

RachelAmabelTambun- FinalSkripsi.docx

by JASA PENGECEKAN PLAGIASI WHATSAPP: 085935293540

Submission date: 16-Jul-2025 07:33AM (UTC+0300)

Submission ID: 2685763611

File name: RachelAmabelTambun-FinalSkripsi.docx (6.53M)

Word count: 11978

Character count: 75284

**RANCANG BANGUN SISTEM *MONITORING* PADA
KANDANG BURUNG PUYUH PETELUR SECARA
OTOMATIS BERBASIS *INTERNET OF THINGS***

SKRIPSI

Disusun Sebagai Salah Satu Syarat untuk Memperoleh Gelar Sarjana Teknik (S.T)



Disusun Oleh

RACHEL AMABEL TAMBUN

NIM. 3332200018

JURUSAN TEKNIK ELEKTRO

FAKULTAS TEKNIK

UNIVERSITAS SULTAN AGENG TIRTAYASA

2025

LEMBAR PERNYATAAN KEASLIAN SKRIPSI

Dengan ini saya sebagai Penulis Skripsi berikut:

Judul	Rancang Bangun Sistem ⁵ Monitoring Pada Kandang Burung Puyuh Petelur Secara Otomatis Berbasis Internet of Things
Nama Mahasiswa	Rachel Amabel Tambun
NIM	3332200018
Fakultas/Jurusan	Teknik/Teknik Elektro

¹ Menyatakan dengan sesungguhnya bahwa Skripsi ini benar-benar hasil asli karya saya dan tidak memuat hasil karya orang lain, kecuali dinyatakan melalui rujukan yang benar dan dapat dipertanggung jawabkan. Apabila di kemudian hari ditemukan hal-hal yang menunjukkan bahwa sebagian atau seluruh karya ini bukan karya saya, maka saya bersedia dituntut melalui hukum yang berlaku. Saya juga bersedia menanggung segala akibat hukum yang timbul dari pernyataan yang secara sadar telah saya nyatakan melalui lembar ini.

Cilegon, 6 Januari 2025

RACHEL AMABEL TAMBUN

NIM. 3332200018

LEMBAR PENGESAHAN

Dengan ini ditetapkan bahwa Skripsi berikut:

Judul	Rancang Bangun Sistem Monitoring Pada Kandang Burung Puyuh Petelur Secara Otomatis Berbasis <i>Internet of Things</i>
Nama Mahasiswa	Rachel Amabel Tambun
NIM	3332200018
Fakultas/Jurusan	Teknik/Teknik Elektro

Telah diuji dan dipertahankan pada tanggal **2 JUNI 2025** melalui sidang Skripsi di Fakultas Teknik Universitas Sultan Ageng Tirtayasa dan dinyatakan **LULUS.**

Dewan Penguji

Tanda Tangan

Pembimbing I	Rian Fahrizal, S.T., M.Eng
Pembimbing II	Ceri Ahendyarti, S.T., M.Eng
Penguji I	Heri Haryanto, S.T., M.T
Penguji II	Dina Estining Tyas Lufianawati, S.T., M.T.
	Mengetahui, Ketua Jurusan Teknik Elektro

Dr. Eng. Rocky Alfanz, M.Sc
NIP. 198103282010121001

PRAKATA

Puji Syukur ke hadirat Tuhan Yang Maha Esa atas karunia-Nya sehingga penulis mampu menyelesaikan skripsi berjudul “Rancang Bangun Sistem Monitoring Pada Kandang Burung Puyuh Petelur Secara Otomatis Berbasis Internet of Things” tepat waktu. Skripsi ini ditulis untuk memenuhi syarat memperoleh gelar Sarjana Teknik Pada Program Studi Teknik Elektro Fakultas Teknik Universitas Sultan Ageng Tirtayasa.

Selama penulisan Skripsi, penulis mendapatkan dukungan berupa pengajaran, bimbingan, saran dan masukan dari berbagai pihak. Atas dukungan tersebut, penulis mengucapkan terima kasih kepada:

1. Bapak Dr. Eng. Rocky Alfan, M.Sc selaku Ketua Program Studi Teknik Elektro Fakultas Teknik Universitas Sultan Ageng Tirtayasa.
2. Kedua orang tua penulis yang telah memberikan dukungan, doa, dan semangat kepada Penulis selama ini.
3. Bapak HM. Hartono, S.T, M.T. selaku Dosen Pembimbing Akademik pada Jurusan Teknik Elektro Fakultas Teknik Universitas Sultan Ageng Tirtayasa.
4. Bapak Rian Fahrizal, S.T., M.Eng selaku Dosen Pembimbing 1 yang telah membimbing dan mengarahkan dalam proses penulisan skripsi.
5. Ibu Ceri Ahendyarti, S.T., M.Eng selaku Dosen Pembimbing 2 yang telah membimbing dan mengarahkan dalam proses penulisan skripsi.

Penulis menyadari penelitian ini memiliki kekurangan karena keterbatasan kemampuan dan ilmu yang dimiliki oleh Penulis. Oleh karena itu, penulis mengharapkan kritik, saran, dan masukan untuk kemajuan hasil penelitian sehingga bermanfaat bagi pembaca.

Cilegon, 6 Januari 2025

Rachel Amabel Tambun

ABSTRAK

Rachel Amabel Tambun

Teknik Elektro

Rancang Bangun **S**istem *Monitoring* Kandang Burung Puyuh Petelur Secara Otomatis Berbasis *Internet of Things*

Burung puyuh memerlukan kondisi lingkungan stabil agar kondisi kandang tetap ideal. Oleh karena itu, dirancang sistem *monitoring* otomatis kandang burung puyuh petelur berbasis *Internet Of Things* (IoT). Sistem monitoring dibuat berfokus dengan tiga parameter pengukuran yakni suhu, kelembapan, dan tingkat kebisingan kandang. Sensor DHT22 mengukur suhu dan kelembapan sedangkan sensor Max4466 mendeteksi tingkat kebisingan kandang. Pengukuran dilakukan pada pagi, siang, sore, malam, dan dini hari. Data pengukuran dikirimkan secara *online* dengan jaringan internet ke *firebase* dan Google Sheet sehingga dapat dipantau *realtime* melalui *website*. Kondisi ideal kandang burung puyuh ketika suhu sebesar 24 s.d. 30°C, kelembapan 70 s.d. 80% dan kebisingan kurang dari 60 dB. Sensor DHT22 yang digunakan memiliki *error* sebesar 3,24% dan sensor Max4466 sebesar 1%. Terakhir, hasil pengukuran divisualisasikan dalam sebuah grafik garis. Penggunaan IoT diharapkan mampu meningkatkan kondisi kandang yang ideal bagi burung puyuh sehingga perkembangbiakan burung puyuh semakin optimal.

Kata Kunci: Mikrokontroler ESP8266, DHT22, Max4466, Burung Puyuh, IoT

ABSTRACT

Rachel Amabel Tambun

Electrical Engineering

43

Design of an Automatic Monitoring System for Layer Quail Cages Based on
Internet Of Things

Quails require stable environmental conditions to maintain ideal cage conditions. Therefore, an automatic monitoring system for laying quail cages based on the Internet of Things (IoT) was designed. The monitoring system focuses on three measurement parameters, namely temperature, humidity, and noise level of the cage. The DHT22 sensor measures temperature and humidity while the Max4466 sensor detects the noise level of the cage. Measurements are taken in the morning, afternoon, evening, night, and early morning. Measurement data is sent online with the internet network to Firebase and Google Sheet so that it can be monitored realtime through the website. The ideal condition of the quail cage is when the temperature is 24 to 30°C, humidity is 70 to 80% and noise is less than 60 dB. The DHT22 sensor used has an error of 3.24% and the Max4466 sensor of 1%. Finally, the measurement results are visualised in a line graph. The use of IoT is expected to improve the ideal cage conditions for quails so that quail breeding is optimised.

Keywords: ESP8266 Microcontroller, DHT22, Max4466, Quail, IoT

DAFTAR ISI

Halaman

LEMBAR PERNYATAAN KEASLIAN SKRIPSI	ii
LEMBAR PENGESAHAN	iii
PRAKATA	iv
ABSTRAK	v
ABSTRACT	vi
DAFTAR ISI	vii
DAFTAR GAMBAR	x
DAFTAR TABEL	xi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan Penelitian	3
1.4 Manfaat Penelitian	3
1.5 Batasan Masalah	4
1.6 Sistematika Penulisan	4
BAB II TINJAUAN PUSTAKA	6
2.1 Burung Puyuh	6
2.2 Mikrokontroler	7
2.3 Sensor DHT22	7
2.4 Sensor Max4466	8
2.5 Lampu Pijar	10
2.6 <i>Internet of Things</i>	11
2.7 <i>Database</i>	11
2.8 <i>Website</i>	12
2.9 <i>Firebase</i>	12
2.10 Perhitungan Nilai Error	13

2.11	Tinjauan Pustaka	14
BAB III	METODOLOGI PENELITIAN	16
3.1	Alur Penelitian	16
3.2	Komponen Penelitian	17
3.3	Metode Penelitian	22
3.3.1	Diagram Alir	23
3.3.2	Diagram Blok	24
3.4	Perancangan <i>Hardware</i>	25
BAB IV	HASIL DAN ANALISIS	27
4.1	Lokasi Pengambilan Data	27
4.2	Analisis <i>Listing Code</i>	31
4.2.1	Inisialisasi dan Konfigurasi	31
4.2.2	Koneksi dan Pengaturan WiFi	34
4.2.3	Tampilan dan Pembacaan Sensor	35
4.2.4	Pengolahan dan Pengiriman Data	38
4.2.5	Mengontrol Logika Berdasarkan Data	40
4.2.6	Fungsi <i>Loop</i> Utama	41
4.3	Analisis Hasil Kalibrasi Alat	43
4.3.1	Analisis Kalibrasi Sensor DHT22	43
4.3.2	Analisis Kalibrasi Sensor Max4466	44
4.4	Analisis Hasil Data	45
4.4.1	Analisis Hasil Pengujian Pagi Hari	45
4.4.2	Analisis Hasil Pengujian Siang Hari	47
4.4.3	Analisis Hasil Pengujian Sore Hari	49
4.4.4	Analisis Hasil Pengujian Malam Hari	50
4.4.5	Analisis Hasil Pengujian Dini Hari	52
BAB V	PENUTUP	54
5.1	Kesimpulan	54
5.2	Saran	54
	DAFTAR PUSTAKA	55
	LAMPIRAN A TAMPILAN ALAT DAN <i>PROTOTYPE</i>	A-1

LAMPIRAN B TAMPILAN <i>FIREBASE</i> DAN <i>WEBSITE</i>	41 B-1
LAMPIRAN C PEROLEHAN DATA	C-1
LAMPIRAN D PERHITUNGAN KALIBRASI.....	D-1
LAMPIRAN E LISTING CODE.....	E-1
LAMPIRAN F RANGKAIAN SKEMATIK.....	F-1

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Burung Puyuh	6
Gambar 2.2 Sensor DHT22	8
Gambar 2.3 Sensor Max4466	9
Gambar 2.4 Lampu Pijar	10
Gambar 3.1 Kandang Burung	17
Gambar 3.2 Mikrokontroler ESP8266	17
Gambar 3.3 Sensor DHT22	18
Gambar 3.4 Sensor Max4466	19
Gambar 3.5 Step-Down LM25966	20
Gambar 3.6 Lampu Pijar	21
Gambar 3.7 Diagram Alir Program IoT	23
Gambar 3.8 Diagram Alir Website	24
Gambar 3.9 Diagram Blok	24
Gambar 3.10 Skematik Hardware	25
Gambar 4.1 Kandang Box	27
Gambar 4.2 Kondisi Prototipe	28
Gambar 4.3 Kondisi Ideal dan Tidak Ideal	29
Gambar 4.4 Tampilan Hasil Pengukuran Pada Website dan Firebase	30
Gambar 4.5 Inisialisasi Library	31
Gambar 4.6 Konfigurasi Firebase dan Google Sheet	32
Gambar 4.7 Inisialisasi Pin Sensor	33
Gambar 4.8 Listing Code Koneksi WiFi	34
Gambar 4.9 Listing Code DHT22	35
Gambar 4.10 Listing Code Pengukuran Kebisingan Kandang	36
Gambar 4.11 Listing Code Pengiriman Data Ke Firebase	38
Gambar 4.12 Contoh Tampilan Data Di Firebase	38
Gambar 4.13 Listing Code Pengiriman Data Ke Google Spread Sheet	39
Gambar 4.14 Listing Code Logika Kontrol LED dan Relay	40
Gambar 4.15 Listing Code Fungsi Loop Utama	41
Gambar 4.16 Grafik Kondisi Kandang Pagi Hari	46
Gambar 4.17 Grafik Kondisi Kandang Siang Hari	48
Gambar 4.18 Grafik Kondisi Kandang Sore Hari	49
Gambar 4.19 Grafik Kondisi Kandang Malam Hari	51
Gambar 4. 20 Grafik Kondisi Kandang Burung Dini Hari	52

DAFTAR TABEL

	Halaman
Tabel 3.1 <i>Pin</i> ESP8266	18
Tabel 3.2 <i>Pin</i> Sensor DHT22	19
Tabel 3.3 <i>Pin</i> Sensor Max4466	20
Tabel 3.4 <i>Pin Step Down</i> LM25966	20
Tabel 3.5 Terminal Lampu Pijar	21
Tabel 3.6 Fungsi Komponen <i>Hardware</i>	26
Tabel 3.7 Jadwal Penelitian.....	Error! Bookmark not defined.
Tabel 4.1 Kalibrasi Sensor DHT22	43
Tabel 4.2 Kalibrasi Sensor Max4466	44
Tabel 4.3 Hasil Pengujian Kandang Pagi Hari	45
Tabel 4.4 Hasil Pengujian Kandang Siang Hari	47
Tabel 4.5 Hasil Pengujian Sore Hari	49
Tabel 4.6 Hasil Pengujian Malam Hari	50
Tabel 4.7 Hasil Pengujian Dini Hari	52

PENDAHULUAN

1.1 Latar Belakang

Protein merupakan nutrisi penting bagi pertumbuhan, perbaikan jaringan, pembentuk sel tubuh, hingga berperan dalam proses metabolisme [1]. Berdasarkan sumbernya, protein digolongkan menjadi dua yakni protein hewani dan protein nabati. Sumber protein hewani seperti telur lebih berkualitas dibandingkan protein nabati [2].

Burung puyuh tergolong unggas petelur yang mampu menghasilkan 250 s.d. 300 butir telur setiap tahun. Walaupun demikian, produktivitas burung puyuh sangat dipengaruhi oleh kondisi lingkungan di sekitar kandang seperti suhu, kelembapan, dan kualitas udara [3]. Pengelolaan kandang burung yang baik diperlukan agar produktivitas burung puyuh tetap optimal. Namun, pengelolaan manual sering tidak efektif mempertahankan kondisi optimal sehingga tingkat produktivitas menurun dan meningkatkan risiko tertular penyakit. Oleh karena itu, dibutuhkan sistem *monitoring* otomatis untuk memantau dan mengendalikan kondisi lingkungan pada kandang secara *realtime* [4].

Di era industri 4.0 *Internet of Things* (IoT) banyak digunakan dalam berbagai sektor termasuk pertanian dan peternakan. Beberapa penelitian sebelumnya telah mengembangkan sistem *monitoring* berbasis IoT pada peternakan unggas khususnya burung puyuh petelur. Pengembangan sistem inkubator telur puyuh dengan mengontrol suhu dan kelembapan menggunakan DHT11 sehingga pemantauan jarak jauh secara *realtime* dan otomatis [5][6]. *Monitoring* yang tidak konsisten membuat suhu kandang meningkat sehingga tingkat konsumsi pakan dan produktivitas burung puyuh menurun [7].

Penelitian selanjutnya menyatakan bahwa kualitas telur puyuh sangat dipengaruhi oleh suhu dan kelembapan dalam kandang [8]. Aplikasi *monitoring* yang terintegrasi meningkatkan proses pemeliharaan secara cepat [9]. Penelitian mengenai implementasi sistem *monitoring* juga pernah dilaksanakan menggunakan

platform Node-Red dan metode Naïve Bayes mencapai akurasi 80% [10]. Kemudian dikembangkan sistem kendali dan *monitoring* IoT menggunakan sensor DHT22 dan sensor MQ135 yang bertujuan menjaga kondisi kandang ayam tetap optimal [11].

Pengembangan sistem *monitoring* menggunakan sensor proximity E-18 D80NK yang terhubung dengan platform IoT melalui MQTT dan dapat dipantau *realtime* mempermudah pengelolaan produksi telur puyuh [12]. Implementasi sistem *monitoring* dan kontrol pakan burung puyuh berbasis mikrokontroler dan IoT secara otomatis memastikan ketersediaan pakan tepat waktu. Lalu dibuat sistem *monitoring* suhu dan kelembapan alat penetas telur ayam berbasis IoT [13]. Selain itu, dikembangkan sistem otomatisasi pakan minum dan *monitoring* kelembapan udara pada kandang ayam berbasis IoT menggunakan sensor DHT22 dan mikrokontroler untuk memantau dan mengontrol kondisi lingkungan dalam kandang, yang dapat diadaptasi untuk kandang burung puyuh guna meningkatkan efisiensi produksi [14].

Kemudian dilakukan kajian distribusi temperatur pada inkubator penetas telur puyuh yang bertujuan untuk mendapatkan sistem pemanas yang seragam dalam ruang inkubator, sehingga proses penetasan telur puyuh dapat berlangsung optimal [15]. Pengembangan sistem pemantauan kualitas udara berbasis IoT di peternakan dengan parameter suhu, kelembapan, dan kadar gas amonia dalam kandang menjaga kesehatan dan produktivitas burung puyuh [16]. Pengembangan sistem berbasis IoT terbukti mengurangi stres panas yang menyebabkan penurunan produktivitas burung puyuh [17].

Dari penelitian sebelumnya terlihat parameter suhu dan kelembapan kandang berperan penting terhadap tingkat produktivitas burung puyuh dan kualitas telur yang dihasilkan. Oleh karena itu, dirancang suatu perangkat IoT yang mampu mendeteksi perubahan suhu dan kelembapan kandang burung dan dapat dipantau dengan jarak jauh melalui sebuah *website*. Pemantauan jarak jauh berbasis *website* menjadi lebih fleksibel dan efisien. Hal ini tentunya akan menguntungkan para peternak burung sehingga dapat meningkatkan tingkat produktivitas burung puyuh.

1.2 Rumusan Masalah

Rumusan masalah yang menjadi dasar dalam melakukan penelitian ini di antaranya:

1. Bagaimana implementasi sistem *monitoring* otomatis berbasis IoT pada kandang burung puyuh?
2. Bagaimana tampilan hasil *monitoring* kandang burung puyuh secara *realtime* dengan *firebase* dan *website*?
3. Apakah sensor DHT22 dan sensor Max4466 akurat digunakan dalam *monitoring* otomatis ?

1.3 Tujuan Penelitian

Berikut tujuan dilaksanakannya penelitian mengenai penggunaan IoT dalam kandang burung puyuh:

1. Mengimplementasikan sistem *monitoring* otomatis berbasis IoT pada kandang burung puyuh.
2. Menampilkan dan memvisualisasikan hasil *monitoring* kandang burung puyuh secara *realtime* melalui *website* dan *firebase*.
3. Menguji keakuratan pembacaan sensor DHT22 dan sensor Max4466 dalam sistem *monitoring* otomatis.

1.4 Manfaat Penelitian

Manfaat penelitian dari *monitoring* kandang burung puyuh petelur antara lain:

1. Mengetahui dampak suhu, kelembapan, dan tingkat kebisingan terhadap burung puyuh.
2. Memberikan inovasi terbaru dalam perancangan sistem *monitoring* otomatis pada kandang burung puyuh.

1.5 Batasan Masalah

Batasan masalah bertujuan membatasi uraian permasalahan sehingga setiap masalah dapat digambarkan dengan jelas. Batasan masalah dalam penelitian ini di antaranya:

1. Objek penelitian hanya berfokus kepada burung puyuh.
2. Parameter pengamatan berfokus kepada suhu, kelembapan, dan tingkat kebisingan kandang.
3. Sistem *monitoring* hanya menjaga suhu dan kelembapan kandang tetap stabil menggunakan lampu pijar.
4. Teknologi IoT untuk melihat data pengukuran suhu, kelembapan, dan tingkat kebisingan kandang burung puyuh.

1.6 Sistematika Penulisan

Sistematika penulisan pada skripsi ini tersusun dalam lima bab. Isi setiap bab dapat diuraikan sebagai berikut:

BAB I

Bab satu menjelaskan orientasi penelitian seperti latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, dan sistematika penulisan. Lalu,

BAB II

Bab dua memaparkan dengan singkat topik yang berkaitan dengan penelitian saat ini. Selain itu, bab dua juga membahas mengenai burung puyuh, *internet of things* (IoT), mikrokontroler ESP8266, sensor DHT22, sensor suara, *relay*, lampu pijar, *firebase*, *spreadsheet*, dan *website*. Selanjutnya,

BAB III

Bab tiga menggambarkan alur pelaksanaan penelitian, komponen penelitian, metode penelitian, hingga perancangan *hardware*. Kemudian,

BAB IV

Bab empat menganalisis secara rinci perolehan data hasil penelitian sesuai dengan tetapan dan parameter yang digunakan sebagai indeks dan batasan pelaksanaan penelitian. Terakhir,

BAB V

Bab lima memberikan ⁷⁴kesimpulan dari penelitian yang telah dilakukan dan saran guna pengembangan penelitian lebih lanjut di masa mendatang.

BAB II

TINJAUAN PUSTAKA

2.1 Burung Puyuh

Burung puyuh pertama kali ditemukan di hutan Jepang. Saat ini tersebar di daratan Asia, Eropa, dan Afrika. Peternakan burung puyuh pertama kali dilakukan oleh Amerika tahun 1870. Di Indonesia budidaya burung puyuh dilakukan sejak akhir tahun 1979 hingga saat ini. Burung puyuh menghasilkan daging dan telur. Berikut gambar 2.1 menunjukkan burung puyuh.



Gambar 2.1 Burung Puyuh [18]

Burung puyuh seperti gambar 2.1 umumnya berukuran kecil, bertubuh gemuk dan bulat, memiliki paruh pendek, kaki kuat kekuningan, memakan biji-bijian, mampu berlari dan terbang dalam jarak rendah, dan berkembang biak dalam usia 6 minggu. Burung puyuh jantan berbulu cokelat dengan berat sekitar 117,60 gram. Sedangkan burung puyuh betina berwarna cokelat kemerahan dengan berat 143,30 gram. Habitat ideal burung puyuh berada pada suhu 24 s.d.30°C, kelembapan 85%, dan tingkat kebisingan kurang dari 70 dB [19][20].

Selain suhu dan kelembapan, kondisi kandang juga berperan penting sehingga burung puyuh tidak mengalami stress. Kandang harus mendapatkan cukup sinar matahari, tidak lembap, dan sirkulasi udara lancar. Kandang berukuran 60×50×30 cm dengan 10 ekor burung puyuh petelur menampilkan performa optimal [21]. Posisi Kandang tidak menempel langsung dengan tanah [22]. Lantai kandang terbuat dari kawat besi dengan kemiringan 10-15° sehingga mudah mengumpulkan telur.

2.2 Mikrokontroler

Mikrokontroler adalah sebuah komputer kecil dikemas dalam bentuk *IC* (*Integrated Circuit*) dan dirancang untuk melakukan operasi tertentu. Sebuah mikrokontroler terdiri dari satu atau lebih inti prosesor, memori serta perangkat *input* dan *output* yang dapat diprogram. Digunakan dalam berbagai perangkat seperti sistem *monitoring*, kontrol mesin mobil, hingga pengendali jarak jauh [23]. Umumnya, mikrokontroler disusun dengan Arsitektur Von Neumann dan Arsitektur Harvard.

Arsitektur Von Neumann diperkenalkan Jhon Von Neumann pada tahun 1945. Keunggulannya instruksi dan data disimpan dalam memori yang sama, bus tunggal instruksi, mampu menyimpan banyak aplikasi, dan strukturnya lebih sederhana. Kelemahannya kecepatan akses data cukup terbatas untuk instruksi dan data tidak dapat dilakukan bersamaan [24].

Arsitektur Harvard memiliki dua bus terpisah untuk membaca data dan melakukan program tertentu sehingga CPU (*Central Processing Unit*) dapat mengakses keduanya secara bersamaan. Arsitektur ini dikembangkan guna mengatasi hambatan pada Arsitektur Von Neumann. Oleh karena itu, penggunaan Arsitektur Harvard membuat kinerja komputer semakin cepat dan lebih sesuai digunakan dalam aplikasi yang membutuhkan kinerja cepat seperti pengendalian perangkat [25].

2.3 Sensor DHT22

Sensor merupakan perangkat yang mampu mendeteksi, mengukur, atau memantau suatu fenomena kemudian mengubahnya menjadi sinyal yang dapat diukur dan diinterpretasikan oleh manusia atau sistem elektronik. Sensor tersusun dari reseptor, transduser, pengkondisian sinyal, *output* dan *feedback*. Keberadaan *feedback* berfungsi meningkatkan akurasi perhitungan data, dan mengimbangi perubahan lingkungan [26]. Tampilan sensor DHT22 dapat dilihat pada gambar 2.2.



Gambar 2.2 Sensor DHT22 [27]

Sensor DHT22 seperti gambar 2.2 digunakan untuk mengukur perubahan suhu dan kelembapan bersamaan. Kapasitor dan *thermistor* akan mengukur suhu lalu menampilkan *output* sinyal pada *pin* data. Sensor DHT22 mampu mengukur suhu dalam dua parameter yakni °F atau °C dan kelembapan dalam satuan persen (%). Sensor ini memerlukan *supply* sebesar 3 s.d. 3,6 VDC. Jangkauan pengukuran suhu sebesar -40°C s.d. 80°C dengan akurasi $\pm 0,5^{\circ}\text{C}$. Kelembapan relatif sebesar 0 s.d. 100%RH dengan akurasi 2 s.d. 5%. Koefisien kalibrasi sebesar $\pm 2\%$ serta resolusi sebesar 1°C dengan tingkat *error* 1,96%. Kualitas sensor DHT22 lebih baik dibandingkan DHT11 serta sangat mudah diaplikasikan pada Arduino [28].

2.4 Sensor Max4466

Sensor suara merupakan perangkat yang mengubah gelombang *sinusoidal* suara menjadi gelombang sinyal listrik (*Alternating Sinusoidal Electric Current*). Gelombang suara yang mengenai membran membuat kumparan kecil membuat membran bergerak naik turun. Kecepatan kumparan akan menentukan besaran gelombang listrik yang dihasilkan. Salah satu komponen penyusun sensor suara adalah *condensor microphone* [29]. Mikrofon berfungsi memberikan *input* ke *buffer*, detektor puncak, dan *amplifier*. Sensor suara dapat dilihat pada gambar 2.3.



Gambar 2. 3 Sensor Max4466 [30]

Sensor Max4466 adalah sensor penguat suara sangat sederhana dan mudah digunakan dengan nilai *gain* mulai 25 s.d. 125 kali yang dapat disesuaikan dengan memutar potensiometer pada modul. Rasio penolakan catu daya sangat baik sebesar 112 dB dan rasio penolakan sebesar 126 dB. Selain itu, sensor Max4466 menghasilkan *bandwidth* $AV \geq 5$ *gain* 600 kHz dari arus sebesar 24 μ A. Dilengkapi dengan mikrofon elektret sehingga mampu menangkap audio sebesar 20 Hz s.d. 20 kHz [31].

Memiliki tiga *pin* yakni VCC, GND, dan OUT. *Pin* VCC memasok daya ke modul sensor sebesar 3 s.d. 5,5 V. *Pin* GND berperan sebagai grounding dan *pin* OUT menampilkan data keluaran digital lalu dikirim ke mikrokontroler. Sensor Max4466 dirancang agar *supply* tegangan dan nilai *gain* bersifat fleksibel dan dapat disesuaikan sehingga sesuai dalam berbagai aplikasi audio seperti pengubah suara *realtime*, perekaman sampel, hingga proyek reaktif audio menggunakan algoritma *Fast Fourier Transform* (FFT). Di bagian belakang modul terdapat potensiometer untuk menyesuaikan nilai *gain*. *Gain* menentukan besaran untuk memperkuat sinyal suara. Jika diputar berlawanan arah jarum jam (CCW) meningkatkan nilai *gain*. Sebaliknya, jika diputar searah jarum jam (CW) mengurangi nilai *gain* [32].

2.5 Lampu Pijar

Lampu pijar tersusun dari bola lampu, filamen, kaki lampu, dan gas inert (gas yang secara kimiawi tidak reaktif dan tidak mudah terbakar). Bola lampu terbuat dari kaca guna melindungi filamen dari oksidasi. Filamen terbuat dari tungsten dan menjadi sangat panas ketika dialiri arus listrik. Kaki menghubungkan lampu pijar ke soket atau *fiting* lampu. Gas inert berupa nitrogen atau argon untuk mencegah proses oksidasi yang akan merusak filamen lebih cepat [33]. Berikut gambar 2.4 menunjukkan lampu pijar.



Gambar 2. 4 Lampu Pijar [34]

Lampu pijar menyalurkan arus listrik melalui filamen yang memanaskan dan menghasilkan cahaya. Keunggulan lampu pijar antara lain menghasilkan panas sehingga suhu lingkungan di sekitar meningkat, teknologi pembuatan sangat sederhana, menghasilkan cahaya sangat mirip dengan cahaya alami, murah, dan mudah ditemukan dalam berbagai macam bentuk dan tegangan mulai dari 1,25 s.d. 300 V.

2.6 ³⁶ Internet of Things

¹⁰⁷ *Internet of Things (IoT)* didefinisikan sebagai konsep di mana sensor hingga perangkat elektronik saling terhubung dan berkomunikasi melalui internet. Hal ini membuat perangkat dapat memantau kondisi tertentu, mengolah data, mencari informasi, hingga mengambil keputusan secara otomatis [35]. Komponen penting IoT diantaranya perangkat, koneksi, *gateway*, dan *cloud*. Perangkat IoT akan mengumpulkan dan melakukan pertukaran data serta saling berinteraksi melalui jaringan internet. Koneksi seperti WiFi dan *bluetooth*, dan jaringan lainnya membuat seluruh perangkat saling terhubung sehingga dapat mengirim dan menerima data. *Gateway* bertindak sebagai perantara jaringan lokal dengan suatu *platform cloud* seperti *firebase*. Selain itu, *gateway* juga mengelola keamanan, manajemen data, dan terjemahan protokol sistem.

Pemrosesan data memerlukan *mobile device* atau komputer yang akan menjalankan aplikasi seperti Android, iOS, atau *web* sehingga dapat berinteraksi dengan perangkat IoT. Setelah ditransmisikan, data dikirim dan disimpan di *cloud* secara terpusat sehingga dapat diakses kapan saja dan dimana saja [36]. Manfaat penggunaan IoT diantaranya efisiensi energi dan biaya, produktivitas meningkat, memberikan kenyamanan, dan mempermudah pengambilan keputusan berbasis data. Oleh karena itu, penggunaan IoT mulai banyak digunakan dalam berbagai sektor seperti *smart home*, *health monitoring*, *smart farming*, manufaktur dan manufaktur hingga *smart agriculture*.

2.7 ¹⁵ Database

Database merupakan kumpulan data yang disimpan dengan sistem tertentu dan saling berhubungan sehingga dapat dikelola dan diakses dengan mudah. Keberadaan *database* sangat penting dalam mengelola data dalam jumlah besar dan terus bertambah. Komponen utama *database* yakni data, *hardware*, sistem operasi, *database management system (DBMS)* dan *database access language*. Data berisi informasi seperti teks, log, hingga gambar serta memiliki struktur tertentu sehingga data mudah diakses. *Hardware device* berfungsi menyimpan dan mengelola data seperti komputer, *disk*, memori, dan *server hosting (online)*.

118
Sistem operasi bertanggung jawab terhadap seluruh sistem yang ada di dalam server komputer. Kemudian *database management system (DBMS)* adalah sebuah aplikasi sistem basis data. *DBMS* mempermudah *user* ketika akan memasukkan dan memperbarui data [37]. Selain itu, *DBMS* juga mengelola dan mengontrol akses *database* guna memastikan integritas, keamanan, dan kinerja maksimum [38]. Terakhir *database access language* merupakan bahasa yang digunakan ketika menuliskan perintah dalam sebuah *database*. Manfaat menggunakan *database* yakni efisiensi pengelolaan data, akses *multi user*, keamanan data, kontrol data secara terpusat, dan kemudahan dalam membuat aplikasi.

2.8 Website

35
103
Website dapat diartikan sebagai halaman yang dapat diakses melalui internet dan dikelola oleh individu atau organisasi. Unsur utama *website* antara lain *domain*, *hosting*, pemrograman, konten dan tampilan. Berdasarkan fungsinya, *website* terdiri dari *website* dinamis dan *website* statis. *Website* dinamis menyediakan konten yang selalu berubah. Sedangkan, *website* statis memiliki konten yang jarang diubah [39].

Website memiliki dua segi pemrograman yakni *server side* dan *client side*. *Server side* mengolah data menggunakan bahasa pemrograman seperti PHP, ASP atau Python sehingga tidak ditampilkan di halaman *user*. *Client side* yakni *website* yang tidak memerlukan *server* dalam menjalankannya dan cukup diakses melalui browser seperti Google Chrome. Selain sebagai sumber informasi, *website* juga berfungsi sebagai sarana komunikasi, hiburan dan transaksi.

2.9 Firebase

56
Firebase merupakan layanan google yang memberikan kemudahan bagi para *developer* aplikasi dalam mengembangkan aplikasi. Dengan *firebase*, *developer* dapat mengembangkan aplikasi tanpa harus fokus dengan *backend*. Sebagai penyedia *firebase* mampu mengatur hal-hal mengenai *database*, *authentication*, *hosting* hingga API. Secara umum, *firebase* memiliki dua layanan yakni *spark* dapat diakses secara gratis dan *blaze* layanan berbayar [40].

Fitur *firebase* cukup beragam di antaranya *analytics*, autentikasi, *cloud messaging*, *test lab*, dan *firebase realtime database*. *Firebase realtime database* menyimpan dan menyinkronkan data *user*. *Analytics* menyajikan data perilaku pengguna untuk meningkatkan performa dan pemasaran aplikasi. Autentikasi membangun sistem aman dan mudah guna mempermudah pengalaman *login* bagi *user*. *Cloud messaging* berupa *tools* untuk mengirim pesan sehingga memungkinkan mengirim pesan di iOS, Android, dan Web. *Test lab* berfungsi untuk menguji aplikasi *cloud* [41].

Seperti layanan lainnya, *firebase* memiliki keunggulan dan kekurangan. Keunggulan *firebase* memiliki versi gratis, cepat responsif, menggunakan JASON, *user friendly*, dan mudah digunakan di berbagai Android, iOS, Java, Node.js, Javascript. Kelemahan *firebase* yakni adanya keterbatasan dalam versi gratis, sulit *recovery* akun, dan sulit digunakan bila memiliki banyak *user*.

2.10 Perhitungan Nilai Error

Sistem *monitoring* otomatis memerlukan sistem yang dapat mengirimkan dan menampilkan data secara akurat. Nilai *error* atau galat menunjukkan perbedaan pada hasil pengukuran sensor dengan referensi (nilai sebenarnya). *Error* disebabkan berbagai faktor yakni gangguan lingkungan, kalibrasi sensor yang kurang tepat, atau keterbatasan *hardware*. Oleh karena itu, dilakukan perhitungan nilai *error* performa sistem guna memastikan keakuratan data yang diperoleh.

Metode yang dapat digunakan dalam menghitung nilai *error* diantaranya *absolute error*, *percentage relative error*, *mean absolute error* (MAE), *mean squared error* (MSE), dan *root mean squared error* (RMSE). Pemilihan metode perhitungan *error* disesuaikan dengan karakteristik data yang diperoleh dan tujuan analisis.

Percentage relative error umumnya digunakan dalam proses kalibrasi alat ukur. Kesalahan relatif mengukur keakuratan suatu pengukuran dan dinyatakan dalam bentuk desimal [42]. Umumnya metode ini digunakan dalam kalibrasi sensor, jumlah data yang dikalibrasi tidak terlalu banyak, hasil perhitungan

dinyatakan dalam bentuk persen, dan memiliki nilai referensi sebagai acuan. Persamaan *percentage error* sebagai berikut:

$$\% \text{ error} = \left| \frac{\text{Data Sensor (X)} - \text{Data Referensi (Y)}}{\text{Data Referensi (Y)}} \right| \times 100\% \quad (2.1)$$

Absolute error merupakan selisih antara nilai hasil pengukuran pada aplikasi pembandingan dengan nilai sesungguhnya dari sensor. Metode ini digunakan ketika data aplikasi dan sensor dinyatakan dalam satuan yang sama. Nilai *absolute error* selalu bernilai positif. *Absolute error* dapat dihitung dengan persamaan:

$$\text{Absolute Error} = |\text{Data Aplikasi} - \text{Data Sensor}| \quad (2.2)$$

Mean absolute error (MAE) diperoleh dari rata-rata *absolute error*. Nilai MEA menilai performa sistem secara keseluruhan berdasarkan perolehan data yang telah tersedia. Berikut persamaan MAE:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (2.3)$$

Nilai *error* kecil menandakan bahwa sensor yang digunakan cukup akurat dan mendekati alat acuan. Dalam sistem *monitoring* menandakan sistem yang dibuat sudah akurat dan mendekati kondisi *realtime*. Sebaliknya, *error* tinggi perlu dilakukan perbaikan pada aspek sensor, kalibrasi, atau algoritma pengolahan data. Oleh karena itu, perhitungan *error* menjadi indikator kunci dalam menilai kualitas dari sistem *monitoring* otomatis yang dibuat.

2.11 Tinjauan Pustaka

Penelitian ini merujuk pada penelitian terdahulu mencakup aspek penelitian baik secara landasan teori, metode penelitian yang digunakan hingga parameter penelitian. Berikut ini lima penelitian terdahulu yang menjadi landasan penelitian yang sedang dilakukan, yakni:

Penelitian pertama, sistem inkubator telur puyuh dengan mengontrol suhu dan kelembapan menggunakan DHT11 sehingga pemantauan jarak jauh secara *realtime* dan otomatis [5][6]. *Monitoring* yang tidak konsisten membuat suhu kandang meningkat membuat konsumsi pakan dan produktivitas burung puyuh menurun [7].

Penelitian kedua menyatakan bahwa kualitas telur puyuh sangat dipengaruhi oleh suhu dan kelembapan dalam kandang [8]. Untuk itu, dikembangkan aplikasi *monitoring* yang terintegrasi agar meningkatkan proses pemeliharaan secara cepat[9]. Penelitian ketiga mengenai implelementasi sistem *monitoring* juga pernah dilaksanakan menggunakan platform Node-Red dan metode Naïve Bayes mencapai akurasi 80% [10]. Kemudian dikembangkan sistem kontrol dan *monitoring* berbasis IoT menggunakan sensor DHT22 dan sensor MQ135 yang bertujuan menjaga kondisi kandang ayam tetap optimal [11].

⁸ Penelitian keempat mengembangkan sistem otomatisasi pakan minum dan *monitoring* kelembapan udara pada kandang ayam berbasis IoT menggunakan sensor DHT22 dan mikrokontroler untuk memantau dan mengontrol kondisi lingkungan dalam kandang, yang dapat diadaptasi untuk kandang burung puyuh guna meningkatkan efisiensi produksi [14].

⁷⁶ Penelitian kelima mengenai sistem pemantauan kualitas udara berbasis IoT di *peternakan* dengan parameter suhu, kelembaban, dan kadar gas amonia dalam kandang menjaga kesehatan dan produktivitas burung puyuh [16]. Pengembangan sistem berbasis IoT terbukti mengurangi stres panas yang menyebabkan penurunan produktivitasburung puyuh [17].

BAB III

METODOLOGI PENELITIAN

3.1 Alur Penelitian

Penelitian ini memiliki beberapa tahapan untuk mencapai hasil yang diharapkan di antaranya:

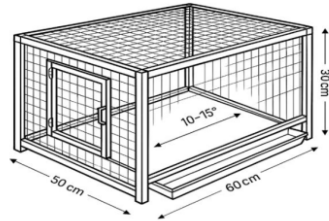
1. Studi literatur mencari referensi dari penelitian terdahulu yang berkaitan tentang karakter lingkungan berupa suhu, kelembapan, dan tingkat kebisingan ideal kandang burung puyuh petelur.
2. Menentukan komponen yang akan digunakan dalam proses pembuatan alat *monitoring* otomatis kandang burung puyuh petelur baik komponen *hardware* atau *software*.
3. Melakukan instalasi komponen yang telah direncanakan sebelumnya.
4. Membuat *listing* program untuk membaca besaran nilai dari sensor dan membuat logika guna *monitoring* suhu kandang menggunakan Bahasa C++.
5. Membuat *listing* program *website* untuk *monitoring* kandang dari jarak jauh.
6. Menghubungkan perangkat NodeMCU ke *firebase* untuk mengirim data secara *realtime* menggunakan library *FirebasseESP8266*.
7. Melakukan uji coba alat dan akurasi dari pembacaan setiap sensor yang digunakan dalam penelitian ini.
8. Menganalisis hasil dari pengambilan data dan menyimpulkan hasil dan saran untuk penelitian selanjutnya.

3.2 Komponen Penelitian

Komponen penelitian yang digunakan dalam penelitian ini sebagai berikut:

1. Kandang Burung

Kandang burung merupakan tempat tinggal bagi hewan dan terlindung dari cuaca buruk. Tampilan kandang burung dapat dilihat pada gambar 3.1.

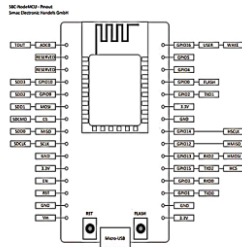


Gambar 3.1 Kandang Burung

Idealnya kandang burung puyuh berukuran 60×50×30 cm dengan kemiringan 10 s.d. 15° sehingga mudah menggelindingkan telur. Kandang harus memiliki ventilasi yang baik sehingga suhu dan kelembapan kandang tetap terjaga.

2. Mikrokontroler ESP8266

Komponen kedua dalam penelitian ini yakni mikrokontroler ESP8266. Berikut gambar 3.2 menampilkan mikrokontroler ESP8266.



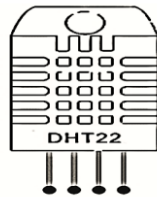
Mikrokontroler ESP8266 pada gambar 3.1 dapat mengirim data jarak jauh menggunakan WiFi secara *realtime*, berukuran kecil, konsumsi daya rendah, dapat diprogram menggunakan Arduino IDE sehingga mudah melakukan pengembangan, harga terjangkau dan instalasi mudah membuat pengguna tertarik menggunakan ESP8266 sebagai komponen pendukung dalam penelitian ini. Berikut *pin* ESP8266:

Tabel 3.1 *Pin* ESP8266

No.	PIN	FUNGSI
1.	Vin	Tegangan Masukan Dari <i>Power Supply</i> Eksternal 5V
2.	3V	Tegangan Masukan 3.3V
3.	GND	Ground
4.	Rx / Tx	Jalur Komunikasi Serial <i>Transmitter- Receiver</i>
5.	ADC0	Terminal <i>Input</i> Analog
6.	GPIO – GPIO16	Terminal <i>Input</i> Dan <i>Output</i> Analog Maupun Digital
7.	Miso	Jalur <i>Download</i> Untuk Menerima Data
8.	Mosi	Jalur <i>Download</i> Untuk Mengirim Data
9	RST	Jalur Reset ESP8266

3. Sensor DHT22

Komponen ketiga dalam penelitian ini sensor DHT22. Berikut gambar 3.3 menunjukkan sensor DHT22.



Gambar 3.3 Sensor DHT22

Sensor DHT22 seperti pada gambar 3.3 lebih mudah digunakan karena satu modul sudah mencakup dua sensor yakni sensor suhu dan sensor kelembapan. Tingkat akurasi dan rentang pengukuran sensor DHT22 lebih luas dibandingkan DHT11. Sensor DHT22 mampu mengukur suhu dari -40° s.d. 80°C dengan tingkat

akurasi sekitar $\pm 0,5^{\circ}\text{C}$ dan kelembapan dengan rentang 0 s.d. 100% RH dengan akurasi ± 2 s.d. 5%.

Sementara itu, DHT11 hanya mampu mengukur suhu 0 s.d. 50°C dengan akurasi $\pm 2^{\circ}\text{C}$ dan kelembapan 20 s.d. 80% RH dengan akurasi $\pm 5\%$. Walaupun harga DHT22 lebih mahal dan waktu respons lebih lambat dibandingkan DHT11, namun tingkat akurasi dan rentang pengukuran lebih luas membuat DHT22 lebih cocok untuk aplikasi yang membutuhkan presisi dan ketelitian tinggi.

Tabel 3.2 *Pin* Sensor DHT22

NO	PIN	FUNGSI
1.	VCC	Tegangan Masukan dari power supply eksternal 3.3V atau 5V
2.	Data	Komunikasi data sensor DHT22 dengan mikrokontroler
3.	GND	Ground sensor DHT22

4. Sensor Max4466

Komponen keempat dalam penelitian ini sensor Max4466. Berikut gambar 3.4 menunjukkan sensor Max4466



Gambar 3.4 Sensor Max4466

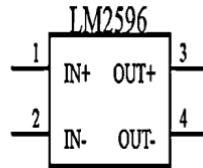
Sensor Max4466 pada gambar 3.4 merupakan sensor penguat suara yang sangat sederhana dan mudah digunakan. Memiliki 3 *pin* yakni VCC, GND, dan OUT. Sensor Max4466 dirancang agar *supply* tegangan dan nilai *gain* bersifat fleksibel sehingga cocok dalam berbagai aplikasi audio seperti pengubah suara *realtime*, perekaman sampel, hingga proyek reaktif audio menggunakan algoritma *Fast Fourier Transform* (FFT). Berikut tabel 3.3 mengenai *pin* sensor Max4466.

Tabel 3.3 *Pin* Sensor Max4466

No.	PIN	FUNGSI
1.	VCC	Tegangan masukan dari power supply sebesar 3.3V atau 5V
2.	GND	Ground sensor
3.	OUT	Menghasilkan <i>output</i> sinyal suara yang sudah diperkuat

5. Step-Down LM2596

Komponen kelima penelitian yaitu Step-Down LM2596 yang dapat dilihat pada gambar 3.5



Gambar 3.5 Step-Down LM2596

LM2596 dalam gambar 3.4 merupakan modul regulator tegangan *step-down* (*buck converter*) yang digunakan untuk menurunkan *input voltage* menjadi besaran tegangan yang lebih rendah secara efisien. Komponen ini dirancang khusus untuk menyediakan daya stabil dan efisiensi tinggi hingga 92% sehingga ideal untuk aplikasi yang membutuhkan penghematan daya. LM2596 dapat menerima *input* s.d. 40V dan mengatur *output* tegangan yang diinginkan mulai dari 1,23V s.d. 37V dengan arus *output* hingga 3A. Dilengkapi potensiometer sehingga tegangan *output* dapat diatur sesuai kebutuhan. Berikut *pin step down* LM2596

Tabel 3.4 *Pin* Step Down LM2596

No.	PIN	FUNGSI
1.	IN (+)	Tegangan masukan dari sumber daya eksternal
2.	IN (-)	Ground tegangan masukan eksternal
3.	OUT (+)	Tegangan ouput positif yang telah diturunkan
4.	OUT (-)	Ground tegangan <i>output</i>

6. Lampu pijar

Komponen keenam dalam penelitian ini yaitu lampu pijar. Berikut tampilan lampu pijar pada gambar 3.6.



Gambar 3.6 Lampu Pijar

Lampu pijar seperti gambar 3.6 digunakan sebagai penghangat suhu kandang. Tergolong dalam jenis lampu yang menghasilkan cahaya dengan memanaskan filamen logam hingga berpijar akibat aliran listrik. Ketika listrik mengalir melalui filamen, resistansi logam tersebut menyebabkan panas yang sangat tinggi, sehingga filamen berpijar dan memancarkan cahaya yang terlihat. Berikut fungsi terminal pada lampu pijar

Tabel 3.5 Terminal Lampu Pijar

No.	PIN	FUNGSI
1.	Terminal Positif (+)	Dihubungkan dengan sumber tegangan positif
2.	Terminal Negatif (-)	Dihubungkan dengan ground

7. Firebase Realtime Database

Firebase Realtime Database merupakan layanan data berbasis *cloud* yang disediakan oleh *firebase* yang memungkinkan penyimpanan dan sinkronisasi data secara *realtime* untuk aplikasi web dan seluler. Basis data ini bersifat NoSQL di mana data disimpan dalam format JSON sehingga memungkinkan aplikasi membaca dan menulis data secara langsung ke server Firebase. Firebase Realtime Database sangat cocok untuk aplikasi yang membutuhkan data secara langsung pada beberapa perangkat seperti aplikasi obrolan, pelacakan lokasi, atau aplikasi IoT.

8. ⁶³ Arduino IDE

Arduino IDE (*Integrated Development Environment*) merupakan perangkat lunak yang digunakan untuk mengembangkan dan memprogram perangkat berbasis Arduino. Pada penelitian ini, Arduino IDE digunakan untuk menyusun logika program *hardware* di ESP8266.

9. Visual Studio Code

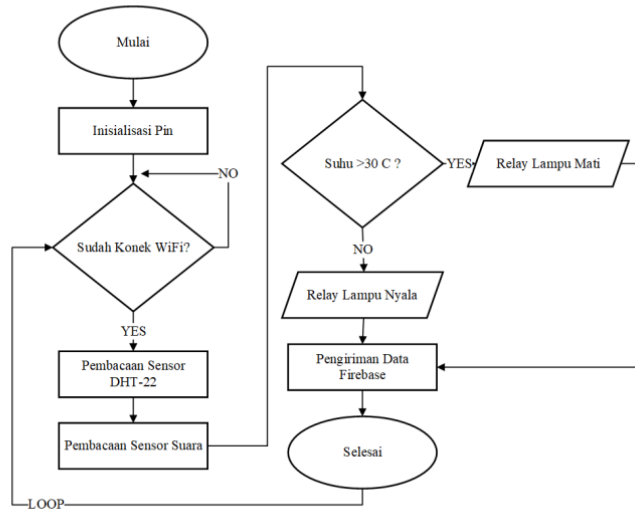
Pada penelitian ini penulis menggunakan Visual Studio Code untuk membuat halaman *website*. ¹²² Visual Studio Code (VS Code) adalah editor kode sumber yang dikembangkan oleh Microsoft, dirancang untuk mendukung berbagai bahasa pemrograman dan alat pengembangan. VS Code dikenal karena tampilannya yang sederhana namun sangat kuat, dengan berbagai fitur yang membantu pengembang dalam menulis, mengedit, dan debug kode secara efisien.

⁴ 3.3 Metode Penelitian

Metode penelitian yang dilakukan meliputi perancangan blok sistem untuk mengetahui proses atau alur kerja dari sistem, skematik untuk melihat bagaimana bentuk alat jika sudah dirangkai dan saling terhubung, diagram koneksi antar mikrokontroler dengan *database firebase*, dan diagram alir cara kerja sistem berdasarkan algoritma yang dipakai dalam menyelesaikan tugas yang diberikan. Penelitian ini bertujuan untuk melihat bagaimana proses rancang bangun *monitoring* otomatis kandang burung puyuh berbasis *internet of things*.

3.3.1 Diagram Alir

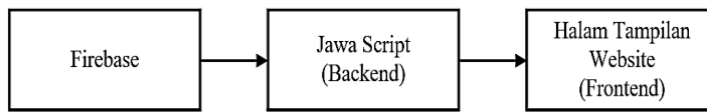
Diagram alir penelitian ini menyajikan rangkaian langkah-langkah yang dijalankan dalam program ESP8266. Diagram alir penelitian dapat dilihat pada gambar 3.6.



Gambar 3.7 Diagram Alir Program IoT

Pada gambar 3.6 dapat dilihat diagram alir cara kerja alat. Proses dimulai dengan inisialisasi *pin* analog sensor dan *relay* untuk mengendalikan lampu pijar, kemudian mengkoneksikan perangkat ESP8266 dengan WiFi. Jika sudah terhubung, sensor DHT-22 akan membaca nilai suhu dan kelembapan, sedangkan sensor Max4466 akan membaca kebisingan secara *realtime*.

Jika suhunya lebih besar dari 30°C maka *relay* lampu pijar mati, namun jika suhunya kurang dari 30°C maka lampu pijar akan menyala untuk menaikkan suhu kandang. Terakhir, data pengukuran berupa suhu, kelembapan, dan kebisingan akan di kirim ke *database firebase* untuk kemudian ditampilkan pada halaman *website*.

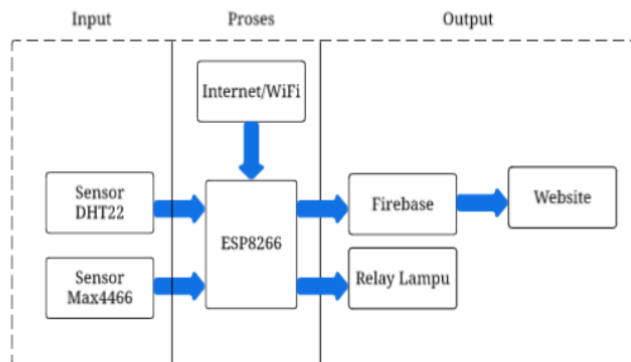


Gambar 3.8 Diagram Alir *Website*

Selain itu, terdapat perancangan algoritma frontend dan backend website. Pada perancangan algoritma *frontend* atau tampilan halaman monitoring menggunakan bahasa pemrograman HTML, selain itu juga menggunakan CSS agar tampilan *website* semakin menarik. Sedangkan untuk program *backend* yang akan bertindak menarik data dari *database firebase* menggunakan bahasa pemrograman Java Script proses tersebut dapat dilihat pada Gambar 3.8.

3.3.2 Diagram Blok

Diagram blok penelitian ini menyajikan rangkaian tahapan yang dilakukan selama penelitian. Diagram blok penelitian dapat dilihat pada gambar 3.8.



Gambar 3.9 Diagram Blok

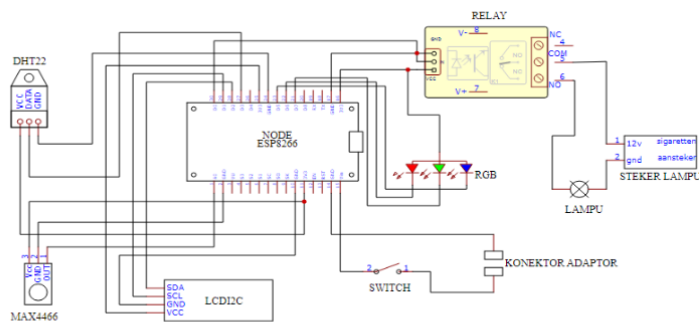
Penjelasan dari alur diagram blok Gambar 3.9 sebagai berikut:

1. Sensor DHT22 dan sensor Max4466 akan membaca besaran nilai suhu, kelembapan, dan kebisingan.

2. Mikrokontroler ESP8266 yang terkoneksi ke jaringan WiFi mengelola semua data *input* dari sensor untuk dikirim ke *firebase* secara *realtime* setiap ada perubahan nilai pada masing-masing besaran.
3. Selain itu, mikrokontroler ESP8266 juga mengontrol lampu pijar sebagai penjaga suhu kandang secara otomatis menggunakan logika if else.
4. Terakhir data dari *firebase* di tampilkan ke halaman *website* yang telah dibuat.

3.4 Perancangan Hardware

Perancangan *hardware* memberikan gambaran skematik rangkaian *hardware* yang akan dibuat seperti sensor, mikrokontroler, dan aktuator akan integrasikan dengan kabel *wiring*.



Gambar 3.10 Skematik Hardware

Gambar 3.9 menampilkan rancangan *hardware* dari sistem IoT yang dibuat untuk melakukan *monitoring*. Secara singkat alur pembacaan data dimulai dengan ESP8266 yang terkoneksi dengan jaringan WiFi akan menghubungkan seluruh komponen dan mengirimkan data ke server atau aplikasi. Kemudian sensor DHT22 dan Max4466 melakukan pengukuran sesuai parameter. Hasil pembacaan sensor

dikirimkan ke ESP8266 yang telah terkoneksi dengan WiFi. Lalu hasil pembacaan ditampilkan secara langsung pada LCD I2C.

Jika hasil pembacaan suhu di bawah rentang ideal, maka *relay* berfungsi sebagai saklar di mana secara otomatis menyalakan lampu guna menghangatkan kandang dan mematikan lampu pijar jika suhu kandang sudah ideal. Penggunaan LED berfungsi hanya sebagai indikator kondisi kandang. Konektor untuk menyediakan daya yang dibutuhkan oleh seluruh komponen. Secara singkat, fungsi beberapa komponen utama yang digunakan untuk merancang sistem perangkat keras dirangkum pada tabel di bawah.

Tabel 3.6 Fungsi Komponen *Hardware*

NO	NAMA	FUNGSI
1.	ESP8266	Melaksanakan perintah yang telah dibuat dalam bentuk program bahasa C++
2.	DHT-22	Sensor yang membaca kelembapan & suhu
3.	Max4466	Sensor suara untuk deteksi kebisingan
4.	Relay	Sebagai saklar elektronis lampu pijar
5.	Lampu Pijar	Sebagai lampu penghangat suhu
6.	LCD I2C	Menampilkan data monitoring
7.	Konektor Adaptor	Lubang konektor untuk adaptor Male

HASIL DAN ANALISIS

4.1 Lokasi Pengambilan Data

Penelitian diawali dengan menentukan lokasi ideal kegiatan *monitoring*. Hal ini perlu dilakukan dikarenakan burung puyuh tergolong hewan sensitif sehingga memerlukan lokasi yang tepat agar penelitian dan pengambilan data dapat berjalan dengan lancar. Kegiatan penelitian dalam pengambilan data dilakukan di Daru, Kecamatan Jambe, Kabupaten Tangerang.

Objek penelitian menggunakan empat ekor burung puyuh terdiri dua ekor burung puyuh jantan dan dua ekor burung puyuh betina yang ditempatkan dalam sebuah *prototype* sederhana berupa keranjang *box* berukuran 36cm×27cm×26 cm.



Gambar 4.1 Kandang *Box*

Kegiatan *monitoring* suhu, kelembapan, dan tingkat kebisingan pada kandang dilakukan secara *realtime*. Seluruh komponen penelitian disusun rapi dalam *casing box* yang diletakkan di atas keranjang dan dihubungkan ke sumber daya. Keranjang *box* digunakan sebagai *prototype* dari kandang burung puyuh. Tampilan *casing box* dan penataan alat dapat dilihat pada gambar 4.2.



Gambar 4.2 Kondisi *Prototype*

Gambar 4.2 menunjukkan penataan *casing box* dan lampu pijar didalam keranjang. Untuk dapat melakukan *monitoring* dan pengendalian *realtime*, perangkat harus tetap terhubung dengan sumber daya. Setelah itu, data pengukuran dikirimkan ke mikrokontroler ESP8266. Mikrokontroler mengolah data tersebut dan mengirimkannya ke *firebase* melalui jaringan WiFi. Kemudian, data tersebut ditampilkan melalui *website* sehingga dapat dilakukan *monitoring* jarak jauh. Mikrokontroler ESP8266 juga mengendalikan lampu pijar sehingga dapat menyala atau mati otomatis sesuai dengan parameter yang terdeteksi oleh sensor. Hasil

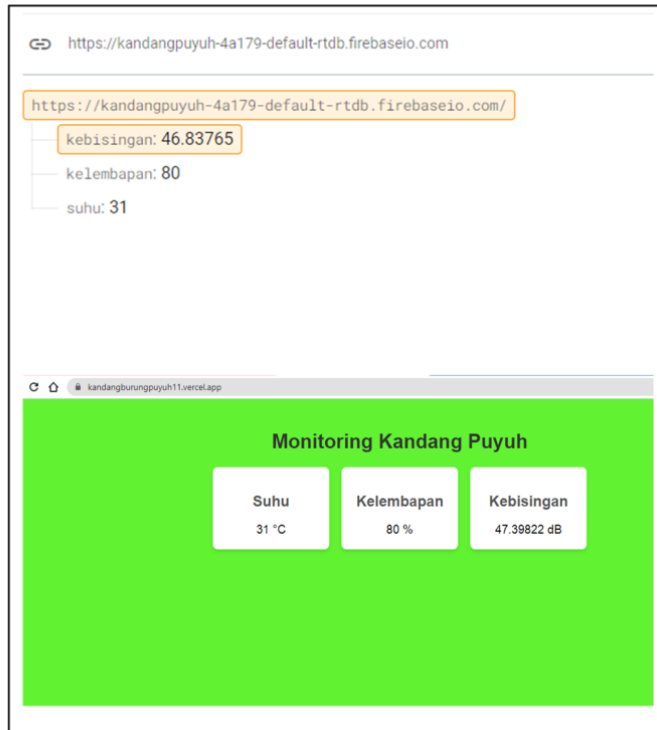
pengukuran *website* terus diperbarui *realtime* selama sensor terhubung dengan sumber daya.

Selama monitoring *realtime* kandang mengalami kondisi ideal dan tidak ideal. Kandang dikatakan ideal ketika seluruh parameter pengukuran berada dalam rentang pengukuran. Sebaliknya, jika salah satu parameter tidak dalam rentang pengukuran. Ketika ideal ditandai dengan LED berwarna hijau sedangkan kondisi tidak ideal ditandai dengan LED berwarna merah. Kondisi ideal dan tidak ideal ditunjukkan pada gambar 4.3.



Gambar 4.3 Kondisi Ideal dan Tidak Ideal

Mikrokontroler ESP8266 akan memproses hasil pengukuran *realtime* sensor DHT22 dan Max4666. Hasil pengukuran dikirimkan ke *website* dan *firebase*. Berikut gambar 4.4 menampilkan hasil pengukuran *realtime* pada *website* dan *firebase*.



Gambar 4.4 Tampilan Hasil Pengukuran Pada *Website* dan *Firebase*

Pembaruan hasil pengukuran halaman *website* mempermudah dilakukannya *monitoring realtime*. Jika suhu kandang terlalu tinggi dan kelembapan tidak ideal, alat memberikan peringatan. Hal ini dilakukan tidak hanya untuk meningkatkan produktivitas telur, tetapi juga meminimalkan resiko penyakit yang dapat muncul akibat kondisi kandang tidak terkontrol. Dengan *monitoring* maka pengelolaan kandang menjadi lebih efisien dan memberikan hasil optimal.

4.2 Analisis Listing Code

Penelitian berfokus kepada *monitoring* kandang burung puyuh sehingga memerlukan alur penelitian yang runtut. Secara singkat, rancangan perangkat lunak menggunakan tiga komponen utama yakni *firebase*, *backend*, dan *fronted*. Pertama, *firebase* mengumpulkan dan menyimpan seluruh data yang diperoleh melalui sensor DHT22 dan Max4466. *Backend* memproses dan mengolah data menggunakan JavaScript kemudian menghubungkan *firebase* dengan *fronted*. Terakhir data ditampilkan pada halaman *website* (*frontend*) sehingga dapat dilakukan monitoring secara *realtime*

Adapun alur penelitian diawali dengan insialisasi dan konfigurasi perangkat, menghubungkan koneksi WiFi, mengatur tampilan pembacaan sensor, mengolah dan mengirimkan data, mengontrol logika berdasarkan data, dan menyusun fungsi loop utama. Berikut penjelasan setiap tahapan alur penelitian yang dilakukan:

4.2.1 Insialisasi dan Konfigurasi

```
#include "DHT.h"
DHT dht(D3, DHT22);
#include <ESP8266WiFi.h>
#include <FirebaseESP8266.h>
#include <ESP8266WebServer.h>
ESP8266WebServer server(80);
#include <WiFiManager.h>
#include <WiFiClientSecure.h>
#include <LCD_I2C.h>
LCD_I2C lcd(0x27, 16, 2); // Default address of most PCF8574
modules
```

Gambar 4.5 Inisialisasi *Library*

Pertama menginisialisasi *library* dan konfigurasi perangkat bertujuan mempersiapkan fungsi sistem, mengatur komunikasi *software* dan *hardware*, mengoptimalkan kinerja sistem, menghindari *error* dan memberikan akses ke fitur khusus. Inisialisasi dilakukan untuk *impor library* dan objek yang digunakan.

Coding di atas melakukan inisialisasi terhadap seluruh *library* yang akan digunakan dan menginstruksikan *mikrokontroler* ESP8266 untuk membaca data dari sensor suhu DHT22. Untuk menginisialisasi *library*, maka sensor DHT22

dihubungkan dengan *pin* D3 ESP8266. Selanjutnya, menghubungkan ESP8266 dengan jaringan WiFi lalu mengirim data pengukuran sensor kepada *firebase*. Lalu, *server web* berinteraksi dan menampilkan hasil pengukuran pada layar LCD.

```
// Konfigurasi Firebase
#define FIREBASE_HOST "https://kandangpuyuh-4a179-default-
rtdb.firebaseio.com/" //
Ganti dengan URL database Firebase Anda
#define FIREBASE_AUTH
"AIZAxyDJxyKnqAvivSFk53D700adZhrMqpFDR9E"//
Ganti dengan Authentication Key Firebase Anda
const char* host = "script.google.com";
const int httpsPort = 443;
WiFiClientSecure client;
String GAS_ID =
"AKfycbxWULS2oDQBdnBNLzcp6YULJ30WvgYRxnXgMaDcTYS6miyGk-
sn57XpTckJWJB2_6w"; //--> spreads
FirebaseData firebaseData; // Objek Firebase
```

Gambar 4.6 Konfigurasi *Firebase* dan Google Sheet

Mikrokontroler ESP8266 memerlukan koneksi dengan beberapa layanan lainnya seperti *firebase realtime database* dan *google apps script*. Konfigurasi *firebase* diawali dengan mendefinisikan URL *host firebase realtime database* yang digunakan dalam berkomunikasi. *Firebase_host* merujuk alamat *firebase* server. URL dibuat untuk menghubungkan aplikasi dengan *database* sehingga harus sesuai dengan *address firebase*. *Firebase_Auth* mengautentikasi ketika mengakses *database*. Lalu token memberikan akses komunikasi *firebase* terhadap *database* sesuai izin yang ditentukan dan dihasilkan di halaman *firebase console*.

Script digunakan untuk berkomunikasi dengan Google Apps Script melalui HTTPS. *WiFiClientSecure client* akan mendeklarasikan objek sehingga koneksi dengan SSL/TLS ke *server*. *Gas_ID* digunakan untuk mengirim atau mengambil data dari Google Apps Script melalui API. *FirebaseData* berinteraksi dengan *Firebase Relatime Database* untuk mengakses data di *firebase*.

```

uint8_t _merah = D6;
uint8_t _hijau = D7;
uint8_t _biru = D5;
int relay_lampu = D0;
const int micPin = A0; // Pin analog yang terhubung ke sensor
mikrofon
int sampleWindow = 50; // Jendela waktu sampling dalam
milidetik
unsigned long lastTime = 0;
unsigned long timerDelay = 10*60*1000;
float suhu, kelembapan, kebisingan;

```

Gambar 4.7 Inisialisasi *Pin* Sensor

Kode di atas menghubungkan beberapa sensor dan perangkat untuk memantau dan mengontrol kondisi lingkungan. *Int* merupakan tipe data untuk menyimpan nilai bilangan bulat 8 bit (0 s.d. 255). Biasanya digunakan untuk menyimpan nilai kecil. *Pin* LED RGB mengontrol warna LED dan sebagai status atau *feedback visual system*. *Relay* lampu bertipe data *int* dan diatur ke *pin* D0 untuk mengontrol *relay* yang menghidupkan atau mematikan lampu. *Micpin* mendeteksi nilai dari sensor mikrofon dan di set ke *pin* A0.

Fungsi *sampleWindow* menunjukkan waktu melakukan sampling data sensor mikrofon setiap 50ms. Variabel *lastTime* menyimpan waktu terakhir terjadinya suatu peristiwa dan digunakan untuk timer interval tertentu dalam program. *timerDelay* untuk menunda peristiwa dan menjalankan suatu peristiwa setelah periode tertentu. *Float* *suhu*, *kelembapan*, dan *kebisingan*; menyimpan nilai berbagai sensor yang terhubung ke mikrokontroler.

4.2.2 Koneksi dan Pengaturan WiFi

Setelah inisialisasi dan konfigurasi, perangkat dihubungkan secara otomatis ke jaringan WiFi menggunakan `wiFiManager`. Jika perangkat belum terhubung, maka portal konfigurasi WiFi akan terbuka.

```
void handleRoot() {  
  server.send(200, "text/plain", "Hello from WiFiManager!");  
}  
  
void connectWiFi() {  
  WiFiManager wifiManager; // Start WiFiManager configuration  
  portal  
  if (!wifiManager.autoConnect("KANDANG PUYUH")) {  
    Serial.println("Failed to connect and hit timeout");  
    delay(3000);  
    // Reset and try again, or maybe put it to deep sleep  
    ESP.reset();  
    delay(5000);  
  }  
  // Route for root  
  server.on("/", handleRoot);  
  // Start server  
  server.begin();  
  lcd.clear();  
  lcd.setCursor(0, 0);  
  lcd.print("TERHUBUNG");  
  delay(1500);  
  lcd.clear();  
}
```

Gambar 4.8 Listing Code Koneksi WiFi

Setelah inisialisasi, koneksi WiFi dibagi menjadi dua fungsi yakni fungsi *handle root* dan fungsi *connect WiFi*. Pertama fungsi `handleRoot()` merupakan fungsi yang dijalankan ketika server menerima *request* pada *root* URL ("/). Didalamnya terdapat perintah yang mengirimkan respons status HTTP 200 (permintaan berhasil) ditandai dengan pesan "Hello from WiFiManager!" yang muncul di halaman *server browser*.

`WiFiManager` digunakan untuk memudahkan pengaturan koneksi WiFi. *Library* ini memungkinkan *mikrokontroler* ESP8266 melakukan konfigurasi WiFi secara otomatis tanpa pengaturan manual. Perintah `if` dalam `WiFiManager` menghubungkan kembali *mikrokontroler* ke jaringan WiFi "Kandang Puyuh". Jika

tidak berhasil terhubung maka program menampilkan pesan "Failed to connect" pada monitor guna menginformasikan kepada *user* bahwa koneksi WiFi gagal.

Fungsi `delay(3000)` menjadi indikator waktu tunggu selama 3 sekon setelah koneksi WiFi gagal. Setelah itu, *mikrokontroler* mereset otomatis dengan perintah `ESP.reset()` dan menghubungkan kembali ke WiFi. Setelah terhubung, *server* HTTP akan *request* ke *root* URL ("/"). Saat mengakses halaman utama *server*, fungsi `handleRoot` dijalankan dan mengirimkan pesan "Hello from WiFiManager!" ke *browser*. *Mikrokontroler* menerima *request* dari *client* (*browser*) saat memulai *server web* dan merespons sesuai dengan pengaturan kode. Kemudian, respons ditampilkan dalam LCD.

4.2.3 Tampilan dan Pembacaan Sensor

Tahap ketiga pembacaan data dengan sensor dan menampilkannya pada LCD. Parameter pertama yakni suhu dan kelembapan yang diukur dengan sensor DHT22.

```
void update_display() {  
  lcd.clear();  
  lcd.setCursor(0, 0);  
  lcd.print("SUHU : ");  
  lcd.print(suhu);  
  lcd.print(" C");  
  lcd.setCursor(0, 1);  
  lcd.print("KELEMBAPAN : ");  
  lcd.print(kelembapan);  
  lcd.print(" %");  
}
```

Gambar 4.9 Listing Code DHT22

Sebelum menampilkan data pengukuran suhu layar LCD dibersihkan terlebih dahulu dari seluruh teks atau informasi sebelumnya. Lalu, tampilan kursor LCD diatur ke baris pertama dan kolom pertama. Kursor ini sebagai tempat teks berikutnya. *Listing code* di atas menampilkan suhu dan kelembapan. Pada baris kelima perintah untuk menampilkan dan mendefinisikan teks Suhu. Kemudian nilai variabel hasil pengukuran suhu sensor DHT11 atau DHT22 dalam °C.

Selain suhu, LCD juga menampilkan variabel persentase kelembapan udara (%). Fungsi void `update_display` memperbarui tampilan LCD dengan informasi suhu dan kelembapan terbaru. Layar LCD dibersihkan dahulu lalu menampilkan suhu (°C) baris pertama dan kelembapan (%) di baris kedua. Penggunaan fungsi `update_display` untuk menampilkan data berupa suhu dan kelembapan secara *real time* sehingga data terbaru selalu ditampilkan LCD.

```
void baca_kebisingan() {
    unsigned long startMillis = millis(); // Catat waktu mulai
    unsigned int peakToPeak = 0;          // Amplitudo sinyal
    puncak-ke-puncak

    79 unsigned int signalMax = 0;
    unsigned int signalMin = 1023;

    // Ambil sampel selama sampleWindow milidetik
    while (millis() - startMillis < sampleWindow) {
        int sample = analogRead(micPin);
        if (sample < 1024) {
            if (sample > signalMax) {
                signalMax = sample; // Perbarui sinyal maksimal
            }
            if (sample < signalMin) {
                signalMin = sample; // Perbarui sinyal minimal
            }
        }
    }

    peakToPeak = signalMax - signalMin; // Hitung
    puncak-ke-puncak
    double voltage = (peakToPeak * 5.0) / 1024.0; // Konversi ke
    voltase

    // Perkiraan desibel
    kebisingan = 20 * log10(voltage * 1000); // Skala ke dB
    relatif
}
```

Gambar 4.10 Listing Code Pengukuran Kebisingan Kandang

Parameter pengukuran kedua yakni mengukur tingkat kebisingan kandang burung puyuh menggunakan sensor Max4466 yang terhubung ke *pin* analog A0. Nilai pembacaan kebisingan oleh sensor dikonversi dalam skala desibel (dB). Dimulai dengan mencatat waktu ketika fungsi mulai dieksekusi dengan variabel `startMillis`. Fungsi `millis` akan mengembalikan data dalam mili sekon (ms) sejak *mikrokontroler* berjalan sehingga dapat mengukur durasi pengambilan sampel.

Dikumpulkan data sinyal dengan berbagai variabel pengukuran. Variabel `peakTopeak` menyimpan amplitudo sinyal dari puncak-ke-puncak yang dihitung setelah mengambil sampel dari mikrofon. Setelah itu, mencatat nilai sinyal tertinggi dan terendah dari gelombang sinyal. Sensor analog memberikan nilai antara 0 s.d. 1023 (ADC 10 bit) sehingga nilai awal akan di set ke 1023 dan nilai yang lebih kecil akan diperbarui. `sampleWindow` mengambil sampel selama periode waktu yang telah ditentukan dalam satuan mili sekon (ms).

Fungsi `analogRead(micPin)` membaca nilai analog dari *pin* sensor suara terhubung. Nilai tersebut merepresentasikan digital sinyal suara yang diterima oleh mikrofon. Fungsi `if` berfungsi sebagai logika dan memeriksa sampel suara masukan. Batas maksimal (*default*) sensor ADC yakni 1024. Jika sampel masukan lebih dari 1024 maka sinyal maksimal akan diperbarui dengan sesuai dengan besaran nilai sampel. Lalu, dihitung amplitudo sinyal yang menggambarkan variasi sinyal selama periode sampling.

Nilai *peak to peak* dihitung dalam bentuk voltase. *Mikrokontroler* ESP98266 menggunakan ADC 10 bit dengan nilai maksimum ADC dengan tegangan referensi sebesar 5V sehingga konversi tegangan diperoleh dengan nilai V_{pp} dikalikan nilai tegangan referensi lalu dibagi dengan nilai ADC maksimum. Selanjutnya, perolehan nilai voltase tegangan diubah menjadi tingkat kebisingan satuan desibel (dB). Hasil pengukuran yang disimpan dalam variabel kebisingan dapat digunakan dalam berbagai aplikasi seperti deteksi kebisingan di sekitar perangkat sensor.

4.2.4 Pengolahan dan Pengiriman Data

```
void send_firebase() { //kirim ke firebase
  Firebase.setFloat(firebaseData, "/suhu", suhu);
  Firebase.setFloat(firebaseData, "/kelembapan", kelembapan);
  Firebase.setFloat(firebaseData, "/kebisingan", kebisingan);
}
```

Gambar 4.11 *Listing Code* Pengiriman Data Ke Firebase

Tahap keempat mengirimkan data hasil pengukuran suhu, kelembapan, dan kebisingan ke layanan eksternal yakni *firebase* dan *spread sheet*. Fungsi *firebaseData* memerlukan tiga parameter yakni objek, lokasi, dan nilai. Objek digunakan dari *firebase data* digunakan untuk berkomunikasi dengan *firebase*. *Path* menunjukkan lokasi penyimpanan data dalam *firebase*. *Value* merupakan nilai yang disimpan. Pada penelitian ini, Fungsi *send_firebase()* akan mengirimkan nilai yang disimpan yakni suhu, kelembapan, dan tingkat kebisingan ke *Firebase Realtime Database* dalam bentuk format *float*.



Gambar 4.12 Contoh Tampilan Data Di *Firebase*

Pada gambar 4.8 menampilkan contoh hasil perolehan data yang dikirimkan ke *firebase*. Pada pengukuran, terlihat kandang dalam kondisi ideal dengan suhu kandang sebesar 30°C, kelembapan 67%, dan tingkat kebisingan 50 dB. Setelah dikirimkan ke *firebase* data juga dikirimkan ke *spread sheet*.

Fungsi `send_spreadsheet()` bertugas mengirimkan data-data pengukuran ke *google spread sheet* melalui Google Apps Scripts. Data dikirim menggunakan HTTP Get request yang berisikan informasi mengenai parameter suhu, kelembapan, dan tingkat kebisingan kandang burung puyuh.

```
void send_spreadsheet() { //kirim ke spread
  if ((millis() - lastTime) > timerDelay) {
    if (!client.connect(host, httpsPort)) {
      Serial.println("connection failed");
      return;
    }

    String url = "/macros/s/" + GAS_ID + "/exec?suhu=" +
String(suhu) + "&kelembapan=" + String(kelembapan) + "&db=" +
String(kebisingan);
    client.print(String("GET ") + url + " HTTP/1.1\r\n" +
"Host: " + host + "\r\n" + "User-Agent:
BuildFailureDetectorESP8266\r\n" + "Connection:
close\r\n\r\n");
    lastTime = millis();
  }
}
```

Gambar 4.13 Listing Code Pengiriman Data Ke Google Spread Sheet

Pengiriman data ke *google spread sheet* diawali dengan fungsi `if ((millis() - lastTime) > timerDelay)` dimana data dikirim ke *google spreadsheet* jika interval waktu yang ditentukan telah tercapai. Fungsi `if (!client.connect(host, httpsPort))` mengkoneksikan HTTPS ke server tujuan. Jika koneksi gagal, maka pesan "connection failed" muncul ke serial monitor dan fungsi berhenti.

Jika koneksi berhasil, dibuat URL untuk *request* HTTP Get. URL ini digunakan untuk mengakses Google App Script yang telah disiapkan sebelumnya. `GAS_ID` merupakan ID script yang telah dibuat di google app script. Fungsi `client.print` digunakan oleh *mikrokontroler* ESP8266 untuk mengirimkan data pengukuran ke Apps Script. Google Apps Script berfungsi sebagai penerima data dan dipublikasikan sebagai Web App dengan ID yang unik. *Web App* menerima parameter pengukuran melalui *query string* dan disimpan ke dalam *google spread sheet*.

4.2.5 Mengontrol Logika Berdasarkan Data

Tahap kelima yaitu mengontrol logika LED indikator dan *relay* dengan data. Pada tahap ini, LED indikator dan *relay* lampu dikontrol berdasarkan kondisi terukur oleh sensor. Jika suhu, kelembapan dan kebisingan berada dalam rentang tertentu, lampu dinyalakan dan LED indikator akan berubah warna merah.

```
if (suhu > 30.0 && kelembapan >= 30.0 && kelembapan <= 80.0 &&
kebisingan <= 80) {
    digitalWrite(_merah, HIGH); //Aktif low
    digitalWrite(_hijau, LOW);
    digitalWrite(_biru, HIGH);
    digitalWrite(relay_lampu, HIGH);
} else {
    digitalWrite(_merah, LOW); //Aktif low
    digitalWrite(_hijau, HIGH);
    digitalWrite(_biru, HIGH);
    digitalWrite(relay_lampu, LOW);
}
```

Gambar 4.14 Listing Code Logika Kontrol LED dan Relay

Listing code diatas tergolong bagian struktur kondisional yang mengatur status LED indikator dan relay berdasarkan variabel suhu, kelembapan, dan kebisingan. Jika terdapat kondisi tertentu terpenuhi, maka LED dan relay diatur ke status yang diinginkan. Demikian sebaliknya, jika tidak maka diatur ke status lainnya.

Fungsi `if (suhu > 30.0 && kelembapan >= 30.0 && kelembapan <= 80.0 && kebisingan <= 80)` berfungsi sebagai parameter ideal kondisi kandang burung puyuh yang diinginkan yakni suhu kandang tidak lebih dari 30°C, kelembapan 30% s.d. 80% dan kebisingan kurang dari 80dB. Jika kandang dalam kondisi ideal di mana suhu tidak terlalu tinggi, kelembapan dalam rentang aman, dan tingkat kebisingan tidak tinggi maka LED merah dan biru akan menyala, LED hijau dimatikan, *relay* lampu diaktifkan sehingga lampu menyala.

Sebaliknya, jika kondisi kandang tidak memenuhi kategori optimal maka LED hijau akan menyala, sementara LED merah dimatikan, relay dan lampu akan dimatikan. Penggunaan LED terdiri dari warna merah, biru, dan hijau di mana

warna LED yang berbeda digunakan sebagai status indikator kandang. LED merah menunjukkan kondisi kandang tidak ideal seperti suhu dan tingkat kebisingan kandang terlalu tinggi. LED hijau menunjukkan kondisi normal bahkan lebih baik. LED biru berfungsi sebagai indikasi tambahan seperti sistem aktif atau kandang dalam mode tertentu.

4.2.6 Fungsi *Loop* Utama

Tahap terakhir yaitu merancang struktur dari fungsi *loop* utama di mana bagian terpenting yang terus berjalan selama perangkat menyala. Dalam fungsi ini tampilan LCD terus diperbarui dengan data terbaru, dilakukan pembacaan data oleh sensor, pengiriman data ke *firebase* dan *google sheets*, dan mengontrol perangkat keras (LED dan *relay*) yang dilakukan berdasarkan data sensor.

```
void loop() {  
  update_display(); //Update tampilan lcd realtime  
  kelembapan = dht.readHumidity();  
  suhu = dht.readTemperature();  
  baca_kebisingan();  
  if (isnan(kelembapan) || isnan(suhu)) {  
    Serial.println(F("Gagal Baca Data DHT sensor!"));  
    return;  
    digitalWrite(_merah, HIGH); //Aktif low  
    digitalWrite(_hijau, HIGH);  
    digitalWrite(_biru, HIGH);  
  }  
  if (suhu > 30.0 && kelembapan >= 30.0 && kelembapan <= 80.0  
&& kebisingan <= 80) {  
    digitalWrite(_merah, HIGH); //Aktif low  
    digitalWrite(_hijau, LOW);  
    digitalWrite(_biru, HIGH);  
    digitalWrite(relay_lampu, HIGH);  
  } else {  
    digitalWrite(_merah, LOW); //Aktif low  
    digitalWrite(_hijau, HIGH);  
    digitalWrite(_biru, HIGH);  
    digitalWrite(relay_lampu, LOW);  
  }  
  server.handleClient(); send_firebase(); send_spreadsheet();  
}
```

Gambar 4.15 Listing Code Fungsi *Loop* Utama

Di dalam `loop()` data suhu, kelembapan, dan kebisingan dibaca secara periodik. Jika data sensor valid, maka status lampu dan LED disesuaikan berdasarkan kondisi suhu, kelembapan, dan kebisingan. Jika suhu lebih dari 30°C, kelembapan sebesar 30% s.d. 50% dan tingkat kebisingan kurang dari 80dB, maka lampu akan menyala. Demikian sebaliknya, ketika kondisi kandang optimal lampu akan dimatikan. Selanjutnya, data dikirimkan ke *firebase* dan *google spreadsheet*.

Fungsi `update_display()` : memperbarui tampilan LCD guna menampilkan informasi suhu dan kelembapan terbaru. Pembaruan informasi memberikan *feedback* visual kepada *user*. `kelembapan = dht.readHumidity();` dan `suhu=dht.readTemperature();` membaca data kelembapan dan sensor suhu pada sensor DHT22. Kelembapan dibaca kembali dengan satuan persentase (%) dan suhu dibaca dalam satuan °C. Setelah parameter suhu dan kelembapan, dilakukan pengukuran tingkat kebisingan menggunakan sensor mikrofon yang terhubung ke *pin* analog. Sensor mengukur amplitudo suara di lingkungan sekitar kandang.

Seluruh *listing code* ini mengimplementasikan sistem *monitoring* kondisi kandang burung puyuh menggunakan sensor suhu, kelembapan, dan kebisingan. Data yang ditampilkan LCD dan dikirimkan ke *firebase* dan *google spread sheet* bertujuan untuk melakukan *monitoring* jarak jauh. Logika dalam sistem akan mengaktifkan lampu dan *relay* berdasarkan kondisi *realtime* suhu, kelembapan, dan kebisingan yang terdeteksi ketika dilakukan pengukuran.

4.3 Analisis Hasil Kalibrasi Alat

Dalam sub bab 4.2 telah dijelaskan bahwa data *monitoring* diperoleh sensor lalu dikirim ke *firebase* oleh mikrokontroler ESP8266. Untuk menjamin keakuratan data dilakukan pengujian sistem kalibrasi sensor DHT22 dan Max4466. Kalibrasi didefinisikan sebagai proses verifikasi keakuratan alat ukur sesuai dengan rancangannya. Kalibrasi bertujuan untuk mengetahui perbedaan nilai dari pembacaan alat dengan membandingkan nilai standar sehingga menjamin perolehan data yang valid [43]. Pada penelitian ini kalibrasi dilakukan terhadap sensor DHT22 dan sensor Max4466.

4.3.1 Analisis Kalibrasi Sensor DHT22

Kalibrasi pertama kali dilakukan terhadap sensor DHT22. Berikut tampilan tabel 4.1 menunjukkan perolehan data pengukuran suhu dan kelembapan.

Tabel 4.1 Kalibrasi Sensor DHT22

No	Pengukuran	Suhu		Kelembapan	
		DHT 22 (Y)	Termometer (X)	DHT 22 (Y)	Higrometer (X)
1.	Pertama	28,70	29,45	72	75
2.	Kedua	29,30	30,50	74	70
Nilai Error		3,24%		4,86%	

Data di atas menunjukkan perolehan data sensor DHT22 dan alat ukur digital serta kalibrasi setelah dilakukan dua kali pengukuran. Perhitungan kalibrasi sensor DHT22 menggunakan persamaan 2.1 lalu hasil perhitungan dinyatakan dalam bentuk persen. Tahapan perhitungan nilai *error* sensor DHT22 selengkapnya dapat dilihat pada lampiran D.

Berdasarkan data tabel 4.1 terlihat nilai *error* suhu sebesar 3.24% dan kelembapan 4,86%. Besaran *error* berperan penting terhadap proses pengukuran dan dipengaruhi oleh berbagai faktor seperti posisi pengukuran, suhu dan keadaan lingkungan, hingga tingkat keakuratan alat ukur pembanding yang digunakan ketika kalibrasi.

4.3.2 Analisis Kalibrasi Sensor Max4466

Kalibrasi kedua dilakukan terhadap sensor Max4466 yang mendeteksi tingkat kebisingan di sekitar kandang. Perolehan data tingkat kebisingan kandang burung puyuh ditampilkan pada tabel 4.2.

Tabel 4.2 Kalibrasi Sensor Max4466

No	Percobaan	Waktu Pengukuran	Data Sensor	Data Aplikasi	Nilai Error	Presentase Error (%)
1.	Pertama	16.36.15	44,00	44,20	0,2	0,45
2.	Kedua	16.36.29	46,00	47,00	1,0	2,12
3.	Ketiga	16.36.51	42,00	41,80	0,2	0,47
Rata-Rata Nilai Error					0,47	1,0

Kalibrasi dilakukan dengan membandingkan data hasil pengukuran sensor Max4466 menggunakan aplikasi pengukur kebisingan (*Sound Meter*). Metode ini digunakan karena harga perangkat asli untuk kalibrasi sensor Max4466 cukup mahal sehingga penulis memilih metode kalibrasi yang lebih terjangkau.

Tabel 4.2 menampilkan hasil kalibrasi yang dilakukan sebanyak tiga kali dengan perolehan data yang berbeda perhitungan data kalibrasi sensor Max4466 menggunakan metode *absolute error*, *percentage relative error*, *mean absolute error* yang dinyatakan dalam persamaan 2.1, 2.2, dan 2.3. Dari perolehan nilai *absolute error* dan *relative error* yang tidak lebih dari 5% dapat disimpulkan bahwa proses pengukuran dan kalibrasi sudah berjalan dengan baik dan data pengukuran bersifat akurat.

4.4 Analisis Hasil Data

Setelah menentukan lokasi ideal serta melakukan kalibrasi sensor DHT22 dan Max4466 selanjutnya dilakukan pengukuran langsung terhadap kondisi kandang burung puyuh dengan waktu berbeda yakni pagi hari, siang hari, sore hari, dan malam hari.

Parameter penelitian berfokus kepada suhu, kelembapan, dan kebisingan kandang. Setiap parameter diamati secara berkala dengan rentang waktu bertujuan untuk memperoleh data yang akurat menggambarkan kondisi kandang. Berikut ini data-data yang diperoleh selama proses penelitian.

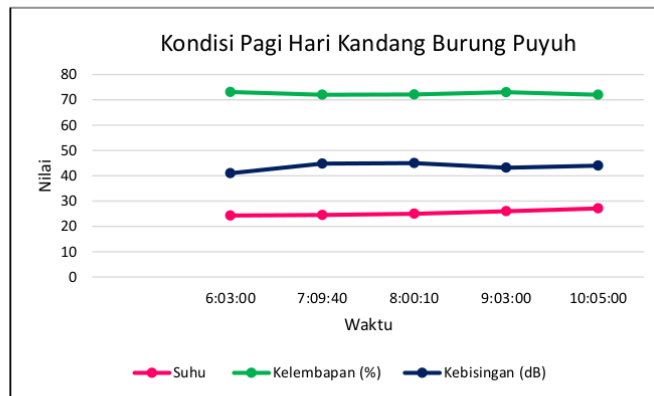
4.4.1 Analisis Hasil Pengujian Pagi Hari

Pengujian pertama dilakukan pagi hari mulai pukul 06.03 s.d. 10.05. Diawali dengan menghidupkan alat dan koneksikan ke jaringan internet. Setelah itu, sensor DHT22 dan sensor Max4466 mengukur suhu, kelembapan, dan tingkat kebisingan kandang. *Backend* mengolah data menggunakan JavaScript lalu menghubungkan *firebase* dengan *fronted*. Terakhir data monitoring ditampilkan pada halaman *website (frontend)* secara *realtime* Berikut tabel 4.3 menunjukkan perolehan data pada pagi hari:

Tabel 4.3 Hasil Pengujian Kandang Pagi Hari

No	Tanggal	Waktu	Suhu (°C)	Kelembapan (%)	Kebisingan (dB)	Relay	LED
1	12-7-25	6:03:00	24,30	73,10	41,00	Mati	Hijau
2	12-7-25	7:09:40	24,50	72,00	44,80	Mati	Hijau
3	12-7-25	8:00:10	25,00	72,10	45,00	Mati	Hijau
4	12-7-25	9:03:00	26,00	73,00	43,20	Mati	Hijau
5	12-7-25	10:05:00	27,10	72,00	44,00	Mati	Hijau

Tabel 4.3 di atas menunjukkan hasil pengukuran kandang pagi hari. Tampak seluruh parameter pengukuran dalam kondisi ideal. Kondisi kandang pada pagi hari ditampilkan pada gambar 4.17 di bawah ini.



Gambar 4.16 Grafik Kondisi Kandang Pagi Hari

Berdasarkan tabel 4.3 dan gambar 4.16 terlihat kondisi kandang pada pagi hari cukup baik. Seluruh parameter pengukuran baik suhu, kelembapan, dan tingkat kebisingan dalam kondisi ideal. Suhu kandang relatif stabil dikarenakan suhu udara pagi hari tidak terlalu panas. Tingkat kelembapan kandang burung pada pagi hari cukup beragam mulai dari 73,10%, 72%, 72,10%, 73% dan 72%. Perbedaan tingkat kelembapan udara disebabkan oleh berbagai faktor eksternal seperti kecepatan angin yang bertiup setiap jam. Sampel visualisasi pengukuran pagi hari pada *website* dan *firebase* dapat dilihat pada lampiran B.

Parameter pengukuran ketiga adalah tingkat kebisingan kandang. Kebisingan kandang menjadi indikator terhadap tingkat kebisingan yang ada baik di sekitar atau di dalam kandang. Tingkat kebisingan sangat mempengaruhi produktivitas burung puyuh. Kandang dikatakan ideal jika tingkat kebisingan tidak lebih dari 70dB. Hal ini disebabkan burung puyuh tidak mampu hidup dengan kondisi lingkungan bising. Ketika kebisingan lingkungan lebih dari 70dB membuat burung puyuh menjadi *stres* dan tidak bertelur sehingga tingkat produktivitas menurun.

Penggunaan *relay* penelitian ini berfungsi untuk menghidupkan lampu pijar pada kandang. Kandang dikatakan ideal jika suhu berada pada 24-30°C. Ketika suhu kandang kurang dari pada 24°C maka *relay* secara otomatis menyalakan lampu pijar.

Ketika menyala, lampu pijar akan menghasilkan panas untuk menghangatkan kondisi kandang dan LED ON.

Demikian sebaliknya jika suhu kandang melebihi kondisi ideal, maka *relay* akan mematikan lampu pijar agar suhu di kandang tidak semakin tinggi. Suhu tinggi dapat membuat burung merasa gelisah dan dapat menurunkan produktivitas burung puyuh. Saat itu, *relay* dalam kondisi OFF dan LED ON. Jika kandang dalam rentang kondisi ideal maka *relay* secara otomatis mematikan lampu pijar. Kondisi *relay* dan LED OFF.

Penggunaan LED bertujuan hanya sebagai sarana notifikasi atau indikator suhu dan lampu pijar pada kandang. Saat suhu kandang berada di bawah batas *relay* ON dan saat suhu di atas kondisi ideal *relay* OFF maka LED ON (berwarna merah) dan LED OFF (berwarna hijau) saat kandang dalam kondisi ideal.

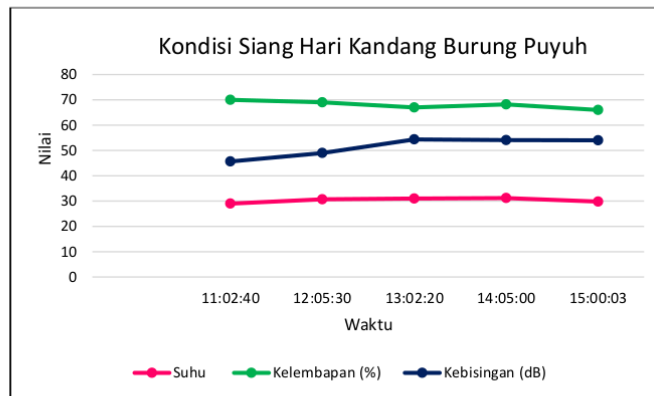
4.4.2 Analisis Hasil Pengujian Siang Hari

Setelah berhasil memperoleh data di pagi hari, penelitian tetap dilanjutkan di siang hari mulai pukul 11:02 s.d. 15:00. Berikut data pengujian siang hari:

Tabel 4.4 Hasil Pengujian Kandang Siang Hari

No	Tanggal	Waktu	Suhu (°C)	Kelembapan (%)	Kebisingan (dB)	Relay	LED
1	12-7-25	11:02:40	29,00	70,00	45,65	Mati	Hijau
2	12-7-25	12:05:30	30,70	69,00	49,00	Mati	Merah
3	12-7-25	13:02:20	31,00	67,00	54,36	Mati	Merah
4	12-7-25	14:05:00	31,20	68,20	54,10	Mati	Merah
5	12-7-25	15:00:03	29,80	66,00	54,00	Mati	Hijau

²³ Tabel 4.4 di atas menunjukkan hasil pengukuran kandang di siang hari. Data pengujian pada siang hari berbeda dengan data di pagi hari. Visualisasi data tabel 4.4 dapat dilihat pada gambar 4.18 dibawah ini.



Gambar 4.17 Grafik Kondisi Kandang Siang Hari

Berdasarkan tabel 4.4 dan gambar 4.17 terlihat suhu dan tingkat kebisingan kandang cenderung mengalami kenaikan. Kenaikan suhu kandang pada siang hari dipengaruhi oleh berbagai faktor seperti perbedaan intensitas cahaya matahari, adanya akumulasi panas di permukaan bumi, hingga radiasi matahari terhadap bumi paling tinggi dikarenakan matahari berada tepat di tengah bumi [44].

Pada siang hari, suhu kandang mencapai 31,20°C sehingga lampu LED berwarna merah. LED berwarna merah menandakan parameter pengukuran berada diluar kondisi ideal. Kenaikan suhu kandang menimbulkan reaksi yakni burung puyuh menjadi gelisah. Oleh karena itu, hal ini penting diperhatikan peternak supaya kondisi kandang tetap stabil.

Bertolak belakang dengan suhu, kelembapan udara cenderung mengalami penurunan. Kenaikan suhu akan mempercepat penguapan dari permukaan bumi, dan tingkat kelembapan relatif menurun sehingga udara yang menguap akan terasa lebih kering. Hal ini membuat kelembapan udara siang hari akan lebih rendah dibandingkan pagi hari. Sedangkan tingkat kebisingan siang hari lebih tinggi dibandingkan dengan pagi hari disebabkan oleh tingginya aktivitas masyarakat yang dapat menimbulkan polusi suara dari luar kandang. Sampel visualisasi pengukuran siang hari pada *website* dan *firebase* dapat dilihat pada lampiran B.

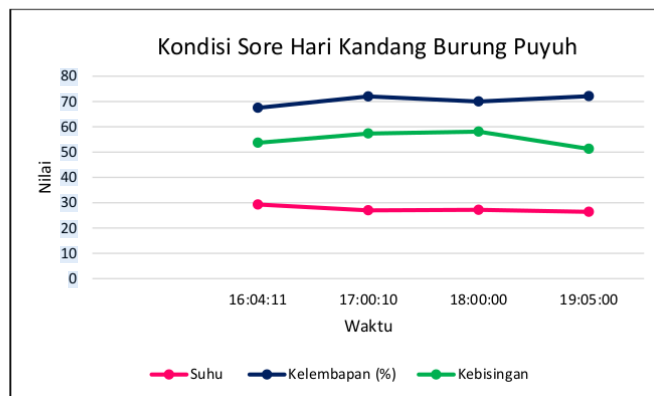
4.4.3 Analisis Hasil Pengujian Sore Hari

Penelitian sore hari dilakukan mulai pukul 16:04 s.d. 19:05. Berikut tabel 4.5 menampilkan data pengujian sore hari:

Tabel 4.5 Hasil Pengujian Sore Hari

No	Tanggal	Waktu	Suhu (°C)	Kelembapan (%)	Kebisingan	Relay	LED
1	12-7-25	16:04:11	29,30	67,50	53,70	Mati	Hijau
2	12-7-25	17:00:10	27,00	72,00	49,00	Mati	Hijau
3	12-7-25	18:00:00	27,20	70,00	49,53	Mati	Hijau
4	12-7-25	19:05:00	26,40	72,10	49,00	Mati	Hijau

Tabel 4.5 menunjukkan kondisi kandang sore hari berbeda dengan siang hari. Pada sore hari suhu kandang menurun sehingga seluruh parameter kembali dalam batas ideal. Hal ini membuat LED kembali berwarna hijau. Kondisi kandang sore hari digambarkan pada gambar 4.18 berikut.



Gambar 4.18 Grafik Kondisi Kandang Sore Hari

Berdasarkan tabel 4.5 dan gambar 4.18 terlihat suhu dan tingkat kebisingan kandang burung puyuh menurun sedangkan kelembapan udara kandang burung meningkat. Penurunan suhu kandang dikarenakan intensitas penyinaran matahari sore tidak seperti siang hari. Hal ini berbanding lurus dengan adanya pergeseran matahari ke arah barat sehingga tidak lagi berada pada posisi zenith. Namun,

penurunan suhu berdampak terbalik dengan kelembapan udara. Saat suhu kandang menurun, maka kelembapan relatif akan semakin meningkat sehingga kelembapan udara dikandang mengalami kenaikan.

Kebisingan kandang mengalami kenaikan disebabkan oleh adanya perubahan perilaku burung. Penurunan intensitas cahaya matahari saat sore hari membuat burung puyuh merasa gelisah. Respons yang ditampilkan oleh burung puyuh gelisah ditandai dengan mengeluarkan desiran dan lebih aktif bergerak. Selain gelisah, burung puyuh juga akan bersuara ketika mencari tempat yang lebih nyaman untuk tidur. Secara umum, peningkatan kebisingan kandang menunjukkan respons alami burung puyuh terhadap kombinasi perubahan lingkungan seperti penurunan intensitas cahaya matahari dan perubahan ritme harian. Sampel visualisasi data pengukuran sore hari pada *website* dan *firebase* dapat dilihat pada lampiran B.

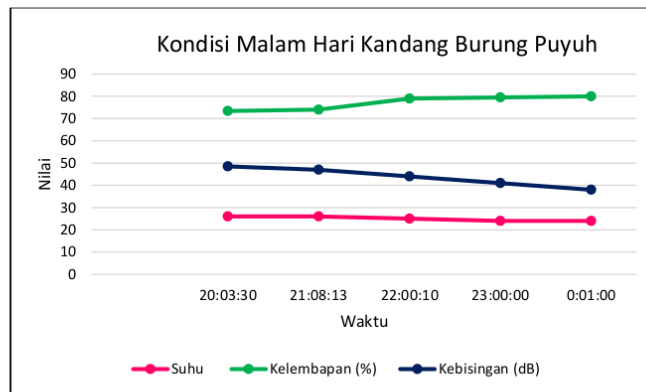
4.4.4 Analisis Hasil Pengujian Malam Hari

Penelitian pada malam hari dilakukan mulai pukul 20:03 s.d. 00:01. Berikut tabel 4.6 menampilkan data pengujian malam hari:

Tabel 4.6 Hasil Pengujian Malam Hari

No	Tanggal	Waktu	Suhu (°C)	Kelembapan (%)	Kebisingan	Relay	LED
1	12-7-25	20:03:30	26,00	73,40	48,50	Mati	Hijau
2	12-7-25	21:08:13	26,00	74,00	47,00	Mati	Hijau
3	12-7-25	22:00:10	25,00	79,00	44,00	Mati	Hijau
4	12-7-25	23:00:00	24,00	79,50	41,00	Mati	Hijau
5	13-7-25	00:01:00	24,00	80,00	38,00	Mati	Hijau

Tabel 4.6 menampilkan hasil pengujian pada malam hari. Suhu kandang cenderung stabil, tingkat kebisingan menurun, dan tingkat kelembapan kandang meningkat. Walaupun demikian, kandang masih dalam kondisi batas normal sehingga LED tetap berwarna hijau. Berikut tampilan gambar 4.20 menunjukkan kondisi malam hari.



Gambar 4.19 Grafik Kondisi Kandang Malam Hari

Gambar 4.19 menunjukkan suhu di malam hari cenderung stabil dan lebih rendah dibandingkan suhu kandang di sore hari. Penurunan suhu ini disebabkan oleh berbagai faktor seperti kurangnya tutupan awan pada malam hari menyebabkan radiasi panas dari permukaan bumi terpancar ke atmosfer tanpa hambatan sehingga terjadi penurunan suhu yang cukup signifikan. Selain itu, angin tenang malam hari akan menghambat udara sehingga udara dingin terperangkap di permukaan bumi [45].

Kebisingan kandang malam hari mengalami penurunan dibandingkan sore hari. Tidak ada aktivitas yang dilakukan sehingga kondisi lingkungan sekitar kandang menjadi hening, keterbatasan penglihatan burung puyuh, dan malam hari menjadi waktu untuk tidur menjadi faktor pendukung penurunan kebisingan pada kandang burung puyuh. Sampel visualisasi pengukuran malam hari pada *website* dan *firebase* dapat dilihat pada lampiran B.

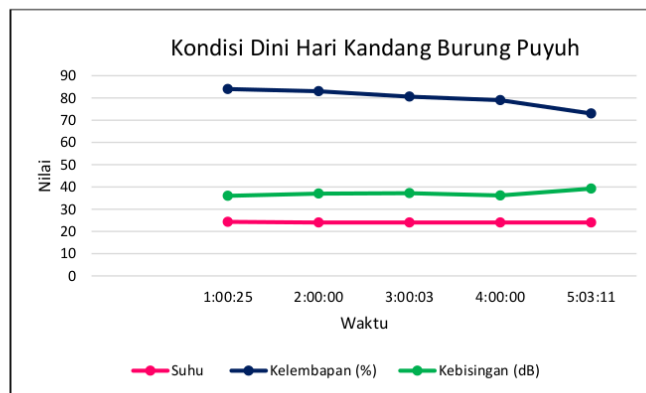
4.4.5 Analisis Hasil Pengujian Dini Hari

Penelitian terakhir dilakukan dini hari mulai pukul 01:00 s.d. 05:03. Berikut tabel 4.7 menunjukkan data pengujian dini hari:

Tabel 4.7 Hasil Pengujian Dini Hari

No	Tanggal	Waktu	Suhu (°C)	Kelembapan (%)	Kebisingan	Relay	LED
1	13-7-25	01:00:25	24,30	84,00	36,00	Mati	Hijau
2	13-7-25	02:00:00	24,00	83,00	37,00	Mati	Hijau
3	13-7-25	03:00:03	24,00	80,60	37,20	Mati	Hijau
4	13-7-25	04:00:00	24,00	79,00	36,15	Mati	Hijau
5	13-7-25	05:03:11	24,00	73,00	39,23	Mati	Hijau

Tabel 4.7 menunjukkan hasil pengukuran suhu pada dini hari memiliki nilai terendah dibandingkan pengukuran pagi hari, siang hari, sore hari, dan malam hari. Data tabel 4.7 ditampilkan pada gambar 4.21 berikut ini.



Gambar 4. 20 Grafik Kondisi Kandang Burung Dini Hari

Pengukuran pada pukul 01:00 menunjukkan suhu sebesar 24.00°C dengan kelembapan udara 82.50%. Suhu rendah menjadikan tingkat kelembapan udara dini hari meningkat. Hal ini telah dibahas pada ulasan data sebelumnya bahwa suhu yang rendah akan menghambat udara sehingga kelembapan udara meningkat.

Data kebisingan pada dini hari cukup beragam mulai 37,39 dB, 42,33 dB, 40,45 dB, 42,18 dB, 44,83 dB. Bila dibandingkan dengan kebisingan malam hari, kondisi kandang dini hari cenderung lebih rendah and berubah-ubah. Perubahan kondisi ini didukung oleh berbagai faktor seperti penurunan suhu dan kenaikan tingkat kelembapan kandang sehingga burung puyuh merasa kurang nyaman, dan adanya perubahan aktivitas burung yang mulai beradaptasi terhadap perubahan waktu dini hari menuju pagi hari.

127

Berdasarkan perolehan data di atas dapat disimpulkan bahwa kondisi suhu, kelembapan, dan tingkat kebisingan kandang dalam setiap waktu pengukuran berbeda-beda. Perolehan data tersebut ditampilkan secara *realtime* pada *website* dan *firebase*. Visualiasi data pada *website* dan *firebase* dapat dilihat pada lampiran B.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan perolehan data selama kegiatan penelitian mengenai sistem *monitoring* dan kontrol kandang burung puyuh dapat disimpulkan bahwa:

1. Implementasi sistem *monitoring* otomatis menunjukkan pada siang hari parameter pengukuran mengalami kenaikan yang cukup signifikan, suhu mencapai 32,50°C, kelembapan 70,80% dan tingkat kebisingan sebesar 65,89 dB.
2. Visualisasi hasil diberikan dalam bentuk grafik dalam setiap waktu *monitoring*. Tampak perubahan grafik yang berbeda-beda dalam pagi hari, siang hari, sore hari, malam hari, dan dini hari.
3. Sensor DHT22 dan Max4466 terbukti akurat dalam *monitoring* kandang. Hal ini terlihat dari hasil kalibrasi yang menampilkan nilai *error* sensor DHT22 sebesar 3,24% dan 4,86% serta error Max4466 sebesar 1%. Nilai *error* pada setiap sensor berada dibawah ambang batas sehingga dapat dinyatakan data yang diperoleh akurat.

5.2 Saran

Beberapa masukan mengenai penelitian ini antara lain:

1. Penambahan sensor pengamatan sehingga perolehan data semakin akurat.
2. Peningkatan variabel pengukuran sehingga kualitas dan produktivitas burung puyuh semakin meningkat.

DAFTAR PUSTAKA

- [1] ²² Cut Bidara Panita Umar, "Penyuluhan Tentang Pentingnya Peranan Protein Dan Asam Amino Bagi Tubuh Di Desa Negeri Lima," *J. Pengabd. Ilmu Kesehat.*, vol. 1, no. 3, pp. 52–56, 2023, doi: 10.55606/jpikes.v1i3.1412.
- [2] ⁹⁵ H. H. Outcomes and E. Impact, "Human Health Outcomes and Environmental Impact," *Nutrients*, vol. 14, p. 5115, 2022.
- [3] ³¹ I. Fahrurrozi, Wahyono, Y. Sari, A. K. Sari, I. Usuman, and B. Ariyadi, "Integrating random forest model and internet of things-based sensor for smart poultry farm monitoring system," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 33, no. 2, pp. 1283–1292, 2024, doi: 10.11591/ijeecs.v33.i2.pp1283-1292.
- [4] ⁵³ F. A. Soelistianto, A. Indrianto, M. A. Anshori, and M. N. R. Adriansyah, "IoT Based Poultry Cage Quality Monitoring System," vol. 02, no. 09, pp. 1780–1786, 2024.
- [5] M. International *et al.*, "Borneo Engineering & Advanced IoT Multiple Temperature Control Quail Egg Hatcher," vol. 3, no. September, pp. 82–87, 2024.
- [6] ⁹¹ T. T. Septian, K. Sari, and F. Setiawan, "Pemantauan Dan Pengendalian Lingkungan Ayam Peliharaan Dengan Implementasi Kandang Ayam Pintar Berbasis Iot," vol. 4, pp. 49–59, 2025.
- [7] ²¹ R. Abou-Elkhair *et al.*, "Effect of a diet supplemented with the moringa oleifera seed powder on the performance, egg quality, and gene expression in japanese laying quail under heat-stress," *Animals*, vol. 10, no. 5, pp. 1–12, 2020, doi: 10.3390/ani10050809.
- [8] ⁴² M. A. Hossain, A. S. M. Mahbub, and S. A. Belal, "Housing and dietary effects on production performance, quality index, and chemical composition of Japanese quail eggs," *Vet. Anim. Sci.*, vol. 23, no. February, p. 100340, 2024, doi: 10.1016/j.vas.2024.100340.
- [9] M. K. Rosikin, L. Perdanasari, T. A. I. Azis, and N. Amalia, "Implementasi

- Sistem Otomatisasi Monitoring Suhu, Kelembapan, dan Amonia pada Kandang Ayam Petelur Menggunakan Metode Fuzzy,” *J. Teknol. Inf. dan Terap.*, vol. 10, no. 2, pp. 75–82, 2023, doi: 10.25047/jtit.v10i2.325.
- [10] N. A. A. S. Aa Zezen Zenal Abidin, “SISTEM MONITORING KANDANG BURUNG PUYUH BERBASIS INTERNET OF THINGS PADA PLATFORM NODE-RED MENGGUNAKAN METODE NAIVE BAYES,” *J. Teknol. Inf. dan Komun.*, vol. 13, no. February, pp. 1–25, 2020.
- [11] F. Aryunita, N. Rasjid, and M. F. Mansyur, “Rancang Bangun Sistem Monitoring Keamanan Kandang Ayam Bloiler Menggunakan Esp32-Cam Berbasis Iot Dengan Aplikasi Android,” *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 1, 2024, doi: 10.23960/jitet.v12i1.3699.
- [12] I. G. N. A. W. Putra, A. Sukmaaji, and V. R. Hananto, “Pemanfaatan sensor proximity untuk memantau kesehatan reproduksi puyuh bagi peternak telur puyuh,” *Techno.Com*, vol. 22, no. 1, pp. 28–35, 2023, doi: 10.33633/tc.v22i1.7111.
- [13] K. A. Septiawan, A. N. Lisdawati, R. Mu, and M. Aminuddin, “Sistem Monitoring dan Kontrol Alat Penetas Telur Ayam Berbasis Internet of Things (IoT),” vol. 9, no. 2, pp. 556–563, 2025, doi: 10.36277/jteuniba.v9i2.1260.
- [14] A. Nahak, Y. P. . Kelen, B. Baso, and W. Sucipto, “Sistem Otomatisasi Pakan Minum dan Monitoring Kelembaban Udara berbasis Internet of Things (IoT),” *bit-Tech*, vol. 7, no. 1, pp. 1–9, 2024, doi: 10.32877/bt.v7i1.1226.
- [15] A. M. Kamarullah, Misswar Abd, Zulfan, Nazaruddin, “Kajian Distribusi Temperatur pada Inkubator Penetas Telur Puyuh,” *Univ. Syiah Kuala, Indones. Jl. Teuku Nyak Arief*, vol. 9, no. 2, pp. 132–139, 2023.
- [16] S. Time, “YAKIN TELUR,” vol. 9, no. 1, pp. 145–150, 2025.
- [17] M. D. Ananda, Y. Saragih, and R. S. Hadikusuma, “Utilization of Telemetry Systems in Quail Farming Using the Iot Concept,” *J. Teknol.*, vol. 15, no. 2, pp. 217–224, 2023.
- [18] Peternak, “Kandang Burung Puyuh,” 2020.

- <https://peternakrakyat.com/kandang-burung-puyuh/> (accessed Oct. 20, 2024).
- [19] ¹⁸ C. M. Ncho, J. I. Berdos, V. Gupta, A. Rahman, K. T. Mekonnen, and A. Bakhsh, "Abiotic stressors in poultry production: A comprehensive review," *J. Anim. Physiol. Anim. Nutr. (Berl.)*, vol. 109, no. 1, pp. 30–50, 2025, doi: 10.1111/jpn.14032.
- [20] R. Adolph, "Puyuh," pp. 1–23, 2021.
- [21] ⁵⁹ M. Datulangngan, L. J. Lambey, and K. M. Utara, ⁵⁹ "Pengaruh kepadatan ternak burung puyuh (*Coturnix coturnix japonica*) dalam kandang terhadap produktivitas telur," vol. 44, no. 1, pp. 116–124, 2024.
- [22] ²⁴ E. A. Susilo and D. S. Panjalu, "Efektivitas Manajemen Kandang untuk Mendukung Optimalisasi Produksi Telur pada Usaha Ternak Puyuh Petelur Panjalu," *Translitera J. Kaji. Komun. dan Stud. Media*, vol. 11, no. 2, pp. 33–38, 2022, doi: 10.35457/translitera.v11i2.2419.
- [23] ²⁹ A. Junaedi, M. D. M. Puspitasari, and M. Maulidina, "Pengaruh (Intensor) Induktor Heater Menggunakan Thermal Sensor Berbasis Mikrokontroler Arduino Nano Dalam Mengolah Logam," *Nusant. Eng.*, vol. 4, no. 2, pp. 169–175, 2021, doi: 10.29407/noe.v4i2.16754.
- [24] ³¹ Pktparticle, "Computer Organization - Von Neumann architecture," 2024. https://www.geeksforgeeks.org/computer-organization-von-neumann-architecture/?_x_tr_pto+=tc (accessed Oct. 21, 2024).
- [25] ⁵⁴ Pktparticle, "Harvard Architecture," 2024. https://www-geeksforgeeks-org.translate.goog/harvard-architecture/?_x_tr_sl=en&_x_tr_tl=id&_x_tr_hl=id&_x_tr_pto=tc (accessed Oct. 21, 2024).
- [26] ⁸⁴ NN, "Basic Principle Of Sensors: Definition & Principle of Sensors," 2024. <https://testbook.com/electrical-engineering/basic-principle-of-sensors> (accessed Nov. 01, 2024).
- [27] F. Nawazi, "DHT22 Digital Temperature and Humidity Sensor," 2021.

- <https://www.circuits-diy.com/dht22-digital-temperature-and-humidity-sensor/> (accessed Nov. 12, 2024).
- [28] ¹⁴ F. Puspasari, T. P. Satya, U. Y. Oktiawati, I. Fahrurrozi, and H. Prisyanti, "Analisis Akurasi Sistem sensor DHT22 berbasis Arduino terhadap Thermohygrometer Standar," *J. Fis. dan Apl.*, vol. 16, no. 1, p. 40, 2020, doi: 10.12962/j24604682.v16i1.5776.
- [29] ⁴³ E. A. Prastyo, "Pengertian dan Penjelasan tentang Sensor Suara," 2022. https://www.arduinoindonesia.id/2022/09/pengertian-dan-penjelasan-tentang_20.html (accessed Nov. 16, 2024).
- [30] ⁹³ M. Damirchi, "Interfacing MAX4466 Microphone Module with Arduino," 2020. <https://electropeak.com/learn/interfacing-max4466-microphone-module-with-arduino/> (accessed Nov. 16, 2024).
- [31] ⁶² J. Joseph, "Interfacing MAX4466 Microphone Amplifier Module with Arduino," 2024. <https://circuitdigest.com/microcontroller-projects/interfacing-max4466-microphone-amplifier-module-with-arduino> (accessed Nov. 16, 2024).
- [32] LastMinuteEngineers, "Interfacing MAX4466 Microphone Amplifier Module with Arduino," 2024. <https://lastminuteengineers.com/max4466-arduino-tutorial/> (accessed Nov. 16, 2024).
- [33] Admin, "Lampu Pijar," 2024. <https://www.griyasatria.co.id/lampu-pijar/> (accessed Nov. 16, 2024).
- [34] S. Liang, "Apa Itu Bola Lampu Pijar, dan Bagaimana Cara Kerjanya?," 2024. <https://vorlane.com/id/apa-itu-bola-lampu-pijar/> (accessed Nov. 16, 2024).
- [35] ⁷¹ R. P. Sari, "Internet of Things (IoT): Pengertian, Cara Kerja dan Contohnya," 2024. <https://www.cloudcomputing.id/pengetahuan-dasar/iot-pengertian-contohnya> (accessed Nov. 20, 2024).
- [36] L. Amelia, "Apa itu Internet of Things? Pengertian, Cara Kerja, dan Contohnya," *Linknet.Id*, 2023. <https://it.telkomuniversity.ac.id/internet-of-things-adalah/> (accessed Nov. 21, 2024).

- [37] L. Cloudeka, "Apa Itu Database? Apakah Penting Untuk Website?," 2023. <https://www.cloudeka.id/berita/web-dev/apa-itu-database/> (accessed Nov. 22, 2024).
- [38] I. Fahzirah *et al.*, "PENGENALAN SISTEM DATABASE : KONSEP DASAR," vol. 1, no. 4, pp. 673–678, 2024.
- [39] Y. Yusuf, "Apa Itu Website? Ini Pengertian Website dan Jenis-Jenisnya," 2024. <https://bif.telkomuniversity.ac.id/website-adalah/> (accessed Nov. 23, 2024).
- [40] Admin, "Firebase: Pengertian, Jenis, Fungsi, dan Cara Kerjanya," 2022. <https://badr.co.id/firebase-adalah/> (accessed Nov. 23, 2024).
- [41] NN, "Firebase," 2021. <https://revou.co/kosakata/firebase> (accessed Nov. 23, 2024).
- [42] C. Gupta, "Absolute Error, Relative Error, and Percentage Error," 2025. <https://eduvigyan.com/absolute-error-relative-error-and-percentage-error/> (accessed Jul. 11, 2025).
- [43] M. Asrori, W. Y. Rezika, A. T. A. Salim, B. Indarto, and R. T. Nudiansyah, "Kalibrasi Alat Ukur Temperatur dan Kelembapan Kereta Rel Diesel Elektrik," *J. Tek. Terap.*, vol. 1, no. 2, pp. 36–41, 2022, doi: 10.25047/jteta.v1i2.14.
- [44] N. Hamdani and M. Dwiputri, "Kajian Perbandingan Suhu Panas Siang Hari Tanpa Naungan Dan Di Bawah Naungan Tanaman Rambat Markisah (*Passiflora Edulis*)," *Lakar J. Arsit.*, vol. 4, no. 2, p. 94, 2021, doi: 10.30998/lja.v4i2.10322.
- [45] D. H. A. Putra, "Suhu Udara Bikin Menggigil, Ternyata Ini Penyebabnya!," 2024. [https://bmkg.go.id/press-release/?p=suhu-udara-bikin-menggigil-ternyata-ini-penyebabnya&tag=press-release&lang=ID#:~:text=Kurangnya](https://bmkg.go.id/press-release/?p=suhu-udara-bikin-menggigil-ternyata-ini-penyebabnya&tag=press-release&lang=ID#:~:text=Kurangnya%20tutupan%20awan%20pada%20malam,dingin%20terperangkap%20di%20permukaan%20bumi.) tutupan awan pada malam,dingin terperangkap di permukaan bumi. (accessed Nov. 24, 2024).

LAMPIRAN A
TAMPILAN ALAT DAN *PROTOYPE* KANDANG BURUNG



LAMPIRAN B
TAMPILAN *FIREBASE* DAN *WEBSITE*

<https://kandangpuyuh-4a179-default-rtdb.firebaseio.com>

<https://kandangpuyuh-4a179-default-rtdb.firebaseio.com/>

kebisingan: 46.83765

kelembapan: 80

suhu: 31

kandangburungpuyuh11.vercel.app

Monitoring Kandang Puyuh

Suhu

31 °C

Kelembapan

80 %

Kebisingan

47.39822 dB

Tampilan Website dan Firebase Tanggal 12 Juli 2025

Pagi Hari (06:03:00)

<https://kandangpuyuh-4a179-default-rtdb.firebaseio.com>

<https://kandangpuyuh-4a179-default-rtdb.firebaseio.com/>

kebisingan: 41.04

kelembapan: 73.105

suhu: 24.3



Siang Hari (12:05:30)

<https://kandangpuyuh-4a179-default-rtdb.firebaseio.com>

<https://kandangpuyuh-4a179-default-rtdb.firebaseio.com/>

kebisingan:	49
kelembapan:	69
suhu:	30.7



Sore Hari (17:00:10)

<https://kandangpuyuh-4a179-default-rtdb.firebaseio.com>

<https://kandangpuyuh-4a179-default-rtdb.firebaseio.com/>

kebisingan:	49
kelembapan:	72
suhu:	27



Malam Hari (23:00:00)

<https://kandangpuyuh-4a179-default-rtdb.firebaseio.com>

<https://kandangpuyuh-4a179-default-rtdb.firebaseio.com/>

kebisingan:	41
kelembapan:	79.5
suhu:	24



Dini Hari (02:00:00)

<https://kandangpuyuh-4a179-default-rtdb.firebaseio.com>

<https://kandangpuyuh-4a179-default-rtdb.firebaseio.com/>

kebisingan:	37.01
kelembapan:	83
suhu:	24



LAMPIRAN C
PEROLEHAN DATA

Data Pengukuran Tanggal 12 Juli 2025

No	Tanggal	Waktu	Suhu (°C)	Kelembapan (%)	Kebisingan (dB)	Relay	LED
1	12-7-25	6:03:00	24,30	73,10	41,00	Mati	Hijau
2	12-7-25	7:09:40	24,50	72,00	44,80	Mati	Hijau
3	12-7-25	8:00:10	25,00	72,10	45,00	Mati	Hijau
4	12-7-25	9:03:00	26,00	73,00	43,20	Mati	Hijau
5	12-7-25	10:05:00	27,10	72,00	44,00	Mati	Hijau
6	12-7-25	11:02:40	29,00	70,00	45,65	Mati	Hijau
7	12-7-25	12:05:30	30,70	69,00	49,00	Mati	Merah
8	12-7-25	13:02:20	31,00	67,00	54,36	Mati	Merah
9	12-7-25	14:05:00	31,20	68,20	54,10	Mati	Merah
10	12-7-25	15:00:03	29,80	66,00	54,00	Mati	Hijau
11	12-7-25	16:04:11	29,30	67,50	53,7	Mati	Hijau
12	12-7-25	17:00:10	27,00	72,00	57,35	Mati	Hijau
13	12-7-25	18:00:00	27,20	70,00	58,1	Mati	Hijau
14	12-7-25	19:05:00	26,40	72,10	51,28	Mati	Hijau
15	12-7-25	20:03:30	26,00	73,40	48,50	Mati	Hijau
16	12-7-25	21:08:13	26,00	74,00	47,00	Mati	Hijau
17	12-7-25	22:00:10	25,00	79,00	44,00	Mati	Hijau
18	12-7-25	23:00:00	24,00	79,50	41,00	Mati	Hijau
19	13-7-25	00:01:00	24,00	80,00	38,00	Mati	Hijau
20	13-7-25	01:00:25	24,30	84,00	36,00	Mati	Hijau
21	13-7-25	02:00:00	24,00	83,00	37,00	Mati	Hijau
22	13-7-25	03:00:03	24,00	80,60	37,20	Mati	Hijau
23	13-7-25	04:00:00	24,00	79,00	36,15	Mati	Hijau
23	13-7-25	05:03:11	24,00	73,00	39,23	Mati	Hijau

LAMPIRAN D
PERHITUNGAN KALIBRASI

KALIBRASI SUHU

Data Termometer (Referensi): X

Data Sensor DHT22 (Sensor atau Hasil) : Y

Persamaan: $\% error = \frac{|Y-X|}{X} \times 100\%$

Pengukuran Pertama:

$$\% error = \frac{|Y - X|}{X} \times 100\%$$

$$\% error = \frac{|28,70 - 29,45|}{29,45} \times 100\%$$

$$\% error = 2,55\%$$

Rata-Rata Error:

$$\frac{2,55\% + 3,93\%}{2} = 3,24\%$$

Pengukuran Kedua:

$$\% error = \frac{|Y - X|}{X} \times 100\%$$

$$\% error = \frac{|29,30 - 30,50|}{30,50} \times 100\%$$

$$\% error = 3,93 \%$$

KALIBRASI KELEMBAPAN

Persamaan: $\% error = \frac{|Y-X|}{X} \times 100\%$

Pengukuran Pertama:

$$\% error = \frac{|Y - X|}{X} \times 100\%$$

$$\% error = \frac{|Y - X|}{X} \times 100\%$$

$$\% error = 4\%$$

Rata-Rata Error:

$$\frac{4\% + 5,71\%}{2} = 4,86\%$$

Pengukuran Kedua:

$$\% error = \left(\frac{|Y - X|}{X} \right) \times 100\%$$

$$\% error = \left(\frac{|74 - 70|}{70} \right) \times 100\%$$

$$\% error = 5,71 \%$$

Menghitung Nilai *Error* Sensor Max4466

Data Sensor : X

Data Aplikasi : Y

Nilai *Error* :

$$|Y - X|$$

% error:

$$\frac{|Y - X|}{X} \times 100\%$$

Percobaan Pertama

$$Error = |Y - X|$$

$$Error = |44 - 44.2| = 0,2$$

Percobaan Kedua

$$Error = |Y - X|$$

$$Error = |46 - 47| = 1,0$$

% error:

$$\begin{aligned} &= \frac{|Y - X|}{X} \times 100\% \\ &= \frac{0,20}{44,20} \times 100\% = 0,45\% \end{aligned}$$

% error:

$$\begin{aligned} &= \frac{|Y - X|}{X} \times 100\% \\ &= \frac{1}{47,00} \times 100\% = 2,13\% \end{aligned}$$

Percobaan Ketiga

$$Error = |Y - X|$$

$$Error = |42 - 41,80| = 0,20$$

Rata-Rata Nilai Error:

Rata-Rata Error Absolut:

$$\frac{0,20 + 1 + 0,20}{3} = 0,47$$

% error:

$$\begin{aligned} &= \frac{|Y - X|}{X} \times 100\% \\ &= \frac{0,20}{41,80} \times 100\% = 0,48\% \end{aligned}$$

Rata-Rata %error:

$$\frac{0,45\% + 2,13\% + 1,20\%}{3} = 1,02\%$$

LAMPIRAN E
LISTING CODE

LISTING CODE

```
// Konfigurasi Firebase
#define FIREBASE_HOST "https://kandangpuyuh-4a179-
default-rtdb.firebaseio.com/" // Ganti dengan URL
database Firebase Anda
#define FIREBASE_AUTH
"AIZAaSyDJxyKnqAvivSFk53D700adZhrMqpFDR9E"
// Ganti dengan Authentication Key Firebase Anda

#include "DHT.h"
DHT dht(D3, DHT22);
#include <ESP8266WiFi.h>
#include <FirebaseESP8266.h>
#include <ESP8266WebServer.h>
ESP8266WebServer server(80);
#include <WiFiManager.h>
#include <WiFiClientSecure.h>
#include <LCD_I2C.h>
LCD_I2C lcd(0x27, 16, 2); // Default address of most
PCF8574 modules

const char* host = "script.google.com";
const int httpsPort = 443;
WiFiClientSecure client;
String GAS_ID =
"AKfycbxWU1S2oDQBdnBNLzcp6YULJ30WvgYRxnXgMaDcTYS6miyGk-
sn57XpTckJWJB2_6w"; //--> spreads
FirebaseData firebaseData; // Objek Firebase

uint8_t _merah = D6;
uint8_t _hijau = D7;
uint8_t _biru = D5;
int relay_lampu = D0;
```

```

const int micPin = A0; // Pin analog yang terhubung ke
sensor mikrofon
int sampleWindow = 50; // Jendela waktu sampling dalam
milidetik
unsigned long lastTime = 0;
unsigned long timerDelay = 10*60*1000;
float suhu, kelembapan, kebisingan;

void handleRoot() {
    server.send(200, "text/plain", "Hello from
WiFiManager!");
}

void connectWiFi() {
    WiFiManager wifiManager;
    // Start WiFiManager configuration portal
    if (!wifiManager.autoConnect("KANDANG PUYUH")) {
        Serial.println("Failed to connect and hit timeout");
        delay(3000);
        // Reset and try again, or maybe put it to deep sleep
        ESP.reset();
        delay(5000);
    }
    // Route for root
    server.on("/", handleRoot);
    // Start server
    server.begin();
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("TERHUBUNG");
    delay(1500);
    lcd.clear();
}

```

```

void update_display() {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("SUHU : ");
    lcd.print(suhu);
    lcd.print(" C");
    lcd.setCursor(0, 1);
    lcd.print("KELEMBAPAN : ");
    lcd.print(kelembapan);
    lcd.print(" %");
}

void baca_kebisingan() {
    unsigned long startMillis = millis(); // Catat waktu
    mulai
    unsigned int peakToPeak = 0;          // Amplitudo
    sinyal puncak-ke-puncak

    unsigned int signalMax = 0;
    unsigned int signalMin = 1023;

    // Ambil sampel selama sampleWindow milidetik
    while (millis() - startMillis < sampleWindow) {
        int sample = analogRead(micPin);
        if (sample < 1024) {
            if (sample > signalMax) {
                signalMax = sample; // Perbarui sinyal maksimal
            }
            if (sample < signalMin) {
                signalMin = sample; // Perbarui sinyal minimal
            }
        }
    }
}

```

```

peakToPeak = signalMax - signalMin;           // Hitung
puncak-ke-puncak
double voltage = (peakToPeak * 5.0) / 1024.0; //
Konversi ke voltase

// Perkiraan desibel
kebisingan = 20 * log10(voltage * 1000); // Skala ke
dB relatif
}

96
void setup() {
    pinMode(_merah, OUTPUT);
    pinMode(_hijau, OUTPUT);
    pinMode(_biru, OUTPUT);
    pinMode(relay_lampu, OUTPUT);
    Serial.begin(115200);
    dht.begin();
    lcd.begin();
    lcd.backlight();

    lcd.setCursor(3, 0);
    lcd.print("MONITORING");
    lcd.setCursor(2, 1);
    lcd.print("KANDANG PUYUH");
    delay(2000);
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("MENGHUBUNGKAN...");

    connectWiFi();
    client.setInsecure();
    Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
}

```

```

void loop() {
  update_display(); //Update tampilan lcd realtime
  kelembapan = dht.readHumidity();
  suhu = dht.readTemperature();
  baca_kebisingan();

  if (isnan(kelembapan) || isnan(suhu)) {
    Serial.println(F("Gagal Baca Data DHT sensor!"));
    return;
    digitalWrite(_merah, HIGH); //Aktif low
    digitalWrite(_hijau, HIGH);
    digitalWrite(_biru, HIGH);
  }
  lcd.setCursor(3, 0);
  lcd.print("MONITORING");
  lcd.setCursor(2, 1);
  lcd.print("KANDANG PUYUH");
  delay(2000);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("MENGHUBUNGKAN...");

  connectWiFi();
  client.setInsecure();
  Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
}

void loop() {
  update_display(); //Update tampilan lcd realtime
  kelembapan = dht.readHumidity();
  suhu = dht.readTemperature();
  baca_kebisingan();

  void loop() {

```

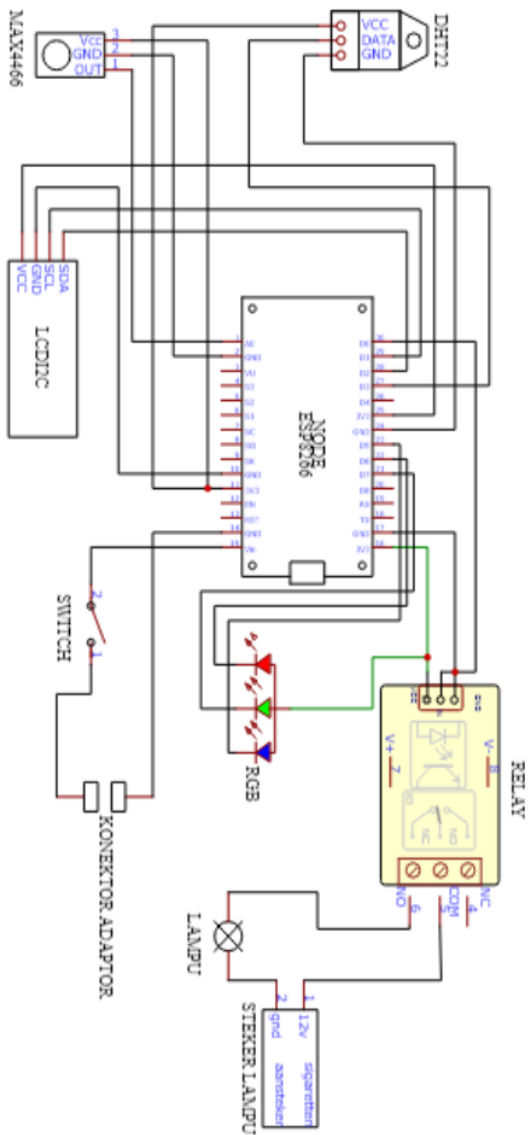
```

if (isnan(kelembapan) || isnan(suhu)) {
    Serial.println(F("Gagal Baca Data DHT sensor!"));
    return;
    digitalWrite(_merah, HIGH); //Aktif low
    digitalWrite(_hijau, HIGH);
    digitalWrite(_biru, HIGH);
}
server.handleClient();
send_firebase();
send_spreadsheet();
}

void send_firebase() { //kirim ke firebase
    Firebase.setFloat(firebaseData, "/suhu", suhu);
    Firebase.setFloat(firebaseData, "/kelembapan",
kelembapan);
    Firebase.setFloat(firebaseData, "/kebisingan",
kebisingan);
}
void send_spreadsheet() { //kirim ke spread
    if ((millis() - lastTime) > timerDelay) {
        19 if (!client.connect(host, httpsPort)) {
            Serial.println("connection failed");
            return;
        }
        String url = "/macros/s/" + GAS_ID + "/exec?suhu=" +
String(suhu) + "&kelembapan=" + String(kelembapan) +
"&db=" + String(kebisingan);
        36 client.print(String("GET ") + url + " HTTP/1.1\r\n" +
"Host: " + host + "\r\n" + "User-Agent:
BuildFailureDetectorESP8266\r\n" + "Connection:
close\r\n\r\n");
        lastTime = millis();
    }
}
}

```

LAMPIRAN F
RANGKAIAN SKEMATIK



TITLE: Skematik Hardware		REV: 1.0
Company: Universitas Sultan Agung Tirtayasa		Sheet: 1/1
Date: 2024-12-04	Drawn By: Rachel Annabel Tambun	

ORIGINALITY REPORT

22%

SIMILARITY INDEX

21%

INTERNET SOURCES

13%

PUBLICATIONS

13%

STUDENT PAPERS

PRIMARY SOURCES

1

Submitted to Universitas Sultan Ageng
Tirtayasa

Student Paper

3%

2

eprints.untirta.ac.id

Internet Source

1%

3

forum.arduino.cc

Internet Source

1%

4

123dok.com

Internet Source

1%

5

www.researchgate.net

Internet Source

<1%

6

Submitted to Middle East Technical University

Student Paper

<1%

7

shelinna.blog.pcr.ac.id

Internet Source

<1%

8

jurnal.kdi.or.id

Internet Source

<1%

9

repository.uin-suska.ac.id

Internet Source

<1%

10	wiki.unloquer.org Internet Source	<1 %
11	jurnalelectron.org Internet Source	<1 %
12	repository.ub.ac.id Internet Source	<1 %
13	journal.eng.unila.ac.id Internet Source	<1 %
14	repository.unisma.ac.id Internet Source	<1 %
15	www.cloudeka.id Internet Source	<1 %
16	Submitted to Universitas Brawijaya Student Paper	<1 %
17	eprints.itn.ac.id Internet Source	<1 %
18	ejournal.nusamandiri.ac.id Internet Source	<1 %
19	stackoverflow.com Internet Source	<1 %
20	www.bmkg.go.id Internet Source	<1 %
21	ksascholar.dri.sa Internet Source	<1 %

22	ejurnal.politeknikpratama.ac.id Internet Source	<1 %
23	pt.scribd.com Internet Source	<1 %
24	Muhammad Mukhairi Rizal, Muhammad Ikhsan, Muhammad Siddik Hasibuan. "Sistem Pemantau Suhu dan Kelembapan Kandang Puyuh Menggunakan Metode Logika Fuzzy Sugeno Berbasis Internet of Things", JURNAL FASILKOM, 2024 Publication	<1 %
25	jurnalteknik.unkris.ac.id Internet Source	<1 %
26	sipora.polije.ac.id Internet Source	<1 %
27	dspace.bracu.ac.bd Internet Source	<1 %
28	repository.dinamika.ac.id Internet Source	<1 %
29	Mardiansyah, Heri Kusnadi, Firman Amir. "Innovative Automatic Winding for Transformers: An Efficient Solution for Modern Transformer Production", ELECTRON Jurnal Ilmiah Teknik Elektro, 2025 Publication	<1 %

30	Submitted to Universitas Bengkulu Student Paper	<1 %
31	nazrul.staff.ugm.ac.id Internet Source	<1 %
32	kc.umn.ac.id Internet Source	<1 %
33	repository.usd.ac.id Internet Source	<1 %
34	www.ijair.id Internet Source	<1 %
35	id.123dok.com Internet Source	<1 %
36	Submitted to Universiti Putra Malaysia Student Paper	<1 %
37	etd.umy.ac.id Internet Source	<1 %
38	repository.its.ac.id Internet Source	<1 %
39	text-id.123dok.com Internet Source	<1 %
40	Submitted to Universitas Islam Malang Student Paper	<1 %
41	Submitted to Westmont College Student Paper	<1 %

42	pmc.ncbi.nlm.nih.gov Internet Source	<1 %
43	Dahira Unsulbar. "PROTOTYPE OF AUTOMATIC WATERGATE MONITORING SYSTEM IN DAMS BASED ON THE INTERNET OF THINGS", Jurnal Informatika dan Teknik Elektro Terapan, 2024 Publication	<1 %
44	Submitted to Universitas Islam Lamongan Student Paper	<1 %
45	www.coursehero.com Internet Source	<1 %
46	Submitted to Universitas Muria Kudus Student Paper	<1 %
47	digilib.esaunggul.ac.id Internet Source	<1 %
48	digilib.uns.ac.id Internet Source	<1 %
49	ijisae.org Internet Source	<1 %
50	Submitted to Universitas Sumatera Utara Student Paper	<1 %
51	community.blynk.cc Internet Source	<1 %

52 esp8266.ru Internet Source <1 %

53 ijpeds.iaescore.com Internet Source <1 %

54 journal.um-surabaya.ac.id Internet Source <1 %

55 tunasbangsa.ac.id Internet Source <1 %

56 Submitted to Universitas Diponegoro Student Paper <1 %

57 digilib.unila.ac.id Internet Source <1 %

58 docplayer.info Internet Source <1 %

59 ejournal.unsrat.ac.id Internet Source <1 %

60 www.neptunus58.nl Internet Source <1 %

61 www.scribd.com Internet Source <1 %

62 Submitted to Asia Pacific University College of Technology and Innovation (UCTI) Student Paper <1 %

Submitted to Politeknik Negeri Lhokseumawe

63

Student Paper

<1 %

64

Submitted to SDM Universitas Gadjah Mada

Student Paper

<1 %

65

Submitted to University of Leeds

Student Paper

<1 %

66

belajarmikrokontroller2022.blogspot.com

Internet Source

<1 %

67

gelecegiyazanlar.turkcell.com.tr

Internet Source

<1 %

68

journal.lppmunindra.ac.id

Internet Source

<1 %

69

ojs.unh.ac.id

Internet Source

<1 %

70

pasjabar.com

Internet Source

<1 %

71

repository.upnjatim.ac.id

Internet Source

<1 %

72

Submitted to Jose Rizal University

Student Paper

<1 %

73

repositori.usu.ac.id

Internet Source

<1 %

74

Submitted to Ciputra University

Student Paper

<1 %

75	Qixin Xian, Sami Salama Hussen Hajjaj. "Chapter 59 Dynamic Music-Responsive Robotics: Development of a Mobile Dancing Robot for Interactive Performance", Springer Science and Business Media LLC, 2025 Publication	<1 %
76	ejurnal.methodist.ac.id Internet Source	<1 %
77	eprints.akakom.ac.id Internet Source	<1 %
78	jurnal.utu.ac.id Internet Source	<1 %
79	mc.dfrobot.com.cn Internet Source	<1 %
80	repository.unair.ac.id Internet Source	<1 %
81	Submitted to Middlesex University Student Paper	<1 %
82	Submitted to Sogang University Student Paper	<1 %
83	Submitted to Telkom University Student Paper	<1 %
84	Submitted to United Colleges Group Student Paper	<1 %

85	j-teta.polije.ac.id Internet Source	<1 %
86	repository.upi.edu Internet Source	<1 %
87	Selvanus Edy, Edwin L. A. Ngangi, Joppy D. Mudeng. "Analisis kelayakan lahan budi daya rumput laut (<i>Ulva</i> sp.) pada lokasi rencana pengembangan North Sulawesi Marine Education Center di Likupang Timur", e-Journal BUDIDAYA PERAIRAN, 2017 Publication	<1 %
88	Submitted to Universitas Negeri Jakarta Student Paper	<1 %
89	repository.pnj.ac.id Internet Source	<1 %
90	Sakir, Agus Pramusinto, Bevaola Kusumasari, Rahmawati Husein. "Disaster-Prone Tourism Villages into Green Tourism: A Case Study of Bantul Regency", IOP Conference Series: Earth and Environmental Science, 2025 Publication	<1 %
91	ojs.trigunadharma.ac.id Internet Source	<1 %
92	repository.umsu.ac.id Internet Source	<1 %

93	Submitted to Organisation et Développement Student Paper	<1 %
94	rstudio-pubs-static.s3.amazonaws.com Internet Source	<1 %
95	www.gbspress.com Internet Source	<1 %
96	www.slideshare.net Internet Source	<1 %
97	Pambudi, Arief Prasetyo. "Monitoring Ruang Operasi Berbasis Fuzzy Logic Dengan Menggunakan IoT", Universitas Islam Sultan Agung (Indonesia), 2024 Publication	<1 %
98	eprints.umg.ac.id Internet Source	<1 %
99	wokwi.com Internet Source	<1 %
100	www.kompas.com Internet Source	<1 %
101	Submitted to Fakultas Ekonomi Universitas Indonesia Student Paper	<1 %
102	arsitektur.studentjournal.ub.ac.id Internet Source	<1 %

103	mariberbagiac.blogspot.com Internet Source	<1 %
104	repository.bakrie.ac.id Internet Source	<1 %
105	repository.unhas.ac.id Internet Source	<1 %
106	repository.unwira.ac.id Internet Source	<1 %
107	senikitin.ru Internet Source	<1 %
108	Ija, Lulu Isnaini. "Analisis Penambahan Sistem Polder Pada Daerah Aliran Sungai Gonjol Dalam Upaya Menanggulangi Banjir di Kecamatan Sayung", Universitas Islam Sultan Agung (Indonesia), 2023 Publication	<1 %
109	Krisnaningsih, Atiek Nuraini. "Sistem Kendali Pemakaian Energi Listrik Berbasis Logika Fuzzy Pada Bangunan Gedung", Universitas Islam Sultan Agung (Indonesia), 2024 Publication	<1 %
110	Sutarno, Sutarno. "Efektivitas pengekanan terhadap perilaku beton mutu tinggi berserat baja", Universitas Islam Sultan Agung (Indonesia), 2023 Publication	<1 %

111	Submitted to Universitas Maritim Raja Ali Haji Student Paper	<1 %
112	Yayan Hendrian. "Perancangan Alat Ukur Suhu Tubuh Dan Hand Sanitizer Otomatis Berbasis IOT", Jurnal Infortech, 2021 Publication	<1 %
113	bio.unsoed.ac.id Internet Source	<1 %
114	dibimbing.id Internet Source	<1 %
115	etheses.uin-malang.ac.id Internet Source	<1 %
116	idr.uin-antasari.ac.id Internet Source	<1 %
117	lib.unnes.ac.id Internet Source	<1 %
118	mchljhncryy.blogspot.com Internet Source	<1 %
119	repository.unigal.ac.id:8080 Internet Source	<1 %
120	simpel.its.ac.id Internet Source	<1 %
121	Sustin Farlinda, Nada Savira Nurjannah, Muhammad Yunus, Mudafiq Riyan Pratama.	<1 %

"Design and Development of a Web-Based Medical Record Retention Information System at Polyclinic X Jember Regency", International Journal of Health and Information System, 2023

Publication

122	Submitted to Universitas Pamulang Student Paper	<1 %
123	docobook.com Internet Source	<1 %
124	ijeecs.iaescore.com Internet Source	<1 %
125	jessicasiboro.blogspot.com Internet Source	<1 %
126	jurnal.umpwr.ac.id Internet Source	<1 %
127	repository.kemdikbud.go.id Internet Source	<1 %
128	repository.ar-raniry.ac.id Internet Source	<1 %
129	repository.ittelkom-pwt.ac.id Internet Source	<1 %
130	Syamir Muhammad , Abdul Muid , Dedi Triyanto. "RANCANG BANGUN SISTEM PEMBERI PAKAN IKAN DAN PENGUKUR pH	<1 %

- 131 Avin Nahak, Yoseph P.K Kelen, Budiman Baso, Willy Sucipto. "Sistem Otomatisasi Pakan Minum dan Monitoring Kelembaban Udara berbasis Internet of Things (IoT)", bit-Tech, 2024 $<1\%$
Publication
-

- 132 Mokhammad Saddam Yusuf, Gigih Priyandoko, Sabar Setiawidayat. "Prototipe Sistem Monitoring dan Controlling HSD Tank PLTGU Grati Berbasis IoT", Jambura Journal of Electrical and Electronics Engineering, 2022 $<1\%$
Publication
-

Exclude quotes Off
Exclude bibliography On

Exclude matches Off