Mode Aktif	Status Jemuran (Output)	Keterangan
1	Kondisi kering & terik matahari	Suhu tinggi, tidak ada air	Otomatis	Terbuka (jemuran keluar)	Berhasil
2	Kondisi turun hujan (gerimis/deras)	Adanya air pada panel sensor	Otomatis	Tertutup (jemuran masuk)	Berhasil
3	Kondisi cuaca mendung / malam	Suhu rendah, kelembapan tinggi	Otomatis	Tertutup (jemuran masuk)	Berhasil
4	Interaksi tombol "buka" pada website	Mengabaikan kondisi sensor	Manual	Terbuka (jemuran keluar)	Berhasil
5	Interaksi tombol "tutup" pada website	Mengabaikan kondisi sensor	Manual	Tertutup (jemuran masuk)	Berhasil

Berdasarkan Tabel 1, pengujian menunjukkan bahwa dalam "Mode Otomatis", logika mikrokontroler berhasil mengeksekusi pergerakan motor berdasarkan perubahan parameter cuaca secara tepat waktu. Begitu pula dalam "Mode Manual", kendali penuh pengguna melalui *dashboard* website mampu mengabaikan pembacaan sensor dan menggerakkan motor secara instan.

b. Hasil Pengujian Kecepatan Transmisi Data (Latency)

Pengujian transmisi dilakukan untuk mengukur waktu yang dibutuhkan data sensor dari ESP32 untuk sampai dan diperbarui pada halaman *dashboard* website melalui *Firestore Realtime Database*. Pengujian dilakukan sebanyak 10 kali percobaan pada kondisi jaringan internet yang stabil. Rata-rata *delay* atau *latency* pengiriman data yang diperoleh berkisar antara 0,4 hingga 1,2 detik. Nilai *latency* yang berada di bawah ambang batas 2 detik ini membuktikan bahwa arsitektur penyimpanan berbasis *Firestore* sangat andal untuk mendukung pemantauan berskala *real-time*.

c. Hasil Pengujian Kelayakan (User Acceptance Test - UAT)

Untuk mengukur tingkat kepuasan dan kelayakan aplikasi dari sudut pandang pengguna akhir, dilakukan pengujian *User Acceptance Test* (UAT) dengan menyebarkan kuesioner kepada responden potensial (mahasiswa dan masyarakat umum dengan mobilitas tinggi). Kuesioner ini menguji beberapa aspek utama, seperti kemudahan penggunaan (*usability*), keandalan sistem kendali jarak jauh, serta kecepatan pembaruan data visual pada website.

Hasil perhitungan dari pengujian UAT memberikan skor persentase kelayakan sebesar 93,61%. Mengacu pada kriteria penilaian indeks kelayakan perangkat lunak, hasil tersebut menempatkan aplikasi monitoring dan kendali purwarupa jemuran otomatis ini ke dalam kategori "Sangat Layak" untuk diimplementasikan dalam membantu efisiensi aktivitas rumah tangga sehari-hari.

4. Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian yang telah dilakukan, penelitian ini berhasil membangun sistem aplikasi monitoring dan kendali purwarupa jemuran otomatis berbasis *Internet of Things* (IoT). Integrasi mikrokontroler ESP32 dengan sensor *Raindrops* dan DHT11 terbukti mampu membaca parameter lingkungan secara akurat dan responsif. Penggunaan *Firestore Realtime Database* memberikan performa transmisi data yang andal dengan rata-rata *latency* yang sangat rendah antara 0,4 hingga 1,2 detik dari perangkat keras ke *dashboard website*. Melalui pengujian *User Acceptance Test* (UAT), sistem ini memperoleh indeks kelayakan sebesar 93,61% yang menempatkannya dalam kategori "Sangat Layak". Hal ini membuktikan bahwa aplikasi yang dibangun sukses memberikan kemudahan bagi pengguna dalam memantau dan mengendalikan jemuran jarak jauh secara efektif.

5. Singkatan:

IoT	: <i>Internet of Things</i>
HTML	: <i>HyperText Markup Language</i>
CSS	: <i>Cascading Style Sheets</i>
UAT	: <i>User Acceptance Testing</i>

6. Daftar Pustaka:

- [1] A. Widodo and A. Sumaedi, "Prototipe Deteksi Hujan Berbasis Arduino Uno," *Jurnal Teknologi Informasi (JTI)*, vol. 4, no. 1, pp. 12–18, 2023. Available: <https://ejournal.antarbangsa.ac.id/jti/article/view/506>
- [2] R. M. Abdurrohman, K. Barriyah, K. Nursuciliyat, K. A. Rochim, and H. Hasanah, "Prototipe Monitoring Suhu dan Kelembapan Secara Real-Time Menggunakan DHT11 dan Arduino Uno R3," *Jurnal ICTEE*, vol. 4, no. 2, pp. 29–36, 2023. Available: <https://pdfs.semanticscholar.org/ciod/fdf5e57e6a7d4ebd903d6ecf9eb8cee092c12.pdf>
- [3] E. M. Yusuf and F. Pratama, "Rancang bangun sistem monitoring suhu dan kelembaban ruang server berbasis IoT menggunakan Arduino pada PT Bintaro Serpong Damai," *Jurnal Sistem Komputer dan Kecerdasan Buatan*, vol. 7, no. 1, pp. 15–22, 2023. Available: <https://jurnal.tau.ac.id/index.php/siskom-kb/article/view/453>
- [4] M. Y. Muhaimin, A. D. Santoso, and R. Maulidiah, "Rancang bangun pendeteksi cuaca menggunakan lora (long range) berbasis mikrokontroler ESP32," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 13, no. 2, pp. 820–828, 2025. Available: <https://repository.poltekpel-sby.ac.id/id/eprint/673/>
- [5] A. S. M. Suradi, A. S. S. Alam, A. Bahtiar, and A. R. Johanis, "Pemanfaatan firebase realtime database dalam perancangan aplikasi penilaian siswa smk negeri 2 pangkep secara realtime," *Jurnal Komputer dan Informatika*, vol. 6, no. 2, pp. 146–154, 2024. Available: <https://www.ioinformatic.org/index.php/JUKI/article/view/444>
- [6] A. Syahfitri, "Internet of Things (IoT), Sejarah, Teknologi, dan Penerapannya," *Jurnal Ilmiah Teknik Elektro*, vol. 3, no. 1, pp. 113–120, 2025. Available: https://www.academia.edu/download/125750981/Internet_of_Things_IoT.pdf
- [7] M. A. Mukhyi, *METODOLOGI PENELITIAN PANDUAN PRAKTIK PENELITIAN YANG EFEKTIF*. Malang: PT. Literasi Nusantara Abadi Grup, 2023.
- [8] Sodik, K. Setiawan, dan R. F. Aula, "Penerapan IoT dengan algoritma fuzzy dan mikrokontroler ESP32 dalam monitoring penyiraman," *J. Indon. Manag. Inform. Commun.*, vol. 5, no. 3, hlm. 2915–2924, 2024. Available: <http://journal.stmiki.ac.id/index.php/jimik/article/view/977>
- [9] R. Gustia, M. Hasanah, W. Febriani, dan A. Fradana, "Perancangan smart home berbasis IoT menggunakan ESP32, Telegram dan Spreadsheet," *J. Pustaka AI*, vol. 5, no. 3, hlm. 695–700, 2025. Available: <https://jurnal.pustakagalerimandiri.co.id/index.php/pustakaai/article/view/1482>
- [10] L. A. Wenno, B. J. Waworuntu, S. N. Rumokoy, H. G. Kaunang, dan S. B. Dodie, "Konsep pemanfaatan teknologi IoT berbasis ESP32 untuk sakelar cerdas pada sistem PLTS rakit rumah apung," *J. Electr.*, vol. 4, no. 2, hlm. 1–9, 2025. Available: <http://jurnal.polimdo.ac.id/index.php/elektrik/article/view/1171>
- [11] D. E. Putra, E. Rosman, H. Amnur, F. G. Katrina, M. Hasanah, dan R. I. Salam, *Konsep Dasar Internet of Things (IoT) dengan Mikrokontroler ESP32*. Padang, Indonesia: Pustaka Galeri Mandiri, 2024.
- [12] J. Aliman dan A. Ibrahim, *Panduan Asas Arduino dan ESP32*, Edisi 1. Melaka, Malaysia: Politeknik Merlimau Melaka, 2025.

-
- [13] R. F. Purnomo, O. W. Purbo, dan R. Z. A. Aziz, *Firestore: Membangun Aplikasi Berbasis Android*. Yogyakarta, Indonesia: Deepublish, 2020.
 - [14] B. Kristiawan and N. Sutarti, "Aplikasi Notulen Rapat berbasis HTML Sebagai Penyimpanan Data Arsip dan Catatan Rapat (Studi Kasus: Program Studi Teknik Informatika UKTS)," *Jurnal Solo Teknologi*, vol. 2, no. 1, pp. 30–41, 2026. Available: <https://jurnal.ukts.ac.id/index.php/soloteknologi/article/view/112>
 - [15] D. Hercog, T. Lerher, M. Truntič, dan O. Težak, "Design and implementation of ESP32-based IoT devices," *Sensors*, vol. 23, no. 15, hlm. 1–20, 2023. Art. no. 6739, doi: 10.3390/s23156739