

# PROTOTIPE SISTEM PENJEMUR PAKAIAN OTOMATIS BERBASIS IOT DENGAN PEMANTAUAN VISUAL ESP32-CAM DAN KENDALI APLIKASI MOBILE

Nur Cahyo Juniyanto<sup>1\*</sup>, Imam Suharjo<sup>2</sup>

<sup>1,2</sup> Program Studi Informatika, Universitas Mercu Buana Yogyakarta, Kampus 2, Depok, Sleman, Yogyakarta 55283, Indonesia

---

Received: xxxx-xx-xx

Accepted: xx-xx-xx

**Keywords:**

Sistem Jemuran Otomatis; Internet of Things; ESP32-CAM; Firebase; Waterfall.

**Corespondent Email:**

nurcahyojuniyanto@gmail.com



JITET is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

**Abstrak.** Ketidakpastian cuaca sering menjadi kendala dalam proses pengeringan pakaian secara konvensional, terutama bagi masyarakat dengan mobilitas tinggi. Penelitian ini bertujuan membangun prototipe sistem penjemur pakaian otomatis berbasis Internet of Things (IoT) yang dilengkapi fitur pemantauan visual real-time. Sistem ini dikembangkan menggunakan metode Waterfall, mengintegrasikan mikrokontroler Wemos D1 R1 dengan sensor hujan, cahaya (LDR), dan suhu (DHT22) untuk mendeteksi kondisi lingkungan, serta motor DC sebagai penggerak otomatis jemuran. Keunggulan utama sistem ini adalah integrasi modul ESP32-CAM yang memungkinkan pengguna memverifikasi kondisi aktual jemuran secara visual melalui aplikasi. Seluruh pemantauan dan kendali sistem, baik mode otomatis maupun manual, dilakukan melalui aplikasi Android yang dibangun dengan Android Studio dan terhubung ke Firebase. Hasil pengujian menunjukkan sistem mampu merespons perubahan cuaca secara akurat untuk melindungi pakaian dari hujan, serta memberikan fleksibilitas monitoring jarak jauh yang meningkatkan efisiensi aktivitas rumah tangga.

**Abstract.** Weather uncertainty often hinders conventional clothes drying processes, particularly for individuals with high mobility. This research aims to develop a prototype of an Internet of Things (IoT)-based automatic clothes drying system equipped with real-time visual monitoring features. Developed using the Waterfall method, the system integrates a Wemos D1 R1 microcontroller with rain, light (LDR), and temperature (DHT22) sensors to detect environmental conditions, utilizing a DC motor for automatic actuation. A key advantage of this system is the integration of the ESP32-CAM module, which enables users to visually verify the actual condition of the laundry via a mobile application. All monitoring and control functions, in both automatic and manual modes, are facilitated through an Android application built with Android Studio and connected to Firebase. Test results demonstrate the system's capability to accurately respond to weather changes to protect clothes from rain, while providing flexible remote monitoring that enhances household efficiency.

---

## 1. PENDAHULUAN

Kegiatan menjemur pakaian merupakan aktivitas rutin rumah tangga yang sangat bergantung pada kondisi cuaca. Di wilayah dengan cuaca yang tidak menentu, terutama saat musim hujan, proses pengeringan pakaian

menjadi tantangan tersendiri. Tidak jarang, pakaian yang sudah dijemur harus diangkat secara mendadak karena hujan turun tiba-tiba. Hal ini tentu merepotkan, terutama bagi penghuni rumah yang memiliki kesibukan di luar rumah[1].

Seiring dengan kemajuan teknologi, khususnya Internet of Things (IoT), berbagai aktivitas rumah tangga kini dapat diotomatisasi untuk meningkatkan efisiensi dan kenyamanan[2]. Dengan memanfaatkan sensor yang dapat mendeteksi perubahan cuaca, sistem dapat didesain agar mampu beradaptasi secara otomatis terhadap perubahan cuaca[3]. Ditambah dengan integrasi ke *system cloud* dapat membuat pengguna mengontrol sistem dari jarak jauh.

Pada penelitian sebelumnya, telah dibahas prototype alat kendali otomatis penjemuran pakaian menggunakan mikrokontroler NodeMCU ESP8266 dengan sensor rain drop dan menggunakan aplikasi Blynk sebagai monitor dan kontrolnya[4]. Selain penggunaan sensor rain drop, beberapa penelitian sebelumnya juga menambahkan sensor LDR [5], [6] dan sensor suhu untuk mendeteksi perubahan cuaca serta menggunakan Telegram bot sebagai media kontrol dan monitornya[7].

Perbedaan penelitian yang sudah dilakukan sebelumnya dengan penelitian ini adalah pengembangan aplikasi pengguna akan dibuat menggunakan Android Studio sehingga menghasilkan aplikasi Android yang memiliki tampilan monitor dan kontrol yang lebih lengkap. Selain itu, sistem juga akan dilengkapi dengan ESP-CAM untuk pemantauan visual sebagai sarana verifikasi langsung dengan cara memperlihatkan kondisi aktual melalui foto di lokasi jemuran sehingga diharapkan akan menghasilkan sebuah sistem yang konsisten dan meyakinkan.

Penelitian ini bertujuan untuk membangun prototipe sistem penjemur pakaian otomatis berbasis IoT dengan menggunakan sensor suhu, cahaya, dan rain drop. Selain itu, penelitian ini juga bertujuan untuk mengintegrasikan ESP32-CAM pada sistem sebagai alat bantu pemantauan visual di area jemuran. Pengembangan aplikasi mobile menjadi bagian dari penelitian ini, yang berfungsi sebagai media untuk monitoring dan kontrol jarak jauh terhadap sistem. Terakhir, penelitian ini akan melakukan pengujian untuk mengukur keefektifan sistem dalam merespons perubahan cuaca secara real-time dan melakukan pengujian keandalan ESP32-CAM dalam sistem ini.

## 2. TINJAUAN PUSTAKA

### 2.1. Internet of Things (IoT)

Internet of Things (IoT) adalah sebuah konsep di mana objek fisik dilengkapi dengan sensor, perangkat lunak, dan teknologi lain untuk terhubung dan bertukar data dengan perangkat dan sistem lain melalui jaringan internet[8].

### 2.2. Wemos D1 R1 ESP8266

Wemos D1 R1 merupakan modul pengembangan mikrokontroler berbasis ESP8266 yang menyediakan solusi ekonomis dalam implementasi sistem berbasis wireless. Modul ini memiliki RAM sebesar 4 MB dan chipset CH340 sebagai konverter antarmuka USB ke serial, sehingga mampu menyediakan koneksi Wi-Fi dengan cepat dan efisien. Wemos D1 R1 mendukung lingkungan pengembangan Arduino IDE, sehingga memudahkan pengembangan aplikasi IoT dengan menggantikan papan Arduino Uno secara langsung tanpa memerlukan perubahan signifikan pada perangkat keras maupun perangkat lunak yang telah digunakan sebelumnya[9].

### 2.3. Modul ESP-CAM

ESP32-CAM adalah modul pengembangan *open source* yang lengkap dengan mikrokontroler terintegrasi, WiFi, Bluetooth, dan kamera video. Modul ini bisa berfungsi mandiri dan cocok untuk berbagai proyek, terutama yang membutuhkan fitur pengambilan gambar, pengenalan, atau deteksi wajah. Pengguna dapat dengan mudah memprogram modul ini melalui Arduino IDE untuk mengakses library dan memanfaatkan semua fitur yang ada, ditambah dukungan slot microSD untuk penyimpanan[10].

### 2.4. Arduino IDE

Arduino IDE berfungsi sebagai lingkungan pengembangan terintegrasi untuk menyusun, menyunting, dan mengunggah kode program (sketch) ke dalam mikrokontroler[11]. Platform ini memiliki kompatibilitas perangkat keras yang luas, selain mendukung keluarga papan Arduino standar seperti Uno, Mega, dan Nano, IDE ini juga dapat dikonfigurasi untuk memprogram mikrokontroler pihak ketiga, termasuk seri ESP8266 dan ESP32[12].

### 2.5. Sensor

Sensor adalah komponen elektronik yang mampu mendeteksi, mengukur, dan mengawasi perubahan fisik dan kimia. Data dari perubahan yang terdeteksi akan diubah menjadi sinyal

yang terukur agar dapat diproses oleh manusia atau sistem elektronik, sehingga informasinya bisa dimanfaatkan. Sensor bisa merespons berbagai perubahan fisik di lingkungan, seperti suhu, cahaya, suara, atau gerakan[13].

### 2.6. Motor DC

Motor DC adalah motor listrik yang beroperasi dengan listrik arus searah (DC). Arah putaran motor ini ditentukan oleh arus dan tegangan, sedangkan kecepatannya dikendalikan oleh tegangan kumparan. Untuk mengubah arah putaran, digunakan H-bridge, sementara variabel resistor atau potensiometer berfungsi untuk mengatur kecepatannya[14].

### 2.7. Motor Driver L298N

Motor driver L298N merupakan komponen yang berfungsi untuk mengaktifkan motor DC dan mengatur arah perputarannya[4]. LM298 adalah komponen yang unggul dalam menangani arus dan tegangan tinggi. Ini mempermudah mikrokontroler, yang memiliki keluaran arus dan tegangan rendah, untuk mengendalikan motor penggerak yang memerlukan arus tinggi[15].

### 2.8. Firebase

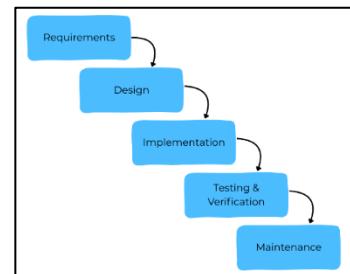
Firebase adalah platform pengembangan aplikasi (mobile dan web) milik Google yang menyediakan berbagai layanan backend (seperti database, autentikasi, dan hosting) yang memungkinkan developer untuk fokus pada pengembangan fitur aplikasi tanpa perlu mengelola server sendiri[16].

### 2.9. Android Studio

Android Studio merupakan perangkat lunak resmi dari Google untuk mengembangkan aplikasi Android. Bersifat sumber terbuka dan gratis, alat ini pertama kali diumumkan pada konferensi Google I/O tanggal 16 Mei 2013. Android Studio telah resmi menggantikan Eclipse sebagai platform utama untuk pengembangan aplikasi Android[17].

## 3. METODE PENELITIAN

Metode *waterfall* merupakan strategi pengembangan yang sistematis dan bertahap, mengikuti alur linier di mana satu fase harus selesai sebelum fase berikutnya dimulai, mirip dengan jatuhnya air terjun[8].



Gambar 1. Waterfall Method

Setiap aspek pengembangan dalam model ini saling berhubungan, meliputi pengumpulan dan analisis kebutuhan, perancangan sistem, proses pengkodean, fase pengujian dan verifikasi, serta tahapan implementasi dan pemeliharaan program[6].

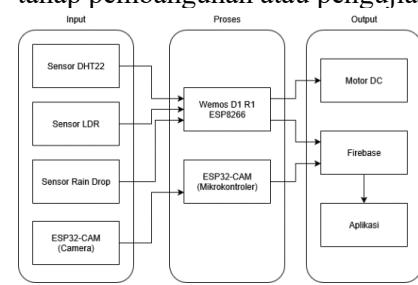
Berdasarkan gambar 1, metode *waterfall* membagi proses pengembangan menjadi tahapan-tahapan yang berurutan dan terstruktur. Tahapan-tahapan ini dibagi menjadi 5 bagian yaitu:

#### 1) Requirements

Tahap ini bertujuan untuk menganalisis kebutuhan apa saja yang diperlukan untuk penelitian ini. Mulai dari pemilihan *hardware*, *software* pengembang serta data cuaca yang akan diolah nantinya.

#### 2) Design

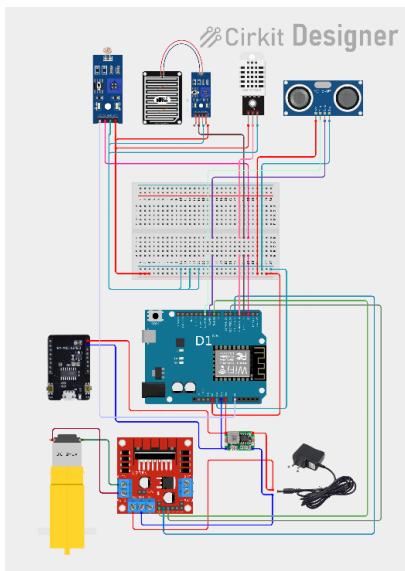
Tahap selanjutnya adalah perancangan sistem yang bertujuan untuk mendapatkan blueprint sistem yang nanti akan diimplementasi. Dengan demikian, dapat meminimalkan perubahan besar di tahap pembangunan atau pengujian.



Gambar 2. Diagram Blok Sistem

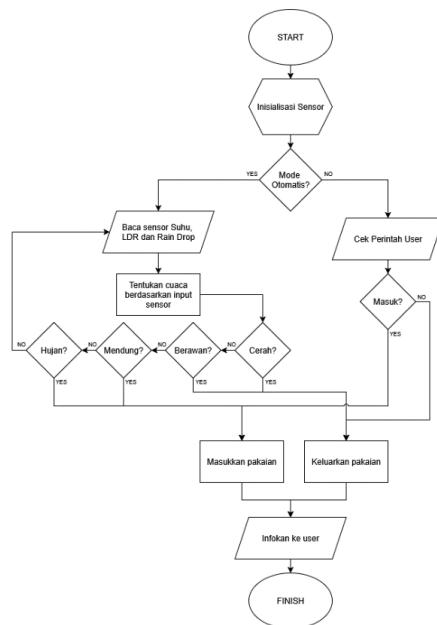
Blok diagram pada gambar menunjukkan sistem penjemur otomatis berbasis IoT yang terdiri dari sensor DHT22, LDR, dan Rain Drop sebagai input untuk membaca kondisi lingkungan, serta ESP32-CAM sebagai kamera pemantau. Sensor-sensor tersebut terhubung ke mikrokontroler Wemos D1 R1 (ESP8266) yang

memproses data dan mengontrol motor DC sebagai aktuator utama. Wemos juga mengirim data ke Firebase Realtime Database, sementara ESP32-CAM secara mandiri mengunggah gambar ke Firebase Storage dan menulis metadata ke Realtime Database. Seluruh data dan kontrol sistem diakses oleh pengguna melalui aplikasi Android yang terhubung ke Firebase.



Gambar 3. Rancangan Rangkaian Alat

Pada Gambar 3, ditunjukkan bahwa komponen utama sistem terdiri atas sensor LDR, sensor suhu, sensor rain drop, sensor ultrasonik HC-SR04, serta driver motor DC yang seluruhnya terhubung ke papan mikrokontroler Wemos D1 R1. Sistem memperoleh catu daya dari adaptor bertegangan 12 volt, yang selanjutnya diturunkan menjadi 5 volt menggunakan modul step down Mini360. Tegangan 5 volt tersebut kemudian dialirkan ke dua perangkat utama, yaitu Wemos D1 R1 sebagai unit kendali sistem, dan ESP32-CAM yang berfungsi sebagai perangkat pengambil gambar. Adapun suplai daya untuk motor DC tidak melalui modul step down, melainkan disambungkan langsung ke adaptor 12 volt. Hal ini dilakukan untuk memastikan motor mendapatkan arus yang memadai guna mendukung kinerja mekanis sistem penjemur otomatis secara optimal.

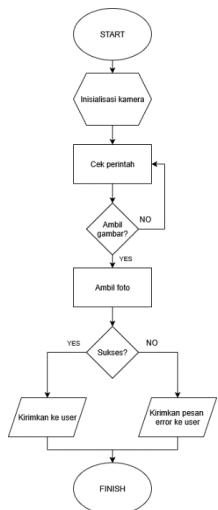


Gambar 4. Flowchart Sistem (Wemos D1 R1)

Flowchart pada Gambar 4 menggambarkan alur logika kerja yang diaplikasikan pada Wemos D1 R1. Sistem dimulai dengan proses inisialisasi seluruh sensor, meliputi sensor suhu, sensor intensitas cahaya (LDR), dan sensor rain drop. Setelah proses inisialisasi selesai, sistem akan memeriksa mode operasi yang digunakan, apakah dalam kondisi otomatis atau manual. Jika sistem berada pada mode otomatis, maka mikrokontroler akan membaca data dari ketiga sensor untuk kemudian menentukan kondisi cuaca berdasarkan parameter input yang diterima.

Proses penentuan cuaca dilakukan secara bertahap melalui pengujian kondisi, mulai dari apakah sedang hujan, mendung, berawan, atau cerah. Jika salah satu kondisi terpenuhi, maka sistem akan mengambil tindakan yang sesuai. Dalam kondisi hujan dan mendung, sistem akan mengaktifkan mekanisme untuk menarik jemuran masuk secara otomatis guna menghindari paparan air hujan. Sebaliknya, jika cuaca terdeteksi cerah dan berawan, sistem akan mendorong jemuran keluar. Setelah aksi dilakukan, sistem akan mengirimkan informasi status ke pengguna.

Apabila sistem tidak berada pada mode otomatis, maka kendali dialihkan sepenuhnya kepada pengguna. Sistem akan memeriksa perintah yang diberikan secara manual, baik untuk menarik jemuran masuk maupun mendorong keluar. Keputusan ini akan dieksekusi sesuai perintah yang diterima, lalu status aksi juga akan diinformasikan kepada pengguna.



Gambar 5. Flow Sistem (ESP32-CAM)

Flowchart pada Gambar 5 menjelaskan alur kerja sistem pengambilan gambar yang dijalankan oleh modul ESP32-CAM sebagai bagian dari sistem monitoring berbasis IoT. Proses dimulai dengan inisialisasi kamera oleh mikrokontroler untuk memastikan perangkat siap digunakan. Setelah inisialisasi selesai, sistem akan masuk ke tahap pemeriksaan perintah yang dikirimkan oleh pengguna melalui antarmuka aplikasi.

Jika perintah yang diterima adalah instruksi untuk mengambil gambar, sistem akan melanjutkan ke proses pengambilan foto menggunakan modul kamera. Namun apabila tidak ada perintah yang sesuai, sistem akan kembali ke proses pemeriksaan perintah hingga perintah yang diharapkan diterima. Setelah proses pengambilan gambar dilakukan, sistem akan melakukan validasi terhadap keberhasilan proses tersebut. Bila gambar berhasil diambil dengan baik,

maka hasil foto akan dikirimkan ke pengguna melalui saluran komunikasi yang telah ditentukan, seperti Firebase Storage atau aplikasi mobile. Sebaliknya, jika proses pengambilan gambar gagal, sistem akan mengirimkan notifikasi kesalahan kepada pengguna agar dapat dilakukan tindak lanjut.

### 3) Implementation

Tahap *implementation* adalah proses merealisasikan rancangan yang sudah ditentukan menjadi sistem yang nyata. Proses ini meliputi perangkaian alat, pengkodean mikrokontroler dan pengkodean aplikasi.

### 4) Testing & verification

Pada tahap *testing* akan dilakukan pengujian terhadap fungsionalitas dari fitur-fitur di aplikasi dengan menggunakan teknik blackbox, pengujian logika penentuan cuaca dan aksi oleh perangkat dan terakhir pengujian keandalan ESP32-CAM dalam mengambil gambar.

### 5) Maintenance

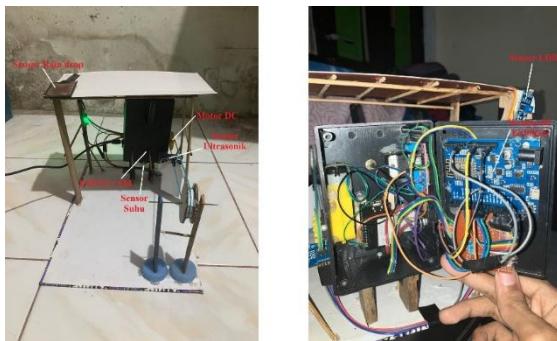
Fase terakhir adalah *maintenance*, di mana sistem yang sudah rampung akan terus disempurnakan dan dikembangkan. Tahap ini menjadi krusial karena kebutuhan akan fitur baru dapat muncul dan kerusakan perangkat keras (misalnya, penggantian sensor yang rusak) bisa terjadi seiring berjalanannya waktu. Dengan demikian, pemeliharaan memastikan sistem selalu adaptif dan berjalan optimal.

## 4. HASIL DAN PEMBAHASAN

### 4.1. Hasil implementasi

#### 4.1.1. Implementasi perangkat keras

Hasil implementasi perangkat keras ditunjukkan pada gambar 6, yang menampilkan prototipe sistem dalam bentuk miniatur. Terlihat struktur jemuran yang dilengkapi Motor DC sebagai penggerak pakaian, serta penempatan sensor rain drop, sensor suhu, sensor ultrasonik, dan ESP32-CAM yang diarahkan ke lintasan jemuran.



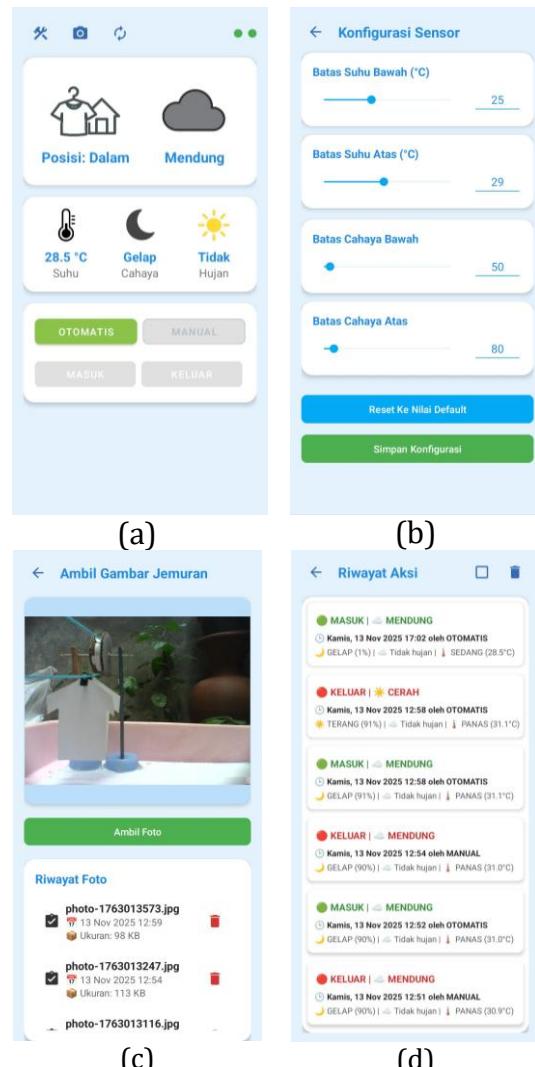
Gambar 6 Implementasi Rangkaian Alat

#### 4.1.2. Implementasi perangkat lunak

Tahap implementasi pengkodean berfokus pada unit kendali utama, Wemos D1 R1 (ESP8266), yang diprogram untuk bertindak sebagai otak sistem. Sesuai dengan alur pada gambar 4, program ini menerapkan logika ganda. Pada mode manual, sistem akan memeriksa perintah dari Firebase, sedangkan pada mode otomatis, sistem akan memanggil fungsi untuk mengambil keputusan berdasarkan hasil klasifikasi cuaca. Fungsi klasifikasi cuaca ini mengolah data dari sensor DHT22, LDR, dan Rain Drop, dengan nilai ambang batas yang diambil secara dinamis dari Firebase. Ambang batas sensor hanya diberlakukan untuk sensor DHT22 dan LDR karena output yang digunakan pada sensor-sensor tersebut adalah output analognya. Pada kondisi awal, batasan sensor suhu adalah 29 (batas suhu panas) dan 25 (batas suhu dingin). Sedangkan untuk sensor cahaya adalah 50 (batas cahaya terang) dan 80 (batas cahaya gelap). Pengaturan ambang batas ini dapat dikonfigurasi oleh pengguna melalui aplikasi mobile. Untuk kontrol aktuator, sensor ultrasonik difungsikan untuk mendeteksi posisi jemuran. Terakhir, Wemos D1 R1 secara aktif mengirimkan data status terbaru ke Firebase untuk monitoring pada aplikasi mobile dan mencatat riwayat aksi ke Firebase, yang juga ditampilkan pada aplikasi mobile.

Secara terpisah, unit pemantauan visual (ESP32-CAM) diprogram sebagai unit yang berdiri sendiri. Alur kerjanya mengikuti flowchart pada gambar 5, di mana perangkat akan secara periodik membaca perintah dari Firebase. Ketika pengguna menekan tombol "Ambil Foto" pada aplikasi mobile, perintah "AMBIL\_FOTO" akan terkirim. ESP32-CAM kemudian akan mengambil gambar, menyimpannya sementara di SD Card, dan mengunggahnya ke Firebase Storage. Jika

berhasil, metadata foto akan dicatat ke Realtime Database untuk ditampilkan di aplikasi mobile, jika gagal, akan muncul pesan error pada aplikasi. Pemisahan logika ini memastikan bahwa proses pengambilan dan pengunggahan foto tidak mengganggu fungsi kendali utama yang sedang berjalan di Wemos D1 R1.



Gambar 7 Tampilan Aplikasi Monitor dan Kontrol

Implementasi aplikasi pada gambar 7, terdiri dari (a) halaman utama yang akan menampilkan informasi kesiapan perangkat keras, posisi pakaian, hasil bacaan sensor-sensor, dan hasil klasifikasi cuaca. Pada halaman utama juga ditampilkan pilihan mode yang ingin digunakan, jika mode manual diaktifkan akan membuat tombol masuk dan keluar menyala. Lalu halaman (b) merupakan halaman konfigurasi batasan sensor suhu dan sensor cahaya yang dapat diubah oleh pengguna,

sehingga dapat disesuaikan dengan kondisi iklim di daerah pengguna. Kemudian halaman (c) untuk mengambil foto pada area jemuran, melihat riwayat foto dan disediakan fitur untuk menghapus riwayat foto. Terakhir halaman (d) menampilkan riwayat aksi dari alat dan informasi detail cuaca ketika aksi dilakukan.

## 4.2. Pengujian perangkat dan aplikasi

### 4.2.1. Pengujian aplikasi (blackbox)

Tabel 1. Pengujian Aplikasi (Blackbox)

Kondisi	Hasil yang diharapkan	Hasil yang didapatkan	Status
Input batasan sensor suhu yang salah	Gagal menyimpan konfigurasi dan muncul pesan error "Suhu harus antara 10-50C"	Konfigurasi gagal disimpan dan pesan error muncul	Sesuai
Input batasan sensor LDR yang salah	Gagal menyimpan konfigurasi dan muncul pesan error "Cahaya harus antara 0-1023"	Konfigurasi gagal disimpan dan pesan error muncul	Sesuai
Ambil foto ketika espcam32 belum siap	Gagal mengambil foto dan aplikasi menampilkan pesan "Perangkat kamera belum siap. Silakan tunggu..."	Foto gagal diambil dan pesan error muncul	Sesuai
Data pada realtime database dan aplikasi sync	Data sensor, cuaca, dan status jemuran tampil seragam antara Firebase dan aplikasi	Data sensor, cuaca, dan status jemuran tampil seragam antara Firebase dan aplikasi	Sesuai
Input konfigurasi batasan baru	Konfigurasi batas sensor berhasil diupdate dan user akan diarahkan ke tampilan utama	Konfigurasi batas sensor berhasil diupdate dan user akan diarahkan ke tampilan utama	Sesuai
Notifikasi muncul setelah ada aksi	Notifikasi muncul setelah perangkat melakukan aksi	Notifikasi muncul	Sesuai

Operasi manual (masuk/keluar)	Jemuran dikeluarkan atau dimasukkan sesuai dengan input dari user	Pakaian berhasil dikeluarkan atau dimasukkan sesuai dengan perintah dari user	Sesuai
Ambil foto menggunakan ESPCAM32	Foto terambil dan ditampilkan pada aplikasi	Foto berhasil diambil dan ditampilkan pada aplikasi	Sesuai
Hapus riwayat foto yang pernah diambil	Foto terhapus di Cloud dan Aplikasi	Foto terhapus di Cloud dan Aplikasi	Sesuai
Lihat riwayat aksi	Riwayat aksi tampil	Riwayat aksi tampil	Sesuai
Hapus riwayat aksi	Riwayat aksi terhapus di Cloud dan Aplikasi	Riwayat aksi terhapus di Cloud dan Aplikasi	Sesuai

Berdasarkan hasil pengujian yang telah dilakukan melalui skenario *positive* dan *negative testing*, seluruh fitur utama dalam aplikasi menunjukkan performa yang sesuai dengan ekspektasi. Fungsi validasi input berhasil mencegah konfigurasi yang tidak valid, dan sistem memberikan respons yang tepat dalam bentuk pesan kesalahan yang informatif. Fitur pengambilan dan penghapusan foto, sinkronisasi data sensor dan status jemuran dengan Firebase, serta kontrol manual dan notifikasi otomatis seluruhnya berjalan dengan baik tanpa kendala. Hasil ini menunjukkan bahwa aplikasi telah bekerja secara stabil, responsif, dan mampu mendukung interaksi pengguna terhadap sistem penjemur otomatis secara optimal.

### 4.2.2. Pengujian deteksi cuaca dan aksi perangkat

Tabel 2. Pengujian Deteksi Cuaca dan Aksi Perangkat

Nilai LDR	Nilai Suhu	Rain Drop	Cuaca	Durasi (Detik)	Aksi	Status
1024	29.9	1	Hujan	2.2	Masuk	Sesuai
6	30.1	1	Hujan	2.7	Masuk	Sesuai
31	35.8	0	Cerah	2.3	Keluar	Sesuai
31	28.7	0	Cerah	2	Keluar	Sesuai
74	30.2	0	Berawan	1.9	Keluar	Sesuai
63	28.5	0	Berawan	1.8	Keluar	Sesuai
57	24.8	0	Berawan	2.5	Keluar	Sesuai

1000	29.8	0	Mendung	2.4	Masuk	Sesuai
589	28.9	0	Mendung	2.2	Masuk	Sesuai
122	24.7	0	Mendung	2.7	Masuk	Sesuai

Tabel 1 menunjukkan hasil pengujian sistem penjemur otomatis berdasarkan kombinasi pembacaan sensor LDR, suhu (DHT22), dan sensor hujan (Rain Drop) untuk menentukan kondisi cuaca serta aksi yang diambil oleh sistem. Setiap aksi dilakukan dalam rentang waktu sekitar 1.8 hingga 2.7 detik, dengan rata-rata durasi eksekusi sebesar 2.3 detik. Hasil pengujian menunjukkan bahwa seluruh keputusan dan aksi yang diambil sistem telah sesuai dengan logika cuaca yang dirancang.

#### 4.2.3. Pengujian pengambilan foto ESP32-CAM

Tabel 3. Pengujian Pengambilan Foto ESP32-CAM

No	Durasi (detik)	Size File (bytes)	Status
1	6	71.742	Sukses
2	7	71.824	Sukses
3	7	71.974	Sukses
4	7	70.883	Sukses
5	6	71.068	Sukses
6	7	71.335	Sukses
7	7	71.861	Sukses
8	7	70.433	Sukses
9	6	70.329	Sukses
10	7	70.460	Sukses

Tabel 2 menunjukkan pengujian yang dilakukan untuk mengevaluasi kinerja ESP32-CAM dalam mengambil dan mengunggah foto ke Firebase Storage secara mandiri. Pengujian dilakukan dengan melakukan 10 kali percobaan pengambilan dan pengunggahan foto, dengan tingkat keberhasilan 100%. Durasi proses pada setiap percobaan berkisar antara 6 hingga 7 detik, dengan ukuran file foto yang dihasilkan berada dalam rentang  $\pm 70\text{--}72$  KB. Hasil ini menunjukkan bahwa ESP32-CAM mampu melakukan proses capture dan upload dalam waktu rata-rata sekitar 7 detik, yang masih tergolong responsif untuk kebutuhan pemantauan visual berkala dalam sistem penjemur otomatis.

## 5. KESIMPULAN

Berdasarkan hasil perancangan dan pengujian yang telah dilakukan, dapat ditarik beberapa kesimpulan sebagai berikut:

1. Prototipe sistem penjemur pakaian otomatis berbasis Internet of Things (IoT) terbukti dapat mendeteksi cuaca dan menentukan aksi yang tepat sesuai dengan kondisi cuaca dengan rata-rata waktu durasi eksekusi adalah 2.3 detik.
2. Integrasi modul ESP32-CAM sebagai fitur pemantauan visual terbukti andal berfungsi sebagai alat validasi jarak jauh yang efektif. Delay waktu 7 detik tergolong cukup baik untuk sistem serupa..
3. Aplikasi Android berfungsi dengan baik sebagai antarmuka pengguna, dengan seluruh fitur seperti konfigurasi sensor, kendali manual, sinkronisasi data real-time, pemantauan foto, dan pengelolaan riwayat dapat dioperasikan secara responsif. Validasi input batas sensor terbukti berhasil mencegah kesalahan dengan membatasi pengaturan suhu pada rentang 10-50°C dan cahaya 0-1023, serta sistem notifikasi berhasil memberikan umpan balik status secara real-time setiap kali aksi perangkat dieksekusi.
4. Saran pengembangan selanjutnya adalah sistem dapat dikembangkan dengan menambahkan sumber daya mandiri seperti panel surya dan baterai cadangan agar alat tetap dapat beroperasi saat terjadi pemadaman listrik dan penambahan metode *machine learning* pada ESP32-CAM agar dapat mengklasifikasi cuaca berdasarkan foto awan yang ada di sekitar jemuran.

## UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada pihak-pihak terkait yang telah memberi dukungan dan arahan terhadap penelitian ini.

## DAFTAR PUSTAKA

- [1] 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, Jul. 2024, doi: 10.47313/jig.v27i1.3710.
- [2] A. Setiawan, I. Istiadi, and G. Priyandoko, "Pengendali Dan Pemantau Arus Tegangan Pada Terminal Listrik Rumah Tangga Berbasis IoT," *JOINTECS (Journal Inf. Technol. Comput. Sci.)*, vol. 8, no. 1, p. 27, Jun. 2023,

- doi: 10.31328/jointecs.v8i1.4633.
- [3] M. Z. Fauzi, H. Susilawati, and S. Nurpadillah, "Rancang Bangun Alat Penjemur Pakaian Semi Otomatis Berbasis Arduino Uno," *Fuse-teknik Elektro*, vol. 4, no. 2, pp. 125–134, Dec. 2024, doi: 10.52434/jft.v4i2.41950.
- [4] 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, Jan. 2024, doi: 10.59562/mediatik.v6i1.1352.
- [5] M. Majid and R. Pramudita, "Perancangan Sistem Jemuran Pakaian Otomatis Berbasis Teknologi Internet of Things (IoT)," *J. Ilm. Teknol. Infomasi Terap.*, vol. 10, no. 3, pp. 111–118, Aug. 2024, doi: 10.33197/jitter.vol10.iss3.2024.2010.
- [6] T. Hidayat, M. Akbar, and M. Mursalim, "Perancangan Prototype Alat Penjemur Pakaian Otomatis Berbasis Mikrokontroler Atmega 328," *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 6, no. 3, pp. 372–377, Jul. 2024, doi: 10.47233/jteksis.v6i3.1333.
- [7] A. Sanaris and I. Suharjo, "Prototype Alat Kendali Otomatis Penjemur Pakaian Menggunakan NodeMCU ESP32 Dan Telegram Bot Berbasis Internet of Things (IOT )," *J. Prodi Sist. Inf.*, no. 84, pp. 17–24, 2020.
- [8] S. Hidayatulloh and J. Aryanto, "Sistem Pengendalian Jemuran Otomatis berbasis IoT dengan Logika Fuzzy untuk Pengkondision Cuaca," *Edumatic J. Pendidik. Inform.*, vol. 7, no. 2, pp. 287–296, Dec. 2023, doi: 10.29408/edumatic.v7i2.21515.
- [9] Y. I. Chandra, I. Irfan, D. Gustina, S. W. Purtiningrum, and N. Yuliani, "Real-Time Prototype Electricity Monitoring and Forecasting System based on Wemos D1 R1 ESP8266 and IoT," *Logist. Oper. Manag. Res.*, vol. 2, no. 2, pp. 1–13, 2023, doi: 10.31098/lomr.v2i2.1551.
- [10] S. A. Arrahma and R. Mukhaiyar, "Pengujian Esp32-Cam Berbasis Mikrokontroler ESP32," *JTEIN J. Tek. Elektro Indones.*, vol. 4, no. 1, pp. 60–66, Feb. 2023, doi: 10.24036/jtein.v4i1.347.
- [11] D. Dasril, H. Indou, and R. Suppa, "PROTOTYPE ALAT PENDETEKSI BANJIR MENGGUNAKAN ARDUINO BERBASIS IOT," *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 3, Aug. 2024, doi: 10.23960/jitet.v12i3.5135.
- [12] H. Darmanto, L. Lamsadi, and H. Asrul, "Monitoring Ketinggian Air Tandon Berbasis IoT Dengan ESP32 Melalui Website," *JUSTER J. Sains dan Terap.*, vol. 4, no. 2, pp. 67–73, May 2025, doi: 10.57218/juster.v4i2.1507.
- [13] R. N. Suharto, "Kajian Penggunaan Sensor dalam Sistem Pengujian dan Pemantauan Kualitas Air Minum Layak Konsumsi," *Polyg. J. Ilmu Komput. dan Ilmu Pengetah. Alam*, vol. 3, no. 1, pp. 59–68, Jan. 2025, doi: 10.62383/polygon.v3i1.396.
- [14] A. Habibi, I. M. B. Suksmadana, and B. Darmawan, "Perbandingan Efektivitas Pengendalian Robot Dengan Penggunaan Pid dan Tanpa Pid Pada Aplikasi Jarak Tertentu," *J. Inform. Teknol. dan Sains*, vol. 6, no. 4, pp. 813–819, Nov. 2024, doi: 10.51401/jinteks.v6i4.4957.
- [15] M. Ikromullah, I. Hafizul Haq, A. Satria Putra, B. Purwahyudi, and R. Watiasih, "Rancang Bangun Sistem Jemuran Pakaian Otomatis Berbasis Arduino," *INTER TECH*, vol. 2, no. 2, pp. 83–92, Nov. 2024, doi: 10.54732/i.v2i2.1092.
- [16] H. Kurnia AR, "Implementasi IoT Pada Sistem Monitoring Suhu dan Kelembaban Menggunakan ESP32, Firebase dan Kodular," *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 9, no. 1, pp. 1781–1787, Jan. 2025, doi: 10.36040/jati.v9i1.12874.
- [17] N. Salshavira, "Perancangan Aplikasi Pengelolaan Keuangan dengan Menggunakan Android Studio," *J. Informatics Adv. Comput.*, vol. 5, no. 2, pp. 35–42, Nov. 2024, doi: 10.35814/jiac.v5i2.8078.