

DASHBOARD MONITORING DAN KONTROL MESIN STAMPING BERBASIS IOT DENGAN *THRESHOLD* KUALITAS DAN FIREBASE REALTIME DATABASE

Intan Adinda Losvita Ningrum^{1*}, Tatyantoro Andrasto², Arief Arfriandi³

^{1,2,3}Program Studi Teknik Elektro, Universitas Negeri Semarang; Sekaran, Kec. Gunungpati, Kota Semarang, Jawa Tengah 50229; (024) 86008700

Keywords:

ESP32-S3; Firebase Realtime Database; *Internet of Things*; *Stamping Machine*; *Web Dashboard*

Correspondent

Email: intanlosvita@students.unnes.ac.id

Email:

Abstrak. Monitoring kualitas dan kontrol mesin stamping di industri manufaktur umumnya masih dilakukan secara lokal dan manual, sehingga membatasi aksesibilitas data dan responsivitas pengambilan keputusan. Web dashboard online berbasis Firebase Realtime Database dikembangkan untuk mengatasi permasalahan tersebut, diintegrasikan dengan ESP32-S3 sebagai IoT gateway dan inspeksi kualitas YOLOv8 pada Raspberry Pi. Dashboard dibangun menggunakan HTML, CSS, dan JavaScript dengan fitur monitoring *real-time*, sistem *threshold* tiga parameter (*Warning*, *Critical*, *Minimum Sample*), *Role-Based Access Control* tiga tingkat, riwayat produksi harian, dan pelaporan otomatis, serta di-deploy secara global melalui Netlify. Firebase RTDB berfungsi sebagai *middleware* pertukaran data tanpa memerlukan server *backend* khusus. Pengujian fungsional pada 55 item uji dalam tujuh kategori menunjukkan tingkat keberhasilan 100%, mencakup akurasi sinkronisasi data, efektivitas *threshold Minimum Sample* dalam mencegah *false positive alarm*, dan keberhasilan eksekusi perintah relay. Sistem ini membuktikan bahwa integrasi Firebase dan web dashboard daring mampu mewujudkan monitoring dan kontrol mesin industri yang aksesibel secara global dengan infrastruktur minimal.



Copyright © 2026 The Author(s), Published by Jurnal Informatika dan Teknik Elektro Terapan. This is an open-access article distributed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

Abstract. *Quality monitoring and control of stamping machines in manufacturing industries are commonly performed locally and manually, limiting data accessibility and decision-making responsiveness. An online web dashboard based on Firebase Realtime Database was developed to address this issue, integrated with an ESP32-S3 as an IoT gateway and YOLOv8-based quality inspection on a Raspberry Pi. The dashboard was built using HTML, CSS, and JavaScript, featuring real-time monitoring, a three-parameter threshold system (Warning, Critical, Minimum Sample), three-tier Role-Based Access Control, daily production history, and automated reporting, deployed globally via Netlify. Firebase RTDB serves as the data exchange middleware without requiring a dedicated backend server. Functional testing on 55 test items across seven categories yielded a 100% success rate, covering data synchronization accuracy, Minimum Sample threshold effectiveness in preventing false positive alarms, and relay command execution. These results demonstrate that integrating Firebase and an online web dashboard can enable globally accessible industrial machine monitoring and control with minimal infrastructure requirements.*

1. PENDAHULUAN

Perkembangan industri manufaktur mendorong penerapan sistem digital untuk

meningkatkan efisiensi proses produksi, menjaga konsistensi kualitas, serta mendukung pengambilan keputusan berbasis data [1]. Pada industri pengemasan, proses stamping box digunakan untuk memberikan identitas produk, informasi produksi, maupun penanda kualitas. Kualitas hasil stamping perlu dipantau secara berkelanjutan karena berpengaruh terhadap keterbacaan informasi dan kesesuaian produk dengan standar yang ditetapkan [2]. Namun, proses monitoring produksi yang masih dilakukan secara manual berpotensi menyebabkan keterlambatan identifikasi masalah, kesalahan pencatatan data, dan kurang optimalnya evaluasi kualitas produksi.

Sistem kontrol industri umumnya menggunakan *Programmable Logic Controller* (PLC) karena memiliki keandalan yang tinggi dalam mengendalikan proses dan aktuator industri [3]. Meskipun demikian, PLC memiliki keterbatasan dalam penyajian data monitoring berbasis jaringan, penyimpanan histori produksi, serta integrasi dengan sistem pemantauan yang dapat diakses secara fleksibel oleh pengguna. Kondisi tersebut mendorong kebutuhan akan sistem monitoring tambahan yang mampu mengelola dan menyajikan data produksi secara terpusat tanpa mengganggu fungsi kontrol utama mesin.

Perkembangan *Internet of Things* (IoT) memungkinkan perangkat industri terhubung dengan jaringan internet sehingga data operasional dapat dikirim, disimpan, dan diakses secara *real-time* [4]. Pemanfaatan ESP32-S3 sebagai perangkat IoT memberikan kemudahan integrasi dengan layanan cloud karena mendukung komunikasi Wi-Fi dan memiliki sumber daya yang memadai untuk aplikasi monitoring. Selain itu, Firebase Realtime Database menyediakan mekanisme sinkronisasi data secara langsung antara perangkat dan aplikasi pengguna sehingga banyak digunakan sebagai media pertukaran data pada sistem IoT modern [5].

Dashboard web menjadi salah satu solusi yang banyak digunakan untuk menyajikan informasi produksi secara terpusat karena mampu menampilkan data monitoring, histori produksi, laporan operasional, dan indikator kinerja dalam satu antarmuka yang mudah diakses [6]. Integrasi dashboard web dengan teknologi IoT juga memungkinkan proses

pemantauan dilakukan secara *real-time* tanpa bergantung pada lokasi pengguna [7].

Berbagai penelitian telah menunjukkan bahwa integrasi ESP32 dan Firebase Realtime Database mampu mendukung monitoring dan kontrol berbasis IoT pada berbagai bidang aplikasi [8]. Selain mendukung pertukaran data secara *real-time*, pendekatan tersebut memungkinkan penyimpanan histori data dan pengendalian perangkat melalui jaringan internet [9]. Namun, penelitian yang secara khusus membahas monitoring hasil produksi stamping box berbasis dashboard web masih relatif terbatas. Sebagian besar penelitian berfokus pada objek monitoring yang berbeda dan belum mengintegrasikan fitur pemantauan kualitas produksi, histori produksi, laporan otomatis, grafik *cycle time*, pembagian hak akses pengguna, pengaturan *threshold* kualitas, serta pengiriman *command* kontrol dalam satu platform monitoring terpadu.

Berdasarkan kondisi tersebut, penelitian ini mengimplementasikan sistem monitoring dan kontrol mesin stamping berbasis IoT menggunakan ESP32-S3, Firebase Realtime Database, dan dashboard web. Sistem dirancang untuk menampilkan data produksi secara *real-time* atau mendekati *real-time* yang meliputi status sistem, status mesin, jumlah produk Good dan Not Good, total produksi, persentase kualitas, *runtime*, *downtime*, serta histori produksi. Selain itu, sistem dilengkapi fitur pengaturan *threshold* kualitas, laporan otomatis, grafik *cycle time*, pembagian hak akses pengguna, dan pengiriman *command* kontrol melalui Firebase Realtime Database. Kontribusi penelitian ini terletak pada integrasi berbagai fitur monitoring dan kontrol tersebut dalam satu platform berbasis web yang difokuskan pada proses produksi stamping box.

2. TINJAUAN PUSTAKA

2.1 Sistem Monitoring Berbasis IoT pada Industri Manufaktur

Sistem monitoring berbasis web online merupakan sistem yang menggunakan halaman web sebagai media utama penyajian data dan dapat diakses melalui *browser*. Berbeda dengan sistem lokal yang terbatas jaringan, sistem berbasis web memberikan kemudahan akses selama tersedia koneksi internet. Integrasi *real-*

time monitoring berbasis web terbukti meningkatkan efisiensi produksi melalui penyajian data langsung dan penyimpanan untuk analisis [1].

Pada lingkungan industri, dashboard web berperan penting dalam menyajikan data operasional jumlah produksi, status mesin, indikator kualitas dalam satu tampilan terpusat [6]. Sistem monitoring berbasis IoT dan web dashboard mampu mendukung pemantauan mesin produksi karena data dapat diakses lebih cepat dan terstruktur [2]. Wang et al. menunjukkan bahwa IoT *real-time* production monitoring efektif meningkatkan efisiensi proses dalam konteks *smart manufacturing* [10].

2.2 Firebase Realtime Database dalam Arsitektur IoT

Firebase Realtime Database digunakan sebagai media penyimpanan dan komunikasi data terpusat pada sistem. Basis data ini menyimpan data dalam format JSON dan mendukung sinkronisasi *real-time* antara perangkat dan aplikasi yang terhubung. Pada sistem ini, Firebase memungkinkan data produksi dari perangkat ditampilkan pada web dashboard online, sementara perintah kontrol dari dashboard dapat diterima oleh ESP32-S3. Dengan demikian, Firebase berfungsi sebagai penghubung utama untuk monitoring data, pencatatan riwayat, pembuatan laporan, dan kontrol relay tanpa memerlukan infrastruktur server terpisah.

2.3 Role-Based Access Control dan Sistem Ambang Kualitas

Role-Based Access Control (RBAC) adalah model pengendalian akses yang membatasi hak pengguna berdasarkan peran yang dimiliki. RBAC pada sistem informasi berbasis web efektif mengatur akses fitur sesuai kewenangan, mengurangi risiko penggunaan fitur kritis yang tidak sah [11]. Mekanisme *threshold* dalam sistem monitoring IoT digunakan untuk menilai kondisi data yang diterima dari perangkat secara otomatis. Witczak dan Szymoniak menunjukkan bahwa penerapan *threshold* memungkinkan dukungan pengambilan keputusan berbasis data yang terdeteksi secara *real-time* [12]. Digitalisasi laporan produksi harian juga terbukti meningkatkan efisiensi,

akurasi data, dan kemudahan akses informasi untuk mendukung pengambilan keputusan [13].

2.4 ESP32-S3 sebagai Perangkat IoT

ESP32-S3 merupakan mikrokontroler yang dirancang untuk aplikasi Internet of Things dan mendukung konektivitas Wi-Fi serta Bluetooth Low Energy. Kemampuan pemrosesan yang dimiliki memungkinkan ESP32-S3 digunakan untuk melakukan komunikasi data dengan layanan *cloud*, menerima *command* kontrol, serta mendukung integrasi dengan berbagai sistem monitoring berbasis internet. Berbagai penelitian menunjukkan bahwa platform ESP32 mampu digunakan sebagai perangkat akuisisi data dan komunikasi pada sistem monitoring berbasis IoT karena mendukung pertukaran data secara *real-time* dengan aplikasi pengguna [14] [15]. Pada penelitian ini, ESP32-S3 berperan sebagai perangkat IoT yang berkomunikasi dengan Firebase Realtime Database untuk mengirim dan menerima data yang berkaitan dengan proses monitoring dan kontrol mesin stamping.

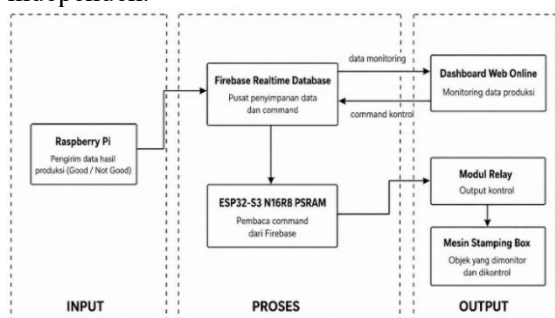
3. METODE PENELITIAN

Penelitian ini menggunakan pendekatan rekayasa (*engineering approach*) dengan metode *Research and Development* (R&D). Model R&D yang digunakan adalah model 4D (*Four-D*) yang mencakup tahapan *Define*, *Design*, *Develop*, dan *Disseminate* sebagai model dasar pengembangan produk [16]. Dalam konteks penelitian rekayasa prototipe teknis ini, tahap *Disseminate* dimodifikasi menjadi *Test and Evaluation*, karena fokus penelitian sampai pada pengujian dan evaluasi kinerja sistem [17]. Modifikasi ini konsisten dengan literatur R&D yang menyatakan pemilihan model harus disesuaikan dengan konteks dan tujuan penelitian [18].

3.1 Arsitektur Sistem

Sistem dirancang sebagai platform monitoring dan kontrol mesin stamping berbasis IoT yang mengintegrasikan inspeksi kualitas, komunikasi data *real-time*, dashboard daring, dan kontrol relay dalam satu alur kerja. Raspberry Pi mengirimkan hasil inspeksi YOLOv8 ke Firebase melalui node *detection_inbox*. ESP32-S3 N16R8 PSRAM

membaca data tersebut, memproses data produksi, memperbarui node monitoring, menghasilkan riwayat dan laporan harian, serta mengeksekusi perintah relay. Web dashboard membaca data dari Firebase dan mengirimkan perintah kontrol melalui REST API. Firebase RTDB berfungsi sebagai pusat pertukaran data antara ketiga subsistem sehingga aktivitas monitoring tidak mengganggu operasi kontrol utama mesin yang dikelola PLC secara independen.



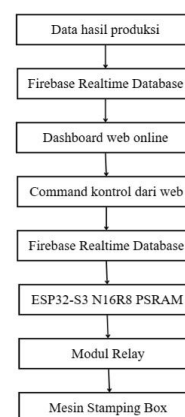
Gambar 1. Diagram Blok Sistem

Gambar 1. Menunjukkan diagram blok sistem dari input, proses, dan output sistem.

3.2 Alur Data dan Perintah Sistem

Alur data dan perintah sistem menggambarkan urutan pertukaran informasi antara sumber hasil produksi, Firebase Realtime Database, dashboard daring, ESP32-S3, modul relay, dan mesin stamping. Alur dimulai ketika data hasil produksi dihasilkan dari proses inspeksi. Data tersebut disimpan dalam Firebase Realtime Database dan kemudian dibaca oleh web dashboard daring untuk keperluan monitoring. Melalui dashboard, pengguna dapat mengamati kondisi produksi dan mengirimkan perintah kontrol saat diperlukan.

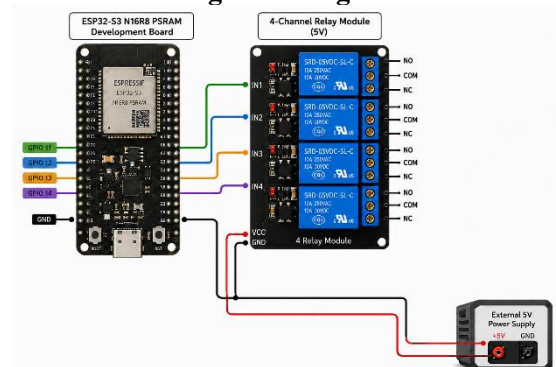
Ketika perintah dikirim dari dashboard, nilai perintah ditulis kembali ke Firebase Realtime Database. ESP32-S3 N16R8 PSRAM secara terus-menerus membaca data perintah dari Firebase. Jika perintah valid terdeteksi, ESP32-S3 memproses perintah tersebut dan mengaktifkan modul relay yang sesuai. Modul relay kemudian berfungsi sebagai keluaran kontrol yang terhubung ke mesin stamping box. Alur ini memastikan bahwa data monitoring dan perintah kontrol dipertukarkan melalui platform basis data *real-time* yang sama.



Gambar 2. Alur Data dan Perintah

Gambar 2. menunjukkan alur data dan perintah pada sistem monitoring dan kontrol mesin stamping berbasis IoT.

3.3 Perancangan Perangkat Keras



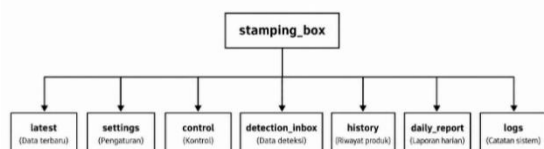
Gambar 3. Wiring ESP32-S3 dan Relay

Gambar 3. Menunjukkan wiring untuk ESP32-S3 dan modul relay. Empat modul relay dihubungkan ke GPIO ESP32-S3: GPIO 11 (START), GPIO 12 (STOP), GPIO 13 (MASTER ON), dan GPIO 14 (EMERGENCY). Durasi *pulse* relay diatur 1500 ms untuk perintah START, STOP, dan MASTER ON, sedangkan perintah Cylinder 1–3 bekerja dalam mode ON/OFF persisten.

3.4 Perancangan Firebase Realtime Database

Firebase RTDB distrukturkan pada node utama *stamping_box* dengan sub-node: *latest* (data produksi *real-time* untuk dashboard), *settings* (parameter *threshold* dan minimum sample), *control* (perintah dari dashboard), *detection_inbox* (data inspeksi masuk dari Raspberry Pi), *history* (riwayat produksi per entri), *daily_report* (ringkasan laporan harian),

logs (catatan aktivitas sistem). Struktur node Firebase ditunjukkan pada Gambar 4.

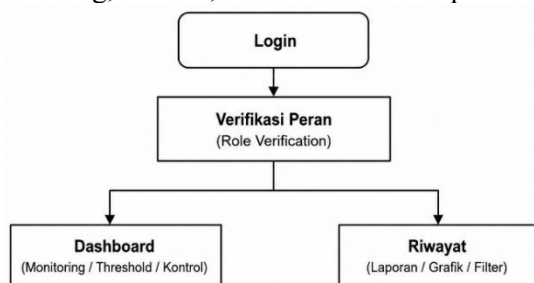


Gambar 4. Struktur Node Firebase RTDB

3.5 Perancangan Dashboard Web

Dashboard dikembangkan menggunakan HTML, CSS, dan JavaScript, terdiri atas tiga halaman: (1) Login untuk autentikasi dan pengalihan berbasis peran; (2) Dashboard untuk monitoring *real-time*, konfigurasi *threshold*, tombol kontrol mesin, mode manual, log produksi terbaru, dan grafik *cycle time*; (3) Histori untuk ringkasan harian, tabel riwayat, laporan otomatis, grafik *cycle time analysis*, dan filter jam. Dashboard di-deploy melalui Netlify untuk akses global melalui *browser*.

Sistem menerapkan RBAC tiga tingkat: Admin (akses penuh termasuk konfigurasi *threshold* dan kontrol mesin), Operator (kontrol mesin tanpa modifikasi *threshold*), dan Viewer (monitoring dan histori hanya-baca). *Threshold* kualitas dikonfigurasi melalui tiga parameter: Warning, Critical, dan Minimum Sample.



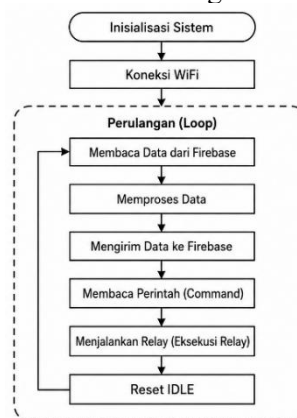
Gambar 5. Workflow Dashboard Web

Gambar 5. Menunjukkan *workflow* web dashboard. Setelah *login* berhasil, sistem mengarahkan pengguna ke tampilan halaman dashboard dan halaman histori yang sesuai berdasarkan peran.

3.6 Pemrograman ESP32-S3

ESP32-S3 diprogram menggunakan Arduino IDE dalam tiga fase siklus: (1) membaca node *detection_inbox* untuk memproses data inspeksi Good/Not Good yang masuk; (2) menghitung *runtime/downtime* dan memperbarui node *latest* dan *daily_report* secara periodik; (3) membaca node control dan

mengeksekusi aksi relay sesuai perintah. Setiap perintah direset ke IDLE setelah eksekusi untuk mencegah eksekusi berulang.



Gambar 6 Diagram Alir Program ESP32-S3

Gambar 6. menunjukkan diagram alir program ESP32-S3. Program berjalan dalam loop siklik yang membaca data produksi masuk, memperbarui data Firebase, dan mengeksekusi perintah kontrol relay.

3.7 Pengumpulan dan Analisis Data

Data dikumpulkan melalui observasi sistem, dokumentasi tangkapan layar dashboard, struktur node Firebase, dan output Serial Monitor ESP32-S3, serta pengujian fungsional terstruktur pada tujuh kategori: login dan kontrol akses, tampilan data monitoring, sistem *threshold* kualitas, tombol kontrol dan mode manual, data ringkasan riwayat harian, fitur halaman histori, dan eksekusi perintah ESP32-S3. Analisis dilakukan secara deskriptif kuantitatif menggunakan tiga persamaan evaluasi:

$$\text{Kesesuaian Data (\%)} = \frac{\text{Data Sesuai}}{\text{Total Data yang Diuji}} \times 100\% \quad (1)$$

$$\begin{aligned} \text{Tingkat Keberhasilan Command (\%)} \\ = \frac{\text{Command Berhasil}}{\text{Total Command yang Diuji}} \times 100\% \end{aligned} \quad (2)$$

$$\text{Persentase NG (\%)} = \frac{\text{Jumlah Not Good}}{\text{Total Produksi}} \times 100\% \quad (3)$$

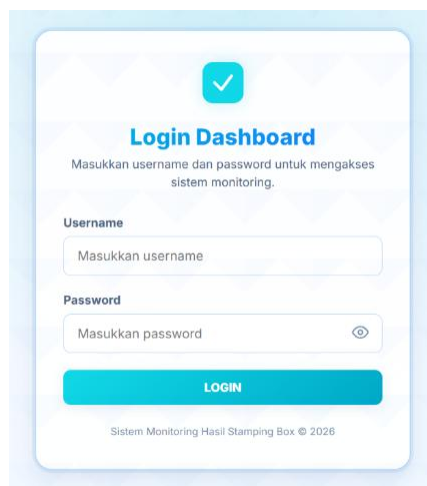
4. HASIL DAN PEMBAHASAN

4.1 Hasil Tampilan Dashboard Web

Implementasi dashboard web menghasilkan antarmuka yang mendukung proses monitoring dan kontrol mesin stamping secara online.

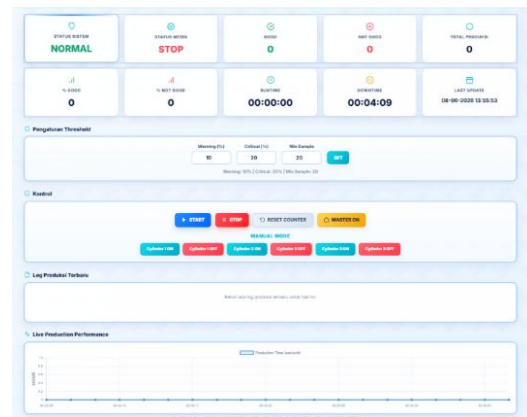
Dashboard terdiri atas halaman login, dashboard utama, dan histori produksi yang digunakan untuk menampilkan data monitoring, kontrol sistem, serta riwayat produksi. Tampilan antarmuka yang dikembangkan ditunjukkan pada Gambar 7. hingga Gambar 9.

Halaman login berfungsi sebagai gerbang autentikasi pengguna sebelum mengakses sistem. Pengguna diwajibkan memasukkan username dan password yang valid, kemudian sistem akan mengarahkan pengguna ke halaman sesuai dengan peran yang dimiliki. Tampilan halaman login ditunjukkan pada Gambar 7.



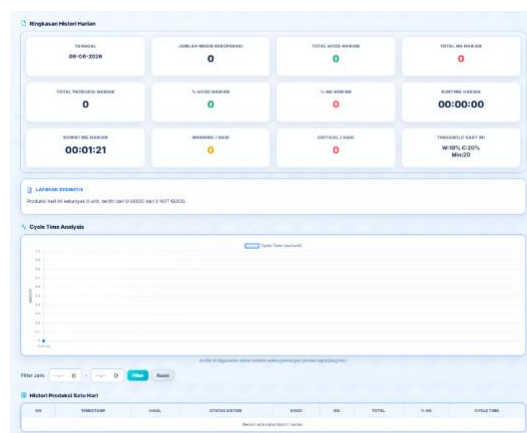
Gambar 7. Tampilan Halaman Login

Setelah proses autentikasi berhasil, pengguna diarahkan ke halaman dashboard utama yang menampilkan informasi monitoring secara *real-time*. Informasi yang disajikan meliputi status sistem, status mesin, jumlah produk Good, jumlah produk Not Good, total produksi, persentase kualitas, *runtime*, *downtime*, grafik *cycle time*, serta fitur kontrol mesin sesuai hak akses pengguna. Dashboard juga menyediakan fitur pengaturan *threshold* kualitas untuk membantu pengguna dalam memantau kondisi produksi secara lebih efektif. Tampilan dashboard utama ditunjukkan pada Gambar 8.



Gambar 8. Tampilan Halaman Dashboard

Selain dashboard utama, sistem juga menyediakan halaman histori yang digunakan untuk menampilkan riwayat produksi yang tersimpan pada Firebase Realtime Database. Halaman ini menampilkan data historis produksi, ringkasan laporan harian, serta grafik analisis *cycle time* yang dapat digunakan untuk mendukung proses evaluasi dan pelacakan data produksi. Tampilan halaman histori ditunjukkan pada Gambar 9.



Gambar 9. Tampilan Halaman Histori

4.2 Login dan Role-Based Access Control (RBAC)

Pengujian login dan kontrol akses dilakukan terhadap 5 skenario yang mencakup seluruh kombinasi peran pengguna dan kondisi akses abnormal. Hasil pengujian disajikan pada Tabel 1.

Tabel 1. Hasil Pengujian Login dan Kontrol Akses Pengguna

Peran /Skenario	Hasil yang Diharapkan	Hasil Pengamatan	Ket.
Admin	Akses penuh fitur sistem	Pengguna dapat mengakses dashboard, histori, pengaturan <i>threshold</i> , kontrol, mode manual, laporan, dan grafik	Sesuai
Operator	Akses kontrol tanpa modifikasi <i>threshold</i>	Pengguna dapat mengakses kontrol dan mode manual, tetapi pengaturan <i>threshold</i> tidak tersedia	Sesuai
Viewer	Akses monitoring dan histori hanya-baca	Pengguna dapat mengakses data, histori, laporan, dan jumlah produksi, tetapi tidak dapat mengakses fitur kontrol dan <i>threshold</i>	Sesuai
Login salah	Sistem menolak akses dan menampilkan pesan kesalahan	Sistem menampilkan pesan username atau password salah	Sesuai
Akses tanpa login	Sistem mengarahkan ke halaman login	Halaman dashboard/histori tidak dapat diakses sebelum login	Sesuai

Seluruh skenario menunjukkan perilaku yang sesuai dengan spesifikasi, menghasilkan tingkat keberhasilan 100%. Sistem RBAC tiga tingkat (Admin, Operator, Viewer) berhasil memisahkan hak akses secara tepat: Admin mendapat akses penuh termasuk konfigurasi *threshold* dan kontrol mesin; Operator dapat mengoperasikan kontrol mesin tanpa kemampuan mengubah parameter *threshold*; dan Viewer hanya dapat membaca data monitoring dan histori tanpa akses ke fungsi kontrol maupun konfigurasi.

Dari perspektif keselamatan operasional industri, stratifikasi akses ini memiliki relevansi praktis yang tinggi. Operator dapat menjalankan mesin tanpa risiko mengubah parameter kualitas secara tidak sengaja, sedangkan manajemen atau pengamat produksi dapat mengakses informasi *real-time* tanpa memicu aksi kontrol yang tidak diinginkan. Hal ini membedakan sistem yang diusulkan dari platform monitoring IoT sederhana yang umumnya tidak memiliki mekanisme otorisasi pengguna bertingkat.

4.3 Tampilan Data Monitoring Dashboard

Akurasi sinkronisasi data antara Firebase RTDB dan dashboard diuji dengan membandingkan 10 parameter data produksi secara berpasangan. Hasil pengujian ditunjukkan pada Tabel 2.

Tabel 2. Hasil Pengujian Tampilan Data Monitoring Dashboard

Parameter	Data Firebase	Data Dashboard	Ket.
Status Sistem	NORMAL	NORMAL	Sesuai
Status Mesin	RUN	RUN	Sesuai
Jumlah Good	14	14	Sesuai
Jumlah Not Good	2	2	Sesuai
Total produksi	16	16	Sesuai
Persentase Good	87.5%	87.5%	Sesuai

Persentase Not Good	12.5%	12.5%	Sesuai
Runtime	00:12:30	00:12:30	Sesuai
Downtime	00:03:15	00:03:15	Sesuai
Pembaruan terakhir	21/05/2026 11:12:55	21/05/2026 11:12:55	Sesuai

Akurasi data monitoring dihitung menggunakan Persamaan (1): Akurasi Data (%) = $(10/10) \times 100\% = 100\%$. Seluruh 10 parameter, mulai dari status sistem hingga *timestamp* pembaruan terakhir, menunjukkan nilai yang identik antara node latest Firebase dan tampilan dashboard. Temuan ini mengonfirmasi bahwa proses pembacaan data dari Firebase ke dashboard berjalan dengan benar dan sinkronisasi terjadi secara *real-time* sesuai mekanisme Firebase RTDB listener.

Integritas data 100% sepanjang alur ESP32-S3 → Firebase → Dashboard menunjukkan bahwa struktur node JSON yang dirancang secara modular mampu menjaga konsistensi data tanpa terjadi kerusakan atau kehilangan selama transmisi. Desain node yang memisahkan data *real-time* (*latest*), histori (*history*), dan perintah (*control*) secara efektif meminimalkan konflik penulisan data dan menyederhanakan arsitektur alur informasi.

4.4 Sistem Threshold dan Peringatan Kualitas

Pengujian sistem *threshold* kualitas dilakukan dengan konfigurasi Warning = 10%, Critical = 20%, dan *Minimum Sample* = 20 unit pada empat skenario yang dirancang untuk memverifikasi tiga status sistem: NORMAL, WARNING, dan CRITICAL. Persentase Not Good dihitung menggunakan Persamaan (3). Hasil pengujian ditampilkan pada Tabel 3.

Tabel 3. Hasil Pengujian *Threshold* dan Peringatan Kualitas

Total Prod.	NG	NG (%)	Status Harapan	Status Dashboard	Ket.
18	5	27.8	NORMAL	NORMAL	Sesuai
20	1	5	NORMAL	NORMAL	Sesuai
20	3	15	WARNING	WARNING	Sesuai
29	9	3	CRITICAL	CRITICAL	Sesuai

Tingkat keberhasilan *threshold*: $(4/4) \times 100\% = 100\%$. Temuan yang paling signifikan adalah skenario 1: meskipun persentase NG mencapai 27,8% (melampaui batas Critical 20%), sistem secara tepat mempertahankan status NORMAL karena total produksi (18 unit) belum mencapai *Minimum Sample* (20 unit). Mekanisme *Minimum Sample* terbukti efektif mencegah false positive alarm pada fase awal produksi.

Relevansi praktis dari desain ini sangat tinggi dalam konteks industri nyata: unit produksi awal pada mesin stamping sering menunjukkan kualitas lebih rendah akibat proses pemanasan (*warm-up*) atau kalibrasi awal mesin. Tanpa parameter *Minimum Sample*, sistem akan menghasilkan alarm palsu yang berulang pada awal setiap sesi produksi, sehingga berpotensi menurunkan kepercayaan operator terhadap sistem monitoring. Witczak dan Szymoniak menegaskan pentingnya mekanisme *threshold* yang dirancang dengan tepat untuk mendukung pengambilan keputusan berbasis data yang bermakna sebuah prinsip yang diimplementasikan secara efektif dalam sistem ini [12]. Sistem *threshold* tiga parameter yang dapat dikonfigurasi ini juga memberikan fleksibilitas adaptasi terhadap berbagai jenis produk dan standar kualitas yang berbeda.

4.5 Tombol Kontrol dan Mode Manual

perintah yang terdiri atas empat perintah operasi utama (START, STOP, RESET COUNTER, MASTER ON) dan enam perintah mode manual (Cylinder 1–3 ON/OFF). Setiap perintah diverifikasi melalui pengamatan nilai node *control* Firebase dan output Serial Monitor ESP32-S3. Hasil pengujian disajikan pada Tabel 4.

Tabel 4. Hasil Pengujian Tombol Kontrol dan Mode Manual

Tombol Dashboard	Perintah Firebase	Keluaran Pengamatan	Ket.
Start	START	Relay START aktif	Sesuai
Stop	STOP	Relay STOP aktif	Sesuai
Reset Counter	RESET	Counter produksi	Sesuai

		kembali ke nol	
Master ON	MASTER ON	Relay MASTER ON aktif	Sesuai
Cylinder 1 ON	CYL1_ON	Relay Cylinder 1 aktif	Sesuai
Cylinder 1 OFF	CYL1_OFF	Relay Cylinder 1 nonaktif	Sesuai
Cylinder 2 ON	CYL2_ON	Relay Cylinder 2 aktif	Sesuai
Cylinder 2 OFF	CYL2_OFF	Relay Cylinder 2 nonaktif	Sesuai
Cylinder 3 ON	CYL3_ON	Relay Cylinder 3 aktif	Sesuai
Cylinder 3 OFF	CYL3_OFF	Relay Cylinder 3 nonaktif	Sesuai

Tingkat keberhasilan perintah: $(10/10) \times 100\% = 100\%$. Seluruh 10 perintah berhasil dikirim dari dashboard ke Firebase dan dieksekusi dengan benar oleh ESP32-S3. Perintah START, STOP, dan MASTER ON beroperasi dalam mode pulse (1500 ms), sementara perintah Cylinder 1–3 bekerja dalam mode ON/OFF persisten. Mekanisme reset IDLE setelah setiap eksekusi terbukti efektif mencegah eksekusi perintah berulang suatu fitur keselamatan penting untuk menghindari operasi mesin yang tidak disengaja.

Keunggulan utama arsitektur kontrol berbasis Firebase ini adalah tidak memerlukan konektivitas jaringan lokal langsung antara dashboard dan ESP32-S3, sehingga memungkinkan kontrol mesin jarak jauh dari lokasi mana pun yang terhubung internet. Namun, arsitektur ini juga menimbulkan ketergantungan terhadap stabilitas koneksi internet. Pada lingkungan industri dengan konektivitas tidak stabil, latensi eksekusi perintah dapat terjadi sebuah trade-off yang melekat pada sistem kontrol IoT berbasis cloud yang perlu dipertimbangkan dalam deployment produksi.

4.6 Data Ringkasan Riwayat Harian

Akurasi sinkronisasi data historis antara Firebase dan halaman history diuji melalui perbandingan 12 parameter ringkasan produksi harian. Hasil pengujian ditampilkan pada Tabel 5.

Tabel 5. Hasil Pengujian Data Ringkasan Riwayat Harian

Item Data	Data Firebase	Data Halaman Histori	Ket.
Tanggal produksi	21/05/2026	21/05/2026	Sesuai
Jumlah mesin beroperasi	1	1	Sesuai
Total Good harian	16	16	Sesuai
Total NG harian	8	8	Sesuai
Total produksi harian	24	24	Sesuai
Persentase Good harian	66.7%	66.7%	Sesuai
Persentase NG harian	33.3%	33.3%	Sesuai
Runtime harian	00:12:21	00:12:21	Sesuai
Downtime harian	00:03:53	00:03:53	Sesuai
Jumlah WARNING harian	0	0	Sesuai
Jumlah CRITICAL harian	13	13	Sesuai
Threshold saat ini	W:10% C:20% Min:20	W:10% C:20% Min:20	Sesuai

Akurasi data riwayat: $(12/12) \times 100\% = 100\%$. Seluruh 12 parameter data historis, termasuk metrik produksi harian, durasi *runtime/downtime*, frekuensi status peringatan, dan konfigurasi *threshold* aktif, menunjukkan kesesuaian sempurna antara Firebase dan halaman histori. Kelengkapan parameter historis ini termasuk jumlah kejadian WARNING dan CRITICAL harian mendukung analisis tren kualitas produksi jangka panjang

yang tidak dapat difasilitasi oleh sistem monitoring berbasis tampilan *real-time* semata.

4.7 Fitur Halaman Histori

Pengujian fitur halaman histori mencakup empat aspek fungsional yang mendukung analisis dan dokumentasi produksi. Hasil pengujian ditunjukkan pada Tabel 6.

Tabel 6. Pengujian Fitur Halaman Histori

Fitur	Hasil yang Diharapkan	Hasil Pengamatan	Ket.
Laporan Otomatis	Menampilkan ringkasan produksi dalam bentuk teks naratif	Laporan muncul sesuai data harian	Sesuai
Grafik <i>Cycle time Analysis</i>	Menampilkan perubahan <i>cycle time</i> secara visual	Grafik muncul pada halaman histori	Sesuai
Filter Jam	Filter histori berdasarkan rentang waktu	Data riwayat terfilter sesuai jam yang dipilih	Sesuai
Tabel Riwayat	Menampilkan rekaman riwayat produksi	Tabel menampilkan Riwayat produksi sehari	Sesuai

Empat fitur berhasil diverifikasi: laporan otomatis menampilkan ringkasan produksi dalam bentuk teks naratif sesuai data harian; grafik *cycle time analysis* menampilkan perubahan *cycle time* secara visual; filter jam memfilter data riwayat sesuai rentang waktu yang dipilih; dan tabel riwayat menampilkan rekaman data produksi terkini dengan benar. Tingkat keberhasilan fitur: $(4/4) \times 100\% = 100\%$.

Kehadiran fitur laporan naratif otomatis dan visualisasi grafik *cycle time* merupakan diferensiasi fungsional dibandingkan sistem monitoring IoT konvensional yang umumnya hanya menyediakan tampilan data tabular. Kemampuan filter jam memungkinkan analisis

performa produksi pada segmen waktu tertentu, yang berguna untuk identifikasi periode *downtime* atau penurunan kualitas secara terlokalisasi.

4.8 Eksekusi Perintah ESP32-S3

Pengujian akhir memverifikasi integritas alur kontrol secara *end-to-end*: web dashboard → Firebase → ESP32-S3 → relay. Hasil pengujian ditampilkan pada Tabel 7.

Tabel 7. Pengujian Perintah

Aksi	Perintah Firebase	Pembacaan ESP32-S3	Status Akhir	Ket.
Start	START	START	IDLE	Sesuai
Stop	STOP	STOP	IDLE	Sesuai
Reset Counter	RESET	RESET	IDLE	Sesuai
Master ON	MASTER_ON	MASTER_ON	IDLE	Sesuai
Cylinder 1 ON	CYL1_ON	CYL1_ON	IDLE	Sesuai
Cylinder 1 OFF	CYL1_OFF	CYL1_OFF	IDLE	Sesuai
Cylinder 2 ON	CYL2_ON	CYL2_ON	IDLE	Sesuai
Cylinder 2 OFF	CYL2_OFF	CYL2_OFF	IDLE	Sesuai
Cylinder 3 ON	CYL3_ON	CYL3_ON	IDLE	Sesuai
Cylinder 3 OFF	CYL3_OFF	CYL3_OFF	IDLE	Sesuai

Seluruh perintah berhasil dibaca dan dieksekusi oleh ESP32-S3 sesuai nilai yang dikirim dari dashboard melalui Firebase, dengan setiap perintah kembali ke status IDLE setelah eksekusi. Tingkat keberhasilan eksekusi: $(10/10) \times 100\% = 100\%$.

4.9 Rekapitulasi dan Pembahasan Umum

Rekapitulasi hasil pengujian seluruh tujuh kategori disajikan pada Tabel 8.

Tabel 8. Rekapitulasi Pengujian Sistem (55 item, tingkat keberhasilan keseluruhan 100%)

Kategori Pengujian	Item Diuji	Item Sesuai	Tingkat Keberhasilan
Login dan hak akses pengguna	5	5	100%
Tampilan monitoring dashboard	10	10	100%
<i>Threshold</i> dan status peringatan kualitas	4	4	100%
Tombol kontrol dan mode manual	10	10	100%
Data ringkasan riwayat harian	12	12	100%
Fitur halaman history	4	4	100%
Eksekusi perintah ESP32-S3	10	10	100%
TOTAL	55	55	100%

7 kategori pengujian dengan total 55 item uji mencapai tingkat keberhasilan 100%. Hasil ini mengonfirmasi bahwa sistem monitoring dan kontrol mesin stamping berbasis IoT yang dikembangkan memenuhi seluruh kebutuhan fungsional sesuai spesifikasi perancangan.

4.10 Pembahasan

Dibandingkan dengan sistem monitoring berbasis ESP32–Firebase yang dilaporkan dalam literatur sebelumnya, sistem yang diusulkan menunjukkan tingkat integrasi fungsional yang lebih komprehensif. Platform ini menggabungkan monitoring multi-parameter *real-time*, RBAC tiga tingkat, konfigurasi *threshold* bertingkat dengan parameter Minimum Sample, manajemen riwayat produksi, pelaporan naratif otomatis, visualisasi grafik *cycle time*, serta pengiriman perintah kontrol relay berbasis Firebase dalam satu dashboard daring yang dapat diakses secara global melalui Netlify. Integrasi ini melampaui fungsi monitoring IoT dasar dan menunjukkan konvergensi antara fungsi monitoring, manajemen data, pelaporan, visualisasi, otorisasi, dan kontrol dalam satu platform.

Arsitektur modular yang memisahkan PLC (kontrol mesin lokal), Raspberry Pi (validasi kualitas berbasis visi), dan ESP32-S3 (IoT gateway dan kontrol relay) sesuai dengan prinsip separation of concerns dalam rekayasa sistem. Keunggulan kritis dari arsitektur ini adalah: kegagalan subsistem monitoring tidak menghentikan operasi produksi karena kontrol mesin tetap dikelola secara independen oleh PLC. Hal ini berbeda secara fundamental dari arsitektur monitoring-kontrol terintegrasi yang rentan terhadap kegagalan tunggal (*single point of failure*).

Meskipun demikian, terdapat beberapa keterbatasan yang perlu diakui. Sistem sepenuhnya bergantung pada stabilitas konektivitas internet; gangguan jaringan akan menghentikan pembaruan dashboard *real-time* dan pengiriman perintah kontrol, yang merupakan risiko operasional signifikan di lingkungan industri. Selain itu, konfigurasi Firebase Rules masih berada pada tingkat pengembangan sehingga deployment produksi memerlukan implementasi Firebase Authentication dengan keamanan berbasis token. Sistem juga belum dievaluasi pada skenario akses multi-pengguna simultan, dan subsistem YOLOv8 sebagai sumber data tidak dievaluasi dalam penelitian ini akurasi model deteksi secara langsung memengaruhi kualitas data monitoring. Fitur ekspor laporan dalam format PDF atau Excel juga belum tersedia, padahal kebutuhan ini umum dalam lingkungan industri.

Dari perspektif implikasi teoritis, penelitian ini mendemonstrasikan bahwa Firebase RTDB dapat berfungsi secara efektif sebagai middleware pertukaran data asinkron antara perangkat IoT *edge* (ESP32-S3), sistem visi komputer (Raspberry Pi + YOLOv8), dan antarmuka web berbasis cloud—tanpa memerlukan server *backend* khusus. Pola arsitektur ini berimplikasi pada desain sistem IoT industri skala kecil hingga menengah yang membutuhkan integrasi monitoring dan kontrol dengan biaya infrastruktur minimal. Secara implementasi, penggunaan Netlify sebagai platform hosting memungkinkan aksesibilitas global tanpa biaya server atau konfigurasi jaringan yang kompleks, yang relevan untuk industri manufaktur terdistribusi dengan kebutuhan monitoring lintas lokasi.

5. KESIMPULAN

Penelitian ini berhasil mengembangkan sistem monitoring dan kontrol mesin stamping berbasis IoT yang mengintegrasikan ESP32-S3, Firebase Realtime Database, dan web dashboard daring. Berdasarkan hasil penelitian, disimpulkan sebagai berikut:

- Sistem memenuhi seluruh spesifikasi fungsional yang dirancang dengan tingkat keberhasilan 100% pada 55 item uji dalam tujuh kategori pengujian.
- Data monitoring *real-time* dari 10 parameter produksi tersinkronisasi secara akurat antara Firebase dan dashboard tanpa kehilangan atau kerusakan data.
- Mekanisme *Minimum Sample* pada sistem *threshold* tiga parameter efektif mencegah false positive alarm di fase awal produksi.
- Seluruh perintah kontrol relay berhasil dieksekusi melalui alur dashboard → Firebase → ESP32-S3, dengan reset IDLE otomatis untuk mencegah eksekusi berulang.
- RBAC tiga tingkat (Admin, Operator, Viewer) berhasil memisahkan hak akses sesuai peran sehingga memperkuat keselamatan operasional sistem.
- Arsitektur modular memastikan kegagalan subsistem monitoring tidak mengganggu operasi mesin karena kontrol utama tetap dikelola secara independen oleh PLC.
- Ketergantungan penuh pada konektivitas internet dan konfigurasi Firebase Rules yang belum production-ready menjadi keterbatasan utama yang perlu ditangani sebelum deployment skala industri.
- Pengembangan selanjutnya meliputi offline buffering, Firebase Authentication berbasis token, evaluasi *end-to-end* akurasi YOLOv8, fitur ekspor PDF/Excel, dan perluasan ke arsitektur multi-mesin.

UCAPAN TERIMA KASIH

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

DAFTAR PUSTAKA

- [1] R. Maududy and D. R. Nursamsi, "Pengembangan *Real-time* Monitoring dan Data Logging Berbasis Web Pada Proses Robot Painting untuk Meningkatkan Efisiensi Produksi," *Informatics Digital Expert*, vol. 5, no. 2, pp. 89–94, 2023, doi: <https://doi.org/10.36423/index.v5i1.1586>.
- [2] G. H. Rajagukguk, A. Widianono, J. Studi, S. Informasi, and U. E. Unggul, "Perancangan dan Implementasi Sistem Monitoring Mesin Produksi Berbasis Lampu IoT dan Web Dashoard di PT Motor Jaya Indonesia," *J. Minfo Polgan*, vol. 14, pp. 2303–2317, 2025, doi: <https://doi.org/10.33395/jmp.v14i2.15413> e-ISSN.
- [3] B. Herdiana, E. B. Setiawan, and U. Sartoyo, "Tinjauan Komprehensif Evolusi , Aplikasi , dan Tren Masa Depan Programmable Logic Controllers A Comprehensive Review of the Evolution , Applications , and Future Trends of Programmable Logic Controllers," *TELEKONTRAN*, vol. 11, no. 2, pp. 173–193, 2023, doi: [10.34010/telekontran.v11i2.12896](https://doi.org/10.34010/telekontran.v11i2.12896).
- [4] A. Hijazi, M. Ando, and Z. Podor, "Data Losses and Synchronization According to Delay in PLC-Based Industrial Automation Systems," *Heliyon*, vol. 10, 2024, doi: [10.1016/j.heliyon.2024.e37560](https://doi.org/10.1016/j.heliyon.2024.e37560).
- [5] B. Padmaja, E. K. R. Patro, S. Mahurkar, and G. Akhila, "Google Firebase based Modern IoT System Architecture," pp. 107–110, 2021.
- [6] A. H. Awad *et al.*, "Low-cost IoT-Based sensors dashboard for monitoring the state of health of mobile harbor cranes: Hardware and software description," *Heliyon*, vol. 10, no. 22, p. e40239, 2024, doi: <https://doi.org/10.1016/j.heliyon.2024.e40239>.
- [7] A. Handojo, E. Setijarso, S. Halim, and T. Octavia, "Integrating *Real-time* IoT Based Monitoring and Dashboard Design for Closed-House Hen Farming," *J. Tek. Ind. J. Keilmuan dan Apl. Tek. Ind.*, vol. 27, pp. 237–248, Nov. 2025, doi: [10.9744/jti.27.2.237-248](https://doi.org/10.9744/jti.27.2.237-248).
- [8] I. N. W. R. Samkayana and B. G. K. Yudistira, "Implementasi Firebase Realtime Database Cat Feeder Berbasis Mikrokontroler ESP32," *JITET (Jurnal Inform. dan Tek. Elektro Ter.)*, vol. 13, no. 3, 2025, doi: <https://doi.org/10.23960/jitet.v13i3.7198>.
- [9] A. Maulida, C. S. Fayzalmi, F. Ata, C. Anaam, and D. Ardiansyah, "Design of IoT-Based Smart Trash Bin Monitoring Using ESP32 And Firebase," *J. Komput. Teknol. Inf. Sist. Komput.*, vol. 4, no. 2, pp. 420–428, 2025, doi: <https://doi.org/10.62712/juktisi.v4i2.420>.
- [10] X. Wang, H. Hu, Y. Wang, and Z. Wang, "IoT *Real-time* Production Monitoring and Automated Process Transformation in Smart Manufacturing," *J. Organ. End User Comput.*, vol. 36, no. 1, 2024, doi: <https://doi.org/10.4018/JOEUC.336482>.
- [11] M. Firmansyah *et al.*, "Implementasi Role-

- Based Acces Control pada Sistem Informasi Layanan Administrasi,” *JMIK (Jurnal Mhs. Ilmu Komputer)*, vol. 6, no. 2, pp. 113–122, 2025, doi: <https://doi.org/10.24127/ilmukomputer.v6i2.9875>.
- [12] D. Witczak and S. Szymoniak, “Review of Monitoring and Control Systems Based on Internet of Things,” 2024. doi: [10.3390/app14198943](https://doi.org/10.3390/app14198943).
- [13] S. Makhmudah, W. T. Setiawan, S. Aprilia, E. A. Suryono, and Y. Prastyo, “Implementation of Digitalization of Daily Production Report in Manufacturing Industry to Improve Data Efficiency and Accuracy,” *Int. J. Res. Innov. Soc. Sci.*, vol. IX, no. 2454, pp. 2949–2956, 2025, doi: <https://dx.doi.org/10.47772/IJRISS.2025.9010239>.
- [14] S. Sarkar, A. Jana, S. Maity, C. Maiti, and C. K. Bera, “IoT Based Performance Display System Using ESP32: Design and Development,” *J. Phys. Conf. Ser.*, vol. 2576, no. 1, p. 12002, 2023, doi: [10.1088/1742-6596/2576/1/012002](https://doi.org/10.1088/1742-6596/2576/1/012002).
- [15] F. Rahayu, A. Octa, and H. Hardiyanto, “IoT-Based Kitchen Monitoring System with ESP32 and Firebase Realtime,” *ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron.*, vol. 14, no. 1, pp. 96–109, 2026, doi: <http://dx.doi.org/10.26760/elkomika.v14i1.96>.
- [16] Bahosin Sihombing, Zamsiswaya, and sawaluddin, “Model Pengembangan 4D (Define, Design, Develop, dan Disseminate) dalam Pembelajaran Pendidikan Islam,” *J. Islam. Educ. El Madani*, vol. 4, no. 1 SE-Articles, pp. 11–19, Dec. 2024, doi: [10.55438/jiee.v4i1.135](https://doi.org/10.55438/jiee.v4i1.135).
- [17] H. Taherdoost, “R&D Process BT - Innovation Through Research and Development: Strategies for Success,” H. Taherdoost, Ed., Cham: Springer Nature Switzerland, 2024, pp. 111–133. doi: [10.1007/978-3-031-52565-0_6](https://doi.org/10.1007/978-3-031-52565-0_6).
- [18] Torang Siregar, “Stages of Research and Development Model Research and Development (R&D),” *DIROSAT J. Educ. Soc. Sci. Humanit.*, vol. 1, no. 4 SE-Articles, pp. 142–158, Oct. 2023, doi: [10.58355/dirosat.v1i4.48](https://doi.org/10.58355/dirosat.v1i4.48).