



**Sistem *Monitoring* Suhu Dan Kelembaban Pada  
Kandang Burung Puyuh Berbasis IoT**

**Proyek Akhir**

**Oleh:**

**Denni Gunawan (3232101043)**

**Program Studi Teknik Instrumentasi  
Jurusan Teknik Elektro  
Politeknik Negeri Batam  
2024**

## Pernyataan Keaslian Proyek Akhir

Saya yang bertandatangan dibawah ini menyatakan bahwa isi sebagian maupun keseluruhan Proyek Akhir saya yang berjudul: "Sistem *Monitoring* Suhu Dan Kelembaban pada Kandang Burung Puyuh berbasis IoT" adalah **hasil karya sendiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diizinkan, dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.** Semua referensi yang dikutip atau dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan saya ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Batam, 10/Januari/ 2024



---

Denni Gunawan

NIM: 3232101043

## Lembar Pengesahan

Proyek Akhir disusun untuk memenuhi salah satu syarat memperoleh gelar  
Ahli Madya Teknik (AMd.T.)  
di  
Politeknik Negeri Batam

Oleh:  
Denni Gunawan (3232101043)

Tanggal Sidang: 10 Januari 2024

Disetujui oleh :

Penguji I



Ir. Muhammad Syafel Gozali, S.T.,M.T.  
NIK: 107050

Penguji II



Aditya Gautama Darmoyono, S.T., M.T  
NIK: 117180

Pembimbing I



Rahmi Mahdaliza, S.Si.M.Si  
NIK: 117195

Pembimbing II



Ir. Kamarudin, S.T., MT., IPM  
NIK: 110071

# Sistem Monitoring Suhu Dan Kelembaban Pada Kandang Burung Puyuh Berbasis IoT

## Abstrak

Di Indonesia berternak burung puyuh telah menjadi salah satu kegiatan yang populer, burung puyuh merupakan salah satu alternatif untuk peningkatan penyediaan protein hewani untuk Masyarakat. Namun, dalam proses pemeliharaan ada beberapa kendala yang harus diperhatikan seperti suhu dan kelembaban dalam mempengaruhi proses biologis dalam tubuh burung puyuh. Melalui observasi yang telah ditemukan beberapa masalah seperti pengontrolan suhu di dalam kandang dan *monitoring* suhu dan kelembaban secara manual. Maka dari itu dirancanglah sebuah proyek untuk mengatasi masalah tersebut, yang menerapkan *Internet of Things* yang berjudul “Sistem Monitoring Suhu Dan Kelembaban Pada Kadang Burung Puyuh Berbasis IoT” yang menggunakan ESP32 WROOM sebagai mikrokontroler, sensor DHT11 sebagai pembacaan data suhu dan kelembaban, dan relay sebagai pengontrol suhu. Pada perancangan ini suhu dan kelembaban dalam kandang akan di monitoring dan ditampilkan pada interface website secara *realtime* serta relay yang berfungsi untuk mengendalikan lampu dan kipas, dimana suhu  $<26^{\circ}\text{C}$  maka relay 1 akan menghidupkan lampu, Ketika suhu  $>29^{\circ}\text{C}$  relay 2 akan menghidupkan kipas. dan suhu  $26-29^{\circ}\text{C}$  kedua relay akan mati.

Kata kunci: Burung Puyuh, *Internet Of things*, Sensor DHT11, Relay, ESP32 WROOM

# ***IoT-Based Temperature and Humidity Monitoring System in Quail Cages***

## **Abstract**

*In Indonesia, quail farming has become one of the most popular activities, quail is one of the alternatives to increase the supply of animal protein for the community. However, in the rearing process there are several obstacles that must be considered such as temperature and humidity in influencing the biological processes in the quail's body. Through observations that have been found several problems such as controlling the temperature in the cage and monitoring temperature and humidity manually. Therefore, a design is designed to overcome these problems, which applies the Internet of Things entitled "Temperature and Humidity Monitoring System in Quail Sometimes Based on IoT" which uses ESP32 WROOM as a microcontroller, DHT11 sensor as temperature and humidity data reading, and relay as a temperature controller. In this design, the temperature and humidity in the cage will be monitored and displayed on the website interface in realtime and relays that function to control the lights and fans, where the temperature is  $< 26^{\circ}\text{C}$  then relay 1 will turn on the lights, when the temperature is  $> 29^{\circ}\text{C}$  relay 2 will turn on the fan. and temperatures  $26\text{-}29^{\circ}\text{C}$  both relays will turn off.*

*Keywords: Quail, Internet Of Things, DHT11 Sensor, Relay, ESP32 WROOM*

## Kata Pengantar

Segala puji dan Syukur penulis panjatkan kepada Allah SWT atas segala rahmat dan karunia-Nya. Shalawat beserta salam penulis panjatkan kepada Nabi Junjungan Alam yakni nabi besar Muhammad SAW, dengan mengucapkan “Allahumma Shalli’alaa Muhammad wa’aali aali Muhammad”. Sehingga penulis dapat menyelesaikan Laporan Akhir dengan judul “Sistem *Monitoring* Suhu Dan Kelembaban Pada Kandang Burung Puyuh Berbasis IoT”. Penulis ini diajukan untuk memenuhi syarat kelulusan mata kuliah Proyek Akhir dan Laporan. Selain itu penulis Proyek Akhir ini juga bertujuan untuk memberikan pengetahuan kepada pembacanya.

Dalam usaha menyelesaikan penulisan Laporan Tugas Akhir Ini Penulis banyak mendapatkan bantuan dari berbagai pihak. Oleh karena itu penulis ingin mengucapkan terima kasih kepada:

1. Allah SWT, karena berkat, hikmat dan penyertaanya penulis dapat menyelesaikan laporan akhir. Ini.
2. Orang tua penulis tercinta yang telah memberikan dorongan dan bantuan baik itu moral, nasehat maupun materi sehingga penulis dapat menempuh dan menyelesaikan laporan tugas akhir ini.
3. Bapak Ir.Kamarudin,.S.T.,M.T.,IPM selaku kepala program studi Teknik Instrumentasi Politeknik Negeri Batam
4. Ibu rahmi mahdaliza S.Si.M.Si. selaku dosen pembimbing proyek akhir serta *Manajer Project Based Learning* di tim monitoring kandang puyuh yang telah mendampingi dan memberikan berbagai masukan dalam buku Proyek Akhir ini.
5. Segenap Dosen Prodi Teknik Instrumentasi Jurusan Teknik Elektro Politeknik Negeri Batam yang telah membimbing dan memberikan ilmunya selama masih duduk dibanku kuliah.
6. Kawan -kawan kelas Instrumentasi B Pagi dan Tim *Project Based Learning Monitoring* Kandang Burung Puyuh atas dukungan dan bantuannya selama ini.

Semoga Allah SWT membalas kebaikan hati setiap orang yang telah bersedia membantu penulis, dan dalam penulisan Proyek Akhir ini penulis menyadari masih terdapat kekurangan untuk itu penulis memohon maaf dikarenakan keterbatasan yang dimiliki penulis baik dalam segi pengalaman maupun segi pengetahuan, sehingga penulisan laporan Proyek Akhir ini masih jauh dari kata sempurna.

Batam,

Denni Gunawan

## Daftar Isi

Pernyataan Keaslian Proyek Akhir .....	i
Lembar Pengesahan .....	ii
Abstrak.....	iii
<i>Abstract</i> .....	iv
Kata Pengantar .....	v
Daftar Isi .....	vi
Daftar Gambar.....	vii
Daftar Tabel.....	viii
Daftar Lampiran .....	ix
Bab 1. Pendahuluan .....	1
1.1    Latar Belakang.....	1
1.2    Rumusan Masalah .....	2
1.3    Tujuan .....	2
1.4    Manfaat .....	3
1.5    Batasan .....	3
Bab 2. Tinjauan Pustaka .....	4
Bab 3. Metode Pelaksanaan .....	8
3.1    Perancangan Elektrikal.....	8
3.2    Perancangan Mekanikal.....	10
3.3    Perancangan Software .....	11
3.4    Pengujian .....	15
BAB 4. Hasil dan Pembahasan .....	17
Bab 5. Kesimpulan dan Saran .....	23
Daftar Pustaka .....	24
Biodata.....	25
Lampiran.....	26

## Daftar Gambar

Gambar 1. Sensor DHT11.....	5
Gambar 2. Relay .....	6
Gambar 3. Blok Diagram Sistem .....	8
Gambar 4. Desain Elektrikal.....	9
Gambar 5. Desain Mekanikal.....	10
Gambar 6. Flowchart sistem.....	11
Gambar 7. Desain Tampilan Website.....	11
Gambar 8. Usecase Website.....	12
Gambar 9. Flowchart program DHT11 .....	13
Gambar 10. Grafik Perbandingan Pembacaan Suhu terhadap Hygrometer .....	18
Gambar 11. Grafik Perbandingan Pembacaan Kelembaban dengan Hygrometer	19
Gambar 12. Hasil Pengujian Data Suhu Terkirim ke <i>Website</i> .....	21
Gambar 13. Hasil Pengujian Data Kelembaban Terkirim ke <i>Website</i> .....	22

## Daftar Tabel

Tabel 1. Spesifikasi Sensor DHT11 .....	5
Tabel 2. <i>Wiring Schedule</i> .....	9
Tabel 3. Alat dan Bahan .....	14
Tabel 4. Pengujian Pembacaan Nilai Suhu .....	17
Tabel 5. Pengujian Pembacaan Nilai Kelembaban.....	18
Tabel 6. Pengujian Sistem Pengendali <i>Temperature</i> .....	20
Tabel 7. Hasil rekap history nilai suhu dan kelembaban pada <i>interface website</i> ..	22

## Daftar Lampiran

Lampiran 1. Etiket Mekanikal .....	26
Lampiran 2. Desain Mekanikal.....	26
Lampiran 3. Desain Elektrikal .....	27
Lampiran 4. Elektrikal Suhu Dan Relay.....	27
Lampiran 5. Data Suhu Dan Kelembaban Pada <i>Serial Monitor</i> .....	28
Lampiran 6. Data Suhu Masuk Ke <i>DataBase</i> .....	28
Lampiran 7. Data Kelembaban Masuk Ke <i>DataBase</i> .....	28
Lampiran 8. Lampiran <i>Hardware</i> .....	29

# Bab 1. Pendahuluan

## 1.1 Latar Belakang

Burung Puyuh merupakan salah satu jenis ternak unggas yang populer diternakkan oleh masyarakat sebagai penghasil telur dan daging. Kandang adalah tempat yang digunakan untuk memelihara hewan ternak, tempat itu dapat berupa wadah, bangunan, atau area yang sesuai dengan kebutuhan hewan ternak tersebut. Kandang yang nyaman juga merupakan tempat bagi burung puyuh untuk tumbuh dan berkembang biak dengan baik. Peternak burung puyuh skala rumahan di Indonesia menghadapi tantangan, terutama karena iklim tropis dan kondisi kandang puyuh yang memiliki rasio populasi tinggi dalam ruang yang terbatas. Hal ini dapat menyebabkan burung puyuh mengalami *heat stress* akibat suhu yang tinggi dan kelembapan yang meningkat, karena terhambatnya sirkulasi udara di dalam kandang puyuh. Suhu yang tinggi dapat menyebabkan penurunan produktivitas dan kualitas daging burung puyuh, sementara kelembapan yang tinggi meningkatkan risiko gangguan kesehatan dan potensi penyebaran penyakit menular serta gangguan sistem pernapasan pada unggas (Febrian widono ramadhana-2021).

Cahaya, suhu, dan kelembapan menjadi faktor eksternal yang dapat mempengaruhi pertumbuhan dan mengontrol berbagai proses biologis dalam tubuh burung puyuh. Kesulitan saat ternak puyuh salah satunya adalah membuat agar suhu dan kelembapan kandang tetap dalam kondisi stabil dan hangat. Karena burung puyuh merupakan salah satu unggas yang rentan dengan kondisi lingkungannya. Dilihat dari tingkat kelembapan kandang maupun lingkungan sekitar kandang juga merupakan faktor dalam menjaga kondisi burung puyuh (Aa Nasrulloh Abdul Aziz Saragih-2020).

Pada penelitian sebelumnya proses *monitoring* suhu dan kelembapan pada kandang burung puyuh telah banyak dikembangkan yakni “Sistem *Monitoring* Kandang Buruh Puyuh Berbasis Internet Of Things Pada Platform Node-red Menggunakan Metode *Naives Bayes*”. (Aa Zezen Zenal Abidin-2020). Pada penelitian tersebut melakukan penegembangan sistem pada *layer* aplikasi dan *layer servis* sistem berbasis data lokal yang menampung data dari *platform* dengan menggunakan metode data mining berupa metode *naives bayes* dan sensor LDR dan DHT11 untuk melakukan akusisi data suhu, kelembapan dan intesistas Cahaya, serta *mikrokontroler* seperti Arduino Mega 2560, NodeMcu ESP8266, dan aplikasi *platform* Node-Red.

Pada penelitian “Rancang Bangun Kandang Pintar Pada Burung Puyuh” (Febrian widoni ramadhana-2021). Dalam penelitian ini memanfaatkan teknologi

*internet of things* untuk *memonitoring* suhu pada kandang burung puyuh dengan menggunakan aplikasi Blynk yang terintegrasi dengan NodeMCU ESP8266, dengan menggunakan sensor DHT11 yang berfungsi sebagai pendeteksi suhu dalam kandang, dan LCD 16x2 sebagai serial monitor.

Pada penelitian “Monitoring Suhu Dan Kelembaban Berbasis *Internet Of Things (IoT)*”, pada Kandang Ayam” (Hendri bagus-2023) pada penelitian ini menggunakan GSM SIM 800L untuk monitoring dengan menggunakan koneksi jaringan GSM dan menggunakan sensor DHT11 untuk pembacaan suhu dan kelembaban pada kandang ayam, kemudian ditampilkan pada *serial monitor* LCD 16x2 dan *Thinkspeak* secara *realtime*.

Berdasarkan pemaparan diatas, maka pada penelitian ini mengembangkan Sistem *Monitoring* Suhu dan Kelembaban Berbasis IoT pada kandang burung puyuh. Pada sistem ini menggunakan sensor DHT11 untuk mendeteksi suhu dan kelembaban dalam kandang, dan menggunakan relay aktuator yang berfungsi untuk *ON* dan *OFF* perangkat yang mengontrol suhu, Dimana Ketika suhu dibawah 26 °C maka relay 1 akan meng-*ON* lampu, jika suhu diatas 29 °C maka relay 2 akan meng-*ON* kipas, dan Ketika suhu diatas 26-28 °C maka kedua relay tersebut akan *OFF*. Data suhu dan kelembaban pada kandang burung puyuh akan *dimonitoring* melalui *interface website* Dimana datanya dikirim secara *relatime* di *database* dan kemudian ditampilkan pada *website*.

## 1.2 Rumusan Masalah

Adapun rumusan masalah dalam perancangan produk “Sistem *Monitoring* Suhu dan Kelembaban pada Kandang Burung Puyuh berbasis IoT” ini, mencakup beberapa hal yaitu:

1. Bagaimana mengatur temperature kandang burung puyuh?
2. Bagaimana membangun sistem *monitoring* untuk pemantauan suhu dan kelembaban kandang burung puyuh pada jarak jauh?

## 1.3 Tujuan

Adapun tujuan yang diharapkan pada proyek akhir ini ialah:

1. Membangun sistem kendali sequencial (*ON/OFF*) kontrol untuk pengaturan *exhaust fan* dan lampu pijar
2. Perancangan *website* pada sistem *monitoring* suhu dan kelembaban dari jarak jauh secara *realtime*

#### **1.4 Manfaat**

Manfaat yang di harapkan pada proyek akhir pembuatan alat ini antara lain:

1. Peningkatan jumlah produksi serta kualitas telur puyuh
2. Memberi kemudahan dalam *memonitoring* suhu dan kelembaban kandang burung puyuh dari jarak jauh secara *realtime*.

#### **1.5 Batasan**

Dalam penulisan laporan proyek ini batasan permasalahan yang ada sebagai berikut:

1. Tidak membahas tentang parameter kesehatan puyuh
2. Jenis unggas yang diujicobakan adalah burung puyuh petelur sebanyak 10-15 ekor

## Bab 2. Tinjauan Pustaka

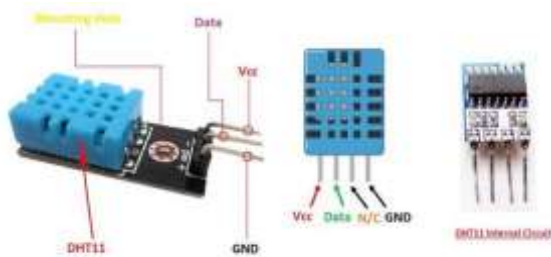
Pada tahun 2022 sebelumnya telah dirancang sebuah proyek berjudul “Rancang Bangun Sistem Pembersih Kotoran Otomatis pada Kandang Burung Puyuh Berbasis Arduino Uno” yang didalamnya juga terdapat penelitian yang membahas mengenai dampak suhu dan pakan terhadap burung puyuh. Namun, pada proyek ini masih terdapat kekurangan yaitu belum adanya penerapan sistem *Internet of Things*. Melalui acuan pembuatan alat yang ada terlebih dahulu maka dirancangkanlah sebuah proyek berjudul “Sistem Monitor dan Kontrol Pada Peternakan Burung Puyuh berbasis IoT” yang menambahkan “Sistem *Internet of Things*” didalamnya berupa website untuk monitoring suhu dan kelembaban kandang yang dilengkapi dengan sensor DHT11 sebagai pendeteksi suhu dan kelembaban dalam kandang serta relay untuk mengendalikan lampu dan kipas.

### 2.1. Sistem Pengendali Suhu

Pada tahun 2023, telah dirancang “Perancangan Sistem *Monitoring* Pada Alat Pengatur Suhu dan Kelembaban kandang Puyuh Berbasis *Internet of Things* (IoT). Dimana pada penelitian ini telah dibuat perancangan kontrol otomatis serta *memonitoring* suhu kandang burung puyuh dengan memanfaatkan teknologi internet of things menggunakan aplikasi Blynk yang terintegrasi dengan modul NodeMCU ESP8266 yang dilengkapi dengan sensor suhu DHT11 yang berfungsi sebagai pendeteksi suhu dalam kandang. dan dilengkapi dengan dengan LCD 16x2 untuk menampilkan hasil pendeteksi suhu, serta relay untuk mengendalikan kipas dan menyalakan lampu pada kandang.

#### 1. Sensor DHT11

Sensor DHT11 adalah perangkat sensor yang dilengkapi dengan kalibrasi sinyal digital untuk memberikan informasi suhu dan kelembaban. Pada perangkat ini diklasifikasikan sebagai komponen dengan Tingkat stabilitas yang sangat baik, respon pembacaannya yang cepat, dan harga yang terjangkau. Sensor DHT11 memberikan fitur kalibrasi yang sangat akurat, koefisien kalibrasi tersebut disimpan dalam program OTP, sehingga Ketika sensor internal mendeteksi suhu atau kelembaban, modul ini secara otomatis membaca koefisien sensor tersebut, DHT11 memiliki ukuran yang *relative* kecil dengan memiliki transmisi sinyal hingga 20 meter (novelan -2020). Gambar Sensor DHT11 dapat dilihat pada gambar



**Gambar 1. Sensor DHT11**

1. *Mounting hole* dapat meningkatkan stabilitas dan keamanan perangkat atau komponen, terutama dalam lingkungan yang bergetar atau bergerak.
2. Konektor atau pin: berfungsi untuk menghubungkan dengan mikrokontroler atau papan pengembangan elektronika lainnya.
3. Pin VCC VCC (Power): memberikan daya untuk sensor DHT11 sehingga dapat beroperasi.
4. Data Out (DHT-OUT): Mengirimkan data suhu dan kelembaban secara serial ke mikrokontroler atau modul yang terhubung.
5. Not Connected (NC): Pin ini biasanya dibiarkan terbuka dan tidak dihubungkan karena tidak terlibat dalam pengukuran suhu dan kelembaban.
6. Ground (GND): Memberikan jalur kembali untuk arus yang mengalir ke dalam sensor dan merupakan referensi ground untuk sinyal.

Spesifikasi dari sensor DHT11 adalah sebagai berikut

**Tabel 1. Spesifikasi Sensor DHT11**

<i>Supply voltage</i>	+5V
<i>Supply current (running)</i>	0.5A typ (2.5mA max)
<i>Supply current (stand-by)</i>	100uA typ (15uA max)
<i>Temperature range</i>	0.+50°C ±2°C
<i>Humidity range</i>	20-90%RH ±5%RH
<i>Interface</i>	Digital
Ukuran	1,05 x 0,7 (konektor tidak termasuk)
Berat	0,1 (2,7g)

## 2. Saklar

Relay adalah perangkat elektromekanis yang digunakan untuk mengendalikan sirkuit listrik dengan cara membuka atau menutup kontak listriknya berdasarkan perubahan dalam sinyal listrik yang diterimanya. Relay bekerja sebagai sakelar elektromekanis yang mengizinkan arus listrik mengalir atau terputuskan dalam sirkuit, tergantung pada apakah relay tersebut aktif atau non-aktif.



**Gambar 2. Relay**

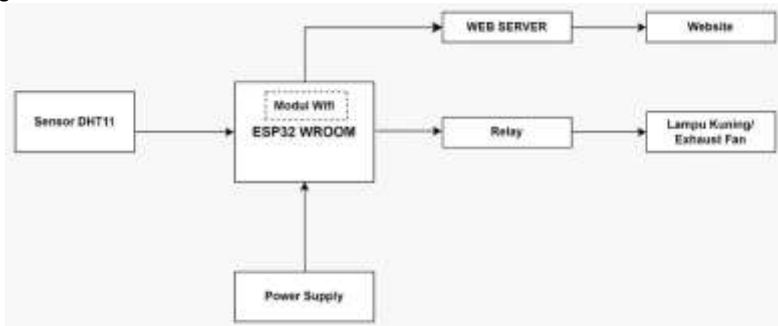
1. *Active Low Control* adalah menunjukkan bahwa relay diaktifkan ketika sinyal kontrolnya berada pada level rendah atau berada pada nilai nol (0).
2. *Signal Input* pada relay adalah sinyal kontrol yang digunakan untuk mengontrol kondisi relay, baik untuk mengaktifkan atau menonaktifkan relay. Fungsi dari sinyal input relay adalah untuk memicu perubahan dalam posisi kontak relay, yaitu untuk membuka atau menutup sirkuit listrik di bagian lain dari sistem.
3. *Power* adalah daya listrik yang diperlukan untuk mengoperasikan relay atau daya yang dapat diakibatkan oleh relay dalam kondisi tertentu
4. *Relay Switched* bekerja berdasarkan prinsip elektromagnetik. Ketika coil (gulungan kawat) di dalam relay diaktifkan oleh sinyal listrik, medan magnet yang dihasilkan menggerakkan kontak mekanis sehingga membuka atau menutup sirkuit listrik pada bagian beban.
5. C: Common (Biasa): Ini adalah kaki biasa yang terhubung dengan satu sisi dari coil (gulungan) relay.
6. NO: Normally Open (Biasa Terbuka): Kontak ini terbuka (tidak terhubung) ketika relay tidak diaktifkan.
7. NC: Normally Closed (Biasa Tertutup): Kontak ini tertutup (terhubung) ketika relay tidak diaktifkan.

Sfesifikasi dari relay adalah sebagai berikut

Supply voltage	3,75v – 6v
Quiescent current	2mA
Current when the relay is active	70 mA
Relay maximum contact voltage	250VAC / 30VDC
Relay maximum current	10A

### Bab 3. Metode Pelaksanaan

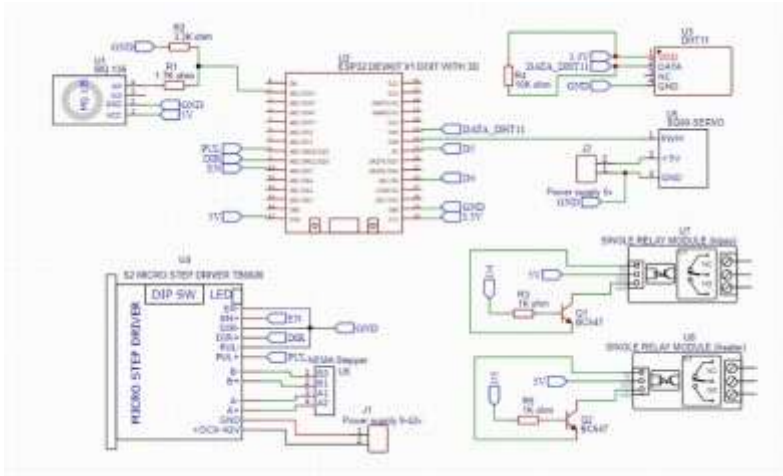
Pada bab ini akan membahas mengenai blok diagram yang merupakan alur sistem dari proyek “Sistem *Monitoring* Suhu Dan Kelembaban Pada Kandang Burung Puyuh Berbasis IoT”. Tersedia juga langkah-langkah perancangan proyek yang dimulai dari perancangan elektrikal, perancangan mekanikal, serta pengujian yang akan dilakukan dalam penelitian ini. Selain itu juga akan memaparkan komponen berupa alat dan bahan yang akan Digunakan. Blok diagram dijadikan acuan untuk membantu dalam perancangan sistem. Pada perancangan Sistem *Monitoring* Suhu Dan Kelembaban Pada Kandang Burung Puyuh Berbasis IoT menggunakan beberapa komponen seperti *power supply* sebagai sumber tegangan. ESP32-WROOM berperan sebagai mikrokontroler untuk mengelola proses data sensor. Terdapat sensor DHT11 digunakan untuk pembacaan nilai suhu dan kelembaban. Hasil dari pembacaan sensor akan diproses ESP32, Pada ESP32 akan mengelola data sensor untuk mengendalikan *temperature* melalui lampu atau kipas. Pembacaan nilai suhu dan kelembaban juga akan *dimonitoring* secara *realtime* melalui sebuah *website*. Blok diagram sistem dapat dilihat pada gambar 3.



Gambar 3. Blok Diagram Sistem

#### 3.1 Perancangan Elektrikal

Perancangan elektrikal ini dibuat menggunakan *software Easy EDA*. Sumber daya yang digunakan berupa *power supply* 24V masuk ke motor driver TB6600 untuk memberi tegangan ke motor stepper 12V. Stepdown digunakan untuk menurunkan tegangan ke 5V lalu, memberi tegangan ke mikrokontroler. Pada proyek ini, ESP32 WROOM digunakan sebagai perangkat lunak untuk memproses data yang masuk. Mikrokontroler ini menyimpan data data yang masuk melalui sensor DHT11. Kemudian menjalankan program keseluruhan sistem.



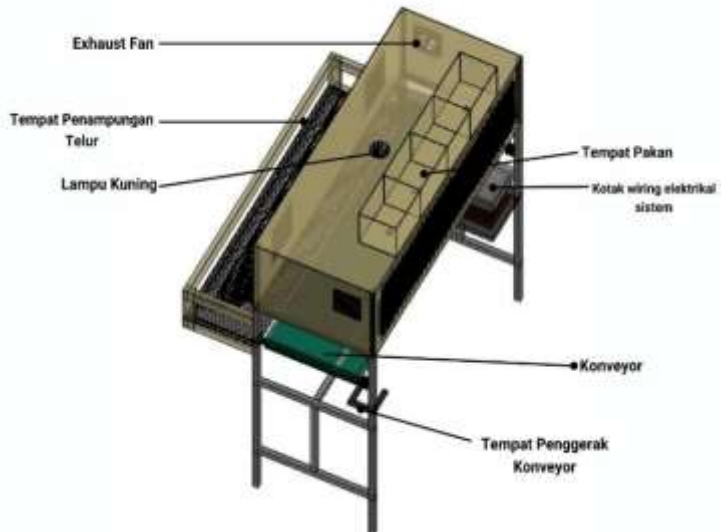
**Gambar 4. Desain Elektrikal**

Konfigurasi pin yang terhubung secara lengkap dari setiap komponen dapat dilihat melalui *wiring schedule*. *Wiring Schedule* dijadikan panduan untuk pengaturan kabel dan konektor untuk mencegah kesalahan dalam merangkai elektrikal. Tabel *wiring schedule* disajikan dibawah ini.

**Tabel 2. Wiring Schedule**

Board EP32-WROOM	Board DHT11	Board Relay 1	Board Relay 2
5V	-	VCC	VCC
GND	GND	GND	GND
3.3V	VDD-	-	-
GPIO 18	-	-	-
GPIO 19	DATA	-	-
GPIO 23	-	IN	-
GPIO 25	-	-	-
GPIO 26	-	-	-IN
GPIO 27	-	-	-
GPIO 34	-	-	-

### 3.2 Perancangan Mekanikal



**Gambar 5. Desain Mekanikal**

Dalam perancangan desain mekanikal alat dibuat menggunakan *software autocad*. Bahan yang akan digunakan untuk membangun kerangka konveyor menggunakan besi holo dan sebuah box untuk tempat komponen. Desain mekanikal ini memiliki dimensi yaitu panjang 1 meter, lebar 50cm, serta tinggi 75cm. Mekanikal ini dirangkai menggunakan mesin las untuk menyatukan setiap bagian kerangka, sementara komponen lain ada yang menggunakan bor untuk pemasangan baut, ada juga yang menggunakan kabel T untuk bagian pemasangan elektrikal.

### 3.3 Perancangan Software



Gambar 6. Flowchart sistem

Pada perancangan *software* sistem dapat dilihat pada diagram alir atau *flowchart*

#### 1. Perancangan Tampilan *Interface Website*



Gambar 7. Desain Tampilan Website

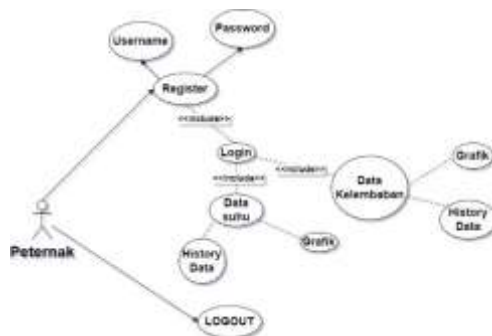
Desain tampilan yang akan dibuat pada aplikasi *Visual Studio Code*, mampu memberi informasi kepada pengguna alat untuk memudahkan monitoring suhu dan kelembaban. Tampilan memuat isi yang berguna dalam penampilan data suhu secara *realtime*.

Ada beberapa bagian dalam tampilan pada website, diantaranya adalah:

1. Kolom untuk menampilkan informasi *username* ketika sudah login atau autentikasi *login* berhasil
2. Kolom yang merupakan “Menu *sidebar navigation*”, terdiri dari menu *dashboard*, *humidity*, *temperature*, dan keluar. Fungsi dari *sidebar* untuk menavigasi ke halaman yang dituju.
3. Kolom Grafik *bar*, untuk menampilkan visualisasi grafik data dari hasil dari rekap data yang diambil berdasarkan 5 data terbaru.
4. Kolom *History* data, berfungsi untuk merekam dan menyimpan informasi *temperature* dan *humidity* terkait peristiwa atau transaksi dari waktu ke waktu. Tampilan *history* data memungkinkan pemantauan dan analisis perubahan serta perkembangan yang terjadi dalam *temperature* dan *humidity* sepanjang waktu.
5. Kolom untuk menampilkan data *temperature* terbaru.

## 2. UseCase Website

Dalam penelitian ini, penggunaan *website* hanya ditujukan untuk pengguna yang ingin memantau suhu dan kelembaban. Tampilan suhu dan kelembaban pada *website* ini hanya menampilkan rangkuman data atau Riwayat harian dari data yang telah diinputkan. Selain itu, tersedia tampilan grafik yang berfungsi sebagai tampilan visualisasi dari riwayat data suhu dan kelembaban yang masuk pada *database website* tersebut.



Gambar 8. Usecase Website

### 3. Flowchart Program

Pada flowchart program, Langkah pertama adalah melakukan inialisasi program *input* dan *output*, kemudian dilakukan proses set-point temperature, selanjutnya sensor akan membaca data sensor dan melakukan pengelolaan data, pada saat pengolahan data sensor, setpoint dari program akan berfungsi dimana Ketika suhu di bawah 26 °C maka relay 1 akan mengaktifkan lampu dan mematikan kipas sedangkan Ketika suhu diatas 29 °C maka relay 2 akan mengaktifkan kipas dan mematikan lampu serta Ketika range suhu diatas 27-28 maka lampu dan kipas akan mati. Ketika suhu tidak dideteksi oleh *mikrokontroler* maka Kembali pada proses pengambilan data, Ketika iya, maka data tersebut akan ditampilkan pada serial monitor dan proses program selesai.



Gambar 9. Flowchart program DHT11

**Tabel 3 Alat dan Bahan**

NO	Nama Komponen	Jumlah	Fungsi
1	ESP32 WROOM	1 (Satu)	Sebagai mikrokontroler, untuk mengelola sistem (Komunikasi Serial).
2	Port USB	1 Pcs	Sebagai sumber daya dari komputer serta berfungsi untuk mengunggah koding (Code) ke Arduino (Mikrokontroler).
3	Kabel AWG	4 Meter	Sebagai media untuk menghubungkan satu komponen ke komponen yang lainnya.
4	Kabel Jumper	20 Pcs	Sebagai penghubung elektrikal untuk menghubungkan tiap komponen
5	Power Supply 24	1 Pcs	Mengubah energi listrik dari sumber daya yang tersedia (seperti listrik dari sumber utama atau baterai) menjadi bentuk energi yang sesuai untuk digunakan oleh komponen-komponen elektronik dalam suatu sistem.
6	Motor Driver TB6600	1 Pcs	Berfungsi untuk mengendalikan arah dan kecepatan putaran motor listrik serta mengatur daya dan arah yang diperlukan untuk menggerakkan motor sesuai dengan kebutuhan aplikasi tertentu.
7	Motor Stepper 12V	1 Pcs	Jenis motor listrik yang bergerak dalam langkah- langkah diskrit atau langkah-langkah kecil dengan setiap impuls listrik yang diterimanya. Setiap langkah ini terkait dengan posisi tertentu dari rotor motor stepper.

8	Sensor MQ135	1 Pcs	Sensor yang berfungsi untuk membaca gas amonia (NH3)
9	Sensor DHT11	1 Pcs	Sensor yang berfungsi untuk mengukur suhu dan kelembapan ditempat sensor dipasang.
10	Motor Servo	3 Pcs	Komponen yang berfungsi untuk mendorong/memutar objek dengan kontrol yang dengan presisi dalam hal posisi sudut, aklerasi, dan kecepatan
11	Relay	2 Pcs	Berfungsi untuk mengendalikan sirkuit listrik dengan cara membuka atau menutup kontak listriknya berdasarkan perubahan dalam sinyal listrik yang diterimanya

### 3.4 Pengujian

Pengujian akan dilakukan dimulai dari proses kalibrasi masing-masing sensor. Kalibrasi diperlukan untuk membuktikan keakuratan nilai pembacaan sensor. Nilai dari pembacaan sensor akan dikelola mikrokontroler menjadi *output* untuk penggerak konveyor, pengendali suhu, serta pemberian pakan otomatis

#### 1. Pengujian Akusisi Data Sensor Suhu menggunakan ESP32

Pengujian pada sensor suhu dilakukan untuk mengetahui karakteristik sensor DHT11 terhadap pengukuran suhu dan kelembaban pada kandang burung puyuh. Suhu yang baik pada kandang burung puyuh menurut penelitian sebelumnya yang dijadikan acuan adalah 26-29 °C, dengan kelembaban 30-80%. Pada pengujian di kandang burung puyuh, dilakukan pengambilan data sensor DHT11 dan di kelola untuk mengetahui *presentase error* dengan menggunakan alat pembanding yaitu hygrometer digital, pengukuran berguna untuk mengetahui tingkat akurasi dari sensor tersebut. Hal ini didapatkan selisih pembacaan sensor DH11 dan alat pembanding dengan nilai alat pembanding kemudian dikalikan 100%. Namun, perhitungan pada excel ditambahkan fungsi ABS agar hasil perhitungan menjadi nilai yang absolut.

#### 2. Pengujian Menggunakan Komunikasi Wi-fi Dengan Hasil Data Sensor Tampil secara *realtime* pada *Interface Website*

Pada pengujian akhir, akan dilakukan komunikasi menggunakan modul *Wi-Fi* Pengujian dilakukan untuk menampilkan data dari sensor suhu yaitu sensor DHT11, Sensor suhu DHT11 akan dipasang di dalam kandang burung puyuh untuk mengukur suhu secara *kontinyu*. Data suhu yang terukur oleh sensor DHT11 akan dikirimkan melalui modul *Wi-Fi ke database*. yang akan tampil pada *website* secara

*realtime* apabila data berhasil terkirim ke *database*.Tampilan pada *website* dirancang menggunakan bahasa java pada software *Visual Studio Code*. Data sensor akan tampil pada *webservice* yang tersedia pada PC/Laptop.

## BAB 4. Hasil dan Pembahasan

### 1. Hasil pengujian pada Sensor Suhu

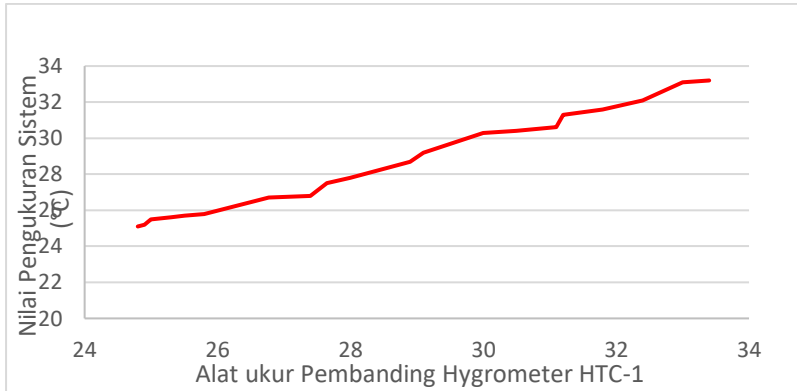
Pada pengujian selanjutnya akan dilakukan pengolahan data sensor suhu mulai dari membandingkan dengan nilai sensor dengan alat ukur pembanding hingga mengumpulkan data dari sistem pengendali *temperature* melalui relay.

#### a. Hasil pengujian perbandingan pembacaan suhu dengan alat ukur pembanding

Tabel 4. Pengujian Pembacaan Nilai Suhu

Pengujian Ke	Serial Monitor DHT11	Alat Pembanding Hygrometer HTC-1	Persentase Error (%)
	Suhu (°C)	Suhu (°C)	
1	33,2	33,4	0,60
2	33,1	33	0,30
3	32,1	32,4	0,93
4	31,6	31,8	0,63
5	31,3	31,2	0,32
6	30,6	31,1	1,61
7	30,4	30,5	0,33
8	30,3	30	1,00
9	29,2	29,1	0,34
10	28,7	28,9	0,69
11	27,8	28	0,71
12	27,5	27,65	0,54
13	26,8	27,4	2,19
14	26,7	26,77	0,26
15	25,8	25,8	0,00
16	25,7	25,5	0,78
17	25,6	25,3	1,19
18	25,5	25	2,00
19	25,2	24,9	1,20
20	25,1	24,8	1,21

<b>Rata-rata error</b>	<b>0,84</b>
------------------------	-------------



**Gambar 10. Grafik Perbandingan Pembacaan Suhu terhadap Hygrometer**

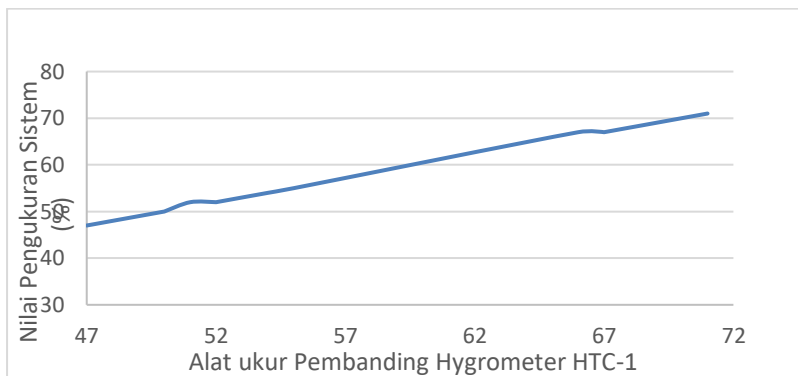
Gambar 10 merupakan grafik perbandingan nilai pembacaan suhu sensor DHT11 dengan alat pembanding hygrometer digital HTC-1. Pada grafik ini, sumbu x merupakan nilai alat ukur pembanding, sementara sumbu y adalah nilai hasil pengukuran dari sensor. Garis berwarna merah merepresentasikan perbandingan nilai suhu yang terbaca pada serial monitor, sedangkan garis berwarna coklat merepresentasikan perbandingan nilai alat ukur pembanding hygrometer digital HTC-1. Maka dari grafik tersebut dapat dilihat bahwa nilai suhu yang terbaca oleh sensor hampir mendekati nilai yang terbaca pada alat ukur pembanding.

**b. Hasil pengujian perbandingan pembacaan kelembaban dengan alat ukur pembanding**

**Tabel 5. Pengujian Pembacaan Nilai Kelembaban**

Pengujian Ke	Serial Monitor DHT11	Alat Pembanding Hygrometer HTC-1	Persentase Error (%)
	Kelembaban (%)	Kelembaban (%)	
1	47	47	0,00
2	47	47	0,00
3	48	48	0,00
4	50	50	0,00
5	50	50	0,00

6	52	51	1,96
7	52	52	0,00
8	52	52	0,00
9	54	54	0,00
10	55	55	0,00
11	67	66	1,52
12	67	67	0,00
13	67	67	0,00
14	67	67	0,00
15	68	68	0,00
16	69	69	0,00
17	70	70	0,00
18	70	70	0,00
19	70	70	0,00
20	71	71	0,00
<b>Rata-rata error</b>			<b>0,17</b>



**Gambar 11. Grafik Perbandingan Pembacaan Kelembaban dengan Hygrometer**

Gambar 11 merupakan grafik perbandingan nilai pembacaan kelembaban sensor DHT11 dengan alat pembanding hygrometer digital HTC-1. Pada grafik ini, sumbu x merupakan nilai alat ukur pembanding, sementara sumbu y adalah hasil pengukuran dari sensor. Garis berwarna biru merepresentasikan

perbandingan nilai kelembaban yang terbaca pada serial monitor, sedangkan garis berwarna coklat merepresentasikan perbandingan nilai alat ukur pembanding hygrometer digital HTC-1. Maka dari grafik tersebut dapat dilihat bahwa nilai kelembaban yang terbaca oleh sensor hampir mendekati nilai yang terbaca pada alat ukur pembanding.

**c. Hasil pengujian Relay terhadap pengendalian suhu**

Pengujian relay dilakukan untuk melihat cara kerja sistem pengendali suhu dimana apabila suhu diatas atau dibawah suhu optimal yang telah diatur pada program maka, kipas/lampu akan menyala. Pengujian relay disajikan pada tabel 6

**Tabel 6. Pengujian Sistem Pengendali *Temperature***

NO	SUHU °C	RELAY 1 KIPAS	RELAY 2 LAMPU
1	25,7	OFF	ON
2	25,5	OFF	ON
3	25,7	OFF	ON
4	25,5	OFF	ON
5	25,5	OFF	ON
6	25,5	OFF	ON
7	25,5	OFF	ON
8	25,5	OFF	ON
9	27,8	OFF	OFF
10	27,8	OFF	OFF
11	27,8	OFF	OFF
12	27,8	OFF	OFF
13	27,8	OFF	OFF
14	27,8	OFF	OFF
15	27,8	OFF	OFF
16	27,8	OFF	OFF
17	30,6	ON	OFF
18	30,6	ON	OFF
19	30,6	ON	OFF
20	30,6	ON	OFF
21	28,7	OFF	OFF

22	28,7	OFF	OFF
23	31,3	OFF	OFF
24	31,3	OFF	OFF
25	31,3	OFF	OFF

## 2. Hasil Pengujian Data Sensor DHT11 menggunakan Komunikasi Wi-fi

### a. Hasil Pengujian Data Sensor terkirim ke *website*

Gambar 12 merupakan tampilan dimana data suhu berhasil terkirim ke *interface website* terdapat nilai suhu, grafik bar, serta rekap dari temperature setiap 30 menit.



**Gambar 12. Hasil Pengujian Data Suhu Terkirim ke *Website***

Gambar 13 dibawah merupakan tampilan dimana data kelembaban berhasil terkirim ke *interface website* terdapat nilai suhu, grafik bar, serta rekap dari kelembaban yang disimpan dari data kelembaban setiap 30 menit.



**Gambar 13.** Hasil Pengujian Data Kelembaban Terkirim ke *Website*

**b. Hasil rekap *history* nilai suhu dan kelembaban dari *interface website***

Pada tabel dibawah dilakukan pengumpulan beberapa sampel data dari rekap *history* yang terdapat pada *interface*.

**Tabel 7.** Hasil rekap *history* nilai suhu dan kelembaban pada *interface website*

No	Hari, Tanggal	Waktu	TAMPILAN WEBSITE	
			Suhu (°C)	Kelembaban (%)
1	Rabu, 17 Januari 2024	20.54	27	95
2	Rabu, 17 Januari 2024	21.24	27	95
3	Rabu, 17 Januari 2024	21.54	27	95
4	Rabu, 17 Januari 2024	22.24	27	95
5	Rabu, 17 Januari 2024	22.54	27	95
6	Rabu, 17 Januari 2024	23.24	27	95
7	Rabu, 17 Januari 2024	23.54	27	95
8	Rabu, 17 Januari 2024	00.24	27	95
9	Rabu, 17 Januari 2024	00.54	27	95
10	Rabu, 17 Januari 2024	01.24	27	95

## Bab 5. Kesimpulan dan Saran

Pada bagian akhir bab dari proyek ini akan membahas mengenai kesimpulan dan saran yang didasarkan pada hasil analisis penelitian dan uraian Sebelumnya yang telah dilakukan pada proyek “Sistem *Monitoring* Suhu Dan Kelembaban Pada Kandang Burung Puyuh Berbasis IoT”. Saran dan kesimpulan yang didapat menurut penelitian yang dilakukan adalah sebagai berikut:

### A. Kesimpulan

Dari implementasi serta hasil pengujian dari proyek “Sistem *Monitoring* Suhu Dan Kelembaban Pada Kandang Burung Puyuh Berbasis IoT” diambil kesimpulan sebagai berikut:

1. Hasil akuisisi data dari sensor DHT11 memiliki perbedaan dengan pembacaan alat ukur pembanding yang terbaca yaitu hygrometer dengan rata-rata *persentase error* sebesar 0,84% pada pembacaan nilai suhu, dan rata-rata *persentase error* 0,17% pada pembacaan nilai kelembaban.
2. Pada sistem monitoring suhu dan kelembaban melalui *website*, data telah terkirim ke *website* melalui koneksi jaringan *Wi-fi* dan akan menyimpan pada *database* berupa *history* suhu & kelembaban setiap 30 menit sekali.
3. Melalui sistem ini, peternak burung puyuh dapat lebih mudah memantau dan menjaga kondisi suhu serta kelembaban dalam kandang agar sesuai dengan kebutuhan burung puyuh. Hal ini dapat meningkatkan produktivitas dan Kesehatan burung puyuh.

### B. Saran

Dari hasil pegujian “Sistem *Monitoring* Suhu Dan Kelembaban Pada Kandang Burung Puyuh Berbasis IoT” didapatkan saran sebagai berikut:

1. Pengembangan interface website agar dapat memberikan tampilan yang lebih baik terutama pada penambahan *fitur control* pada sistem keamanan suhu agar secara otomatis dapat mengaktifkan kipas dan lampu kuning secara langsung.
2. Meningkatkan *fitur* pada *website* seperti penambahan *admin* pada *website* dimana pada *fitur* sebelumnya *user* yang mendaftarkan peternak melalui *database* MySQL, sehingga perlu dikembangkan lagi pada *interface Website*.
3. Perlu dilakukan pengujian secara lebih komperehensif dan dalam jangka waktu yang lebih lama untuk mengetahui kendala dan ketahanan sistem dalam kondisi operasional sebenarnya.

Dengan pengimplementasikan saran diatas, diharapkan Sistem *Monitoring* Suhu Dan Kelembaban Pada Kandang Burung Puyuh Berbasis IoT dapat dikembangkan lebih baik dari segi pengontrolan suhu dan tampilan *fitur website*

## Daftar Pustaka

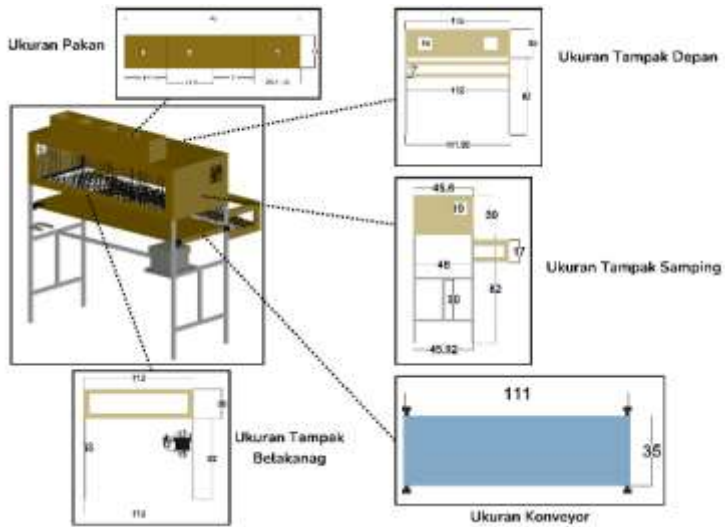
- Ramadhana, F. W., & Sigit, A. A. (2021). *Rancang Bangun Kandang Pintar Pada Burung Puyuh* (Doctoral dissertation, Universitas Muhammadiyah Surakarta).
- Abidin, A. Z. Z., & Saragih, N. A. A. (2020). Sistem Monitoring Kandang Burung Puyuh Berbasis Internet of Things pada Platform Node-RED Menggunakan Metode Naive Bayes. *Jurnal Teknologi Informasi dan Komunikasi ISSN*, 2252, 4517.
- Bagus, H., & Atiq, M. (2023). INTERNET OF THINGS (IOT) BASED TEMPERATURE AND HUMIDITY MONITORING IN CHICKEN CAGES: MONITORING SUHU DAN KELEMBABAN BERBASIS INTERNET OF THINGS (IOT) PADA KANDANG AYAM. *Jurnal Pendidikan, Elektro dan Informatika (EDUKASI ELEKTROMATIKA)*, 4(01), 6-12.
- Amin, M. And Ananda, R., 2021. Sistem Kendali Jarak Jauh Robot Pemadam Api Dengan Menggunakan Sensor Flam Dan Sensor Mq Berbasis Motor Pompa. *Journal Of Science And Social Research*, 4(2), Pp.136-141.
- Qamar, Badrul, Winarno Winarno, And Muhammad Rizal Arief. "Rancang Bangun Pembersih Kotoran Kandang Ayam Berdasarkan Berat Berbasis Arduino UNO R3." *Computing Insight: Journal Of Computer Science 1.1* (2019).
- Patonra, Arfian Habib, Et Al. "Rancang Bangun Media Pembelajaran Praktik Motor Stepper." *Mechatronics Journal In Professional And Entrepreneur (MAPLE) 2.1* (2020): 7-11.
- Alsumady, M., Et Al. "Controlling Of DC-DC Buck Converters Using Microcontrollers." *International Journal Of Circuits* 15 (2021): 197-202.
- Aa Zezen Zenal Abidin, & Nasrulloh Abdul Aziz Saragih (2020). Sistem Monitoring Kandang Burung Puyuh Berbasis Internet Of Things Pada Platform Node-Red Menggunakan Naive Bayes. *Jurnal Teknologi Informasi Dan Komunikasi*, 13(1). 2252-4517
- Jajang Winanjar & Deffy Susanti. (2021). Rancang Bangun Sistem Informasi Administrasi Desa Berbasis Web Menggunakan PHP Dan Mysql. *Jurnal Prosiding Seminar Nasional Aplikasi Sains & Teknologi (SNAST)*, 1979-911X
- Refira Dinda Cahyani Putri, Nita Rizkqi Amalia & Alfin Hidayat ,(2021). Pengembangan Infrastruktur Kandang Burung Puyuh Terintegrasi Berbasis Iot Didesa Genteng Wetan. *Jurnal (SENTRINOV) Seminar Nasional Terapan Riset Inovatif*. 7(3).

## Biodata

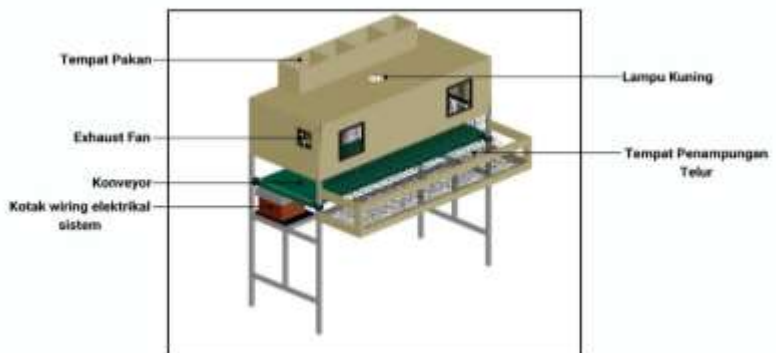


Nama : Denni Gunawan  
TTL : Natuna, 22 Oktober 2003  
Agama : Islam  
Alamat : Terapi Tepung Ajaib Pak ismail, Pantai  
Ketapang kamp.melayu, Nongsa  
Email : dennigunawan2@gmail.com  
Riwayat : SMP N 1 Palmatak  
Pendidikan SMA N 1 Palmatak

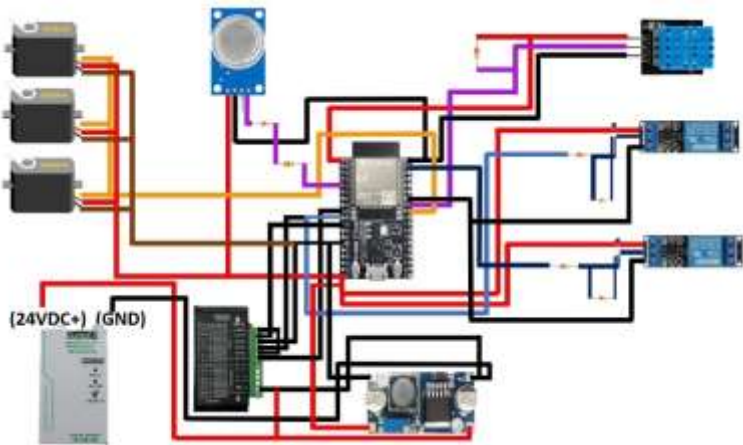
## Lampiran



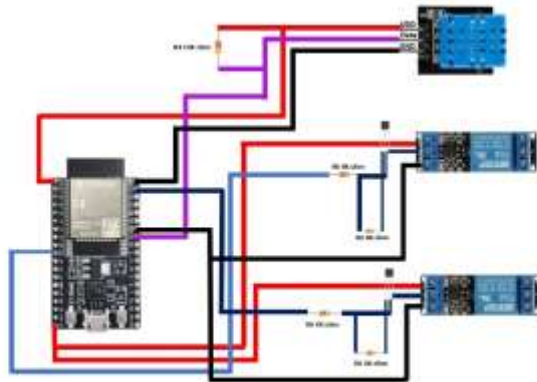
Lampiran 1. Etiket Mekanikal



Lampiran 2. Desain Mekanikal



Lampiran 3. Desain Elektrikal



Lampiran 4. Elektrikal Suhu Dan Relay





Lampiran 8. Lampiran *Hardware*

## Coding Alat Keseluruhan

```
#include <Arduino.h>
#include <WiFiManager.h> //library untuk wifimanager
#include <ArduinoJson.h> //library untuk json
#include "SendToServer.h" //library untuk mengirimkan ke server
#include "easy_WM.h" //library untuk wifimanager yang custom
#include <MQUnifiedsensor.h> //library untuk mq135
#include <Adafruit_Sensor.h> //library untuk dht
#include <DHT.h> //library untuk dht
#include <DHT_U.h> //library untuk dht
#include "delayMillis.h" //library untuk delay non block
#include <time.h> //library untuk waktu
#include <ESP32Servo.h> //library untuk servo
#define ID_Alat "001"
#define Servo_pin 18 //pin untk servo
#define PULSE_PIN 25 //pin untk stepper
#define DIRECTION_PIN 26 //pin untk stepper
#define ENABLE_PIN 27 //pin untk stepper
#define LED_PIN2 2 //LED bawaan (warna biru)
#define pinRelay1 4 //Pin Relay1
#define pinRelay2 5 //Pin Relay2
#define TRIGGER_PIN 0 //trigger untuk pin ondemand wifi
#define DHTPIN 19 //pin untk DHT, ganti sesuai selera
#define DHTTYPE DHT11 // tipe sensor DHT 11
#define Board ("ESP-32") //untuk MQ135
#define Pin (33) //pin MQ-135
#define Type ("MQ-135") //MQ135 or other MQ Sensor, if change this verify
your a and b values.
#define Voltage_Resolution (3.3) // 3V3 <- IMPORTANT. Source:
https://randomnerdtutorials.com/esp32-adc-analog-read-arduino-ide/
#define ADC_Bit_Resolution (12) // ESP-32 bit resolution. Source:
https://randomnerdtutorials.com/esp32-adc-analog-read-arduino-ide/
#define RatioMQ135CleanAir (3.6) // Ratio of your sensor, for this example an
MQ-3

float nilaiMQ135;
char serverAddr[150] = "https://httpbin.org/post";
uint32_t delayMS;
struct tm timeinfo; // Variabel untuk menyimpan waktu
volatile bool sensorSuhuFlag = 0; // Flag untuk trigger task sensor 0 = false, 1 =
true
```

```
volatile bool sensorGasFlag = 0; // Flag untuk trigger task sensor
volatile bool ServoTriggerFlag = 0; // Flag untuk trigger task actuator
volatile bool StepperTriggerFlag = 0; // Flag untuk trigger task actuator
```

```
easyWM easy_wm(TRIGGER_PIN);
SendData sendData;
DHT_Unified dht(DHTPIN, DHTTYPE);
delayMillis delayMillis_1(1000); //1000ms = 1 detik
delayMillis delayMillis_2(10000); //10000ms = 10 detik
delayMillis delayMillis_3(500); //10000ms = 10 detik
delayMillis delayMillis_4(3600000); //10000ms = 10 detik
MQUnifiedsensor MQ135(Board, Voltage_Resolution, ADC_Bit_Resolution, Pin,
Type);
Servo myservo;
```

```
String buatJSON(const char* idAlat, const char* jenisSensor, const char*
idSensor, const char* nilaiSensor, int flagbahaya) {
    const size_t capacity = JSON_OBJECT_SIZE(5);
    DynamicJsonDocument doc(capacity);
```

```
    doc["Id alat"] = idAlat;
    doc["Jenis Sensor"] = jenisSensor;
    doc["Id Sensor"] = idSensor;
    doc["nilai Sensor"] = nilaiSensor;
    doc["Peringatan"] = flagbahaya;
```

```
    String jsonString;
    serializeJson(doc, jsonString);
    return jsonString;
}
```

```
void BacaTdanH(){
    sensors_event_t event;
    dht.temperature().getEvent(&event);
    float temp = event.temperature;
    dht.humidity().getEvent(&event);
    float rel_hum = event.relative_humidity;
    int flagPeringatan = 0;
    const char* JenisSensor = "SnK";
    const char* idSensor = "TnH1";
    Serial.print("Temp: "); Serial.print(temp); Serial.print(" C");
```

```

Serial.print("\t\t");
Serial.print("Humidity: "); Serial.print(rel_hum); Serial.println(" %");
if (temp >= 27){ //lakukan sesuatu saat temperatur melebihi 30 derajat
    flagPeringatan = 1;
    sensorSuhuFlag = 1;
}
else{
    flagPeringatan = 0;
    sensorSuhuFlag = 0;
}

```

```

char buffer[100]; // Buffer harus cukup besar untuk menampung hasil
sprintf(buffer, sizeof(buffer), "%.2f,%.2f", temp, rel_hum);
const char* nilaiSensor = buffer;
if(delayMillis_4.isReady()){
    String buffData = buatJSON(ID_Alkat, JenisSensor, idSensor, nilaiSensor,
flagPeringatan); //harus dalam bentuk JSON
    Serial.println(sendData.SendDataJSON(buffData));
}

```

```

float bacaMQ135 () {
    float cFactor = 0; //ganti bagian 0 dengan angka kalibrasi untuk mengubah rasio
rs/r0
    MQ135.update(); // Update data, the arduino will read the voltage from the
analog pin
    nilaiMQ135 = MQ135.readSensor(); // Sensor will read PPM concentration
using the model, a and b values set previously or from the setup
    // if (cFactor > 0){
    // nilaiMQ135 = MQ135.readSensor(false, cFactor); //dipakai apabila faktor
koreksi digunakan untuk peningkatan akurasi
    // }
    //apabila ingin menggunakan rumus/metode kalibrasi sendiri maka
    // nilaiMQ135 = nilaiMQ135 (lakukan operasi matematika disini seperti * atau
/ atau +) variabelkoreksi
    Serial.print ("Gas: ");
    Serial.print (nilaiMQ135);
    Serial.println (" PPM");
    int flagPeringatan = 0;
    const char* JenisSensor = "Gas";
    const char* idSensor = "GAS1";

```

```

char buffer[100]; // Buffer harus cukup besar untuk menampung hasil
sprintf(buffer, sizeof(buffer), "%.2f", nilaiMQ135);
const char* nilaiSensor = buffer;
String buffData = buatJSON(ID_Alat, JenisSensor, idSensor, nilaiSensor,
flagPeringatan);//harus dalam bentuk JSON
// Serial.println(sendData.SendDataJSON(buffData));
return nilaiMQ135;
}

void myConfigModeCallback(WiFiManager* myWiFiManager) {
    easy_wm.configModeCallback(myWiFiManager);
}

void printLocalTime()
{
    if(!getLocalTime(&timeinfo)){
        Serial.println("Failed to obtain time");
        return;
    }
    Serial.println(&timeinfo, "%A, %B %d %Y %H:%M:%S");// Menampilkan waktu
dalam format yang lebih mudah dibaca
}

// Definisi fungsi untuk task sensor
void sensorTask(void *parameter) {
    for (;;) {
        // Masukkan kode untuk membaca sensor di sini
        if(delayMillis_1.isReady()){
            BacaTdanH();
            if (bacaMQ135() > 8){ //lakukan sesuatu saat nilai MQ135 melebihi 20
                sensorGasFlag = 1;
            }
            else{
                sensorGasFlag = 0;
            }
        }
    }
}

// Definisi fungsi untuk task kontrol

```

```

void controlTask(void *parameter) {
  for (;;) {
    // Masukkan kode untuk kontrol waktu/WiFi di sini
    if (WiFi.status() == WL_CONNECTED) {
      if(delayMillis_3.isReady()){
        printLocalTime(); // Menampilkan waktu lokal
      }
    }

    if(sensorGasFlag != 0){
      while (!delayMillis_2.isReady()) //selama belum lewat 10 detik maka akan
      terus berputar steppernya
      {
        digitalWrite(DIRECTION_PIN, LOW); //ubah ke HIGH untuk ganti arah
        putaran.
        digitalWrite(ENABLE_PIN, LOW);
        digitalWrite(PULSE_PIN, HIGH);
        delayMicroseconds(20);
        digitalWrite(PULSE_PIN, LOW);
        delayMicroseconds(20);
      }
    }

    //CONTOH Menjalankan aksi pada jam 14:30
    if (timeinfo.tm_hour == 17 && timeinfo.tm_min == 00 && timeinfo.tm_sec
    <= 01) { //timeinfo.tm_sec <= 01 bagian ini berfungsi memberi waktu agar servo
    bisa dieksekusi
      Serial.println("gerakan servo");
      myservo.write(30);
      delay(500); // Delay 1 detik (1000 milidetik)
      myservo.write(0);
      delay(1650); // D
      myservo.write(30); }
    if (timeinfo.tm_hour == 8 && timeinfo.tm_min == 59 && timeinfo.tm_sec
    <= 01) { //timeinfo.tm_sec <= 01 bagian ini berfungsi memberi waktu agar servo
    bisa dieksekusi
      Serial.println("gerakan servo");
      myservo.write(30);
      delay(500); // Delay 1 detik (1000 milidetik)
      myservo.write(0);
      delay(1650); // D

```

```

        myservo.write(30);
        // Tempatkan kode aksi yang ingin dijalankan di sini untuk waktu yang
        ditentukan
    }

    if (sensorSuhuFlag != 1){ //kendalikan batas sensor suhu di fungsi
    bacaTdanH()
        digitalWrite(pinRelay1, LOW); //relay 1 hidup
        digitalWrite(pinRelay2, HIGH); //relay 2 mati
    }
    else{
        digitalWrite(pinRelay1, HIGH); //relay 1 mati
        digitalWrite(pinRelay2, LOW); //relay 2 hidup
    }

    }
}

void setup() {
    // put your setup code here, to run once:
    Serial.begin(115200);
    delay(500);
    pinMode(LED_PIN2, OUTPUT);
    pinMode(PULSE_PIN, OUTPUT);
    pinMode(DIRECTION_PIN, OUTPUT);
    pinMode(ENABLE_PIN, OUTPUT);
    pinMode(pinRelay1, OUTPUT);
    pinMode(pinRelay2, OUTPUT);
    dht.begin();
    // Allow allocation of all timers
    ESP32PWM::allocateTimer(0);
    ESP32PWM::allocateTimer(1);
    ESP32PWM::allocateTimer(2);
    ESP32PWM::allocateTimer(3);
    myservo.setPeriodHertz(50); // Standard 50hz servo
    myservo.attach(Servo_pin, 500, 2400); // attaches the servo on pin 18 to the
    servo object

    // using SG90 servo min/max of 500us and 2400us
    // for MG995 large servo, use 1000us and 2000us,

```

```

        // which are the defaults, so this line could be
        // "myservo.attach(Servo_pin);"

//wifi manager
bool forceConfig = false;
bool spiffsSetup = easy_wm.loadConfigFile(serverAddrs);
if (!spiffsSetup){
    Serial.println(F("Forcing config mode as there is no saved config"));
    forceConfig = true;
}
WiFiManager wm;
wm.setDebugOutput(true);
wm.setSaveConfigCallback(std::bind(&easyWM::saveConfigCallback,
&easy_wm));
wm.setAPCallback(myConfigModeCallback);
bool res;
WiFiManagerParameter serverBox("ServerAPI", "Server API",
"https://yourServerAPIHere.com", 150);
wm.AddParameter(&serverBox);

res = wm.autoConnect("Burung puyuh", "12345678");
if(!res) {
    Serial.println("Failed to connect");
    ESP.restart();
}
else {
    //if you get here you have connected to the WiFi
    Serial.println("connected");
}
char serverAddrs_save[150];
strncpy(serverAddrs_save, serverBox.getValue(),
sizeof(serverBox.getValue()));

if (shouldSaveConfig){
    easy_wm.saveConfigFile(serverAddrs_save);
}

Serial.println(serverAddrs);
Serial.println("***");
Serial.println("WiFi connected");
Serial.println("IP address: ");
digitalWrite(LED_PIN2, HIGH);

```

```

Serial.println(WiFi.localIP());
sendData.setServer(serverAddrs);
configTime(8 * 3600, 0, "pool.ntp.org", "time.nist.gov"); //untuk WITA
//configTime(7 * 3600, 0, "pool.ntp.org", "time.nist.gov"); //untuk WIB

```

```

Serial.println("Menunggu untuk mendapatkan waktu");
while (!time(nullptr)) {
  Serial.print(".");
  delay(1000);
}

```

```

//Set math model to calculate the PPM concentration and the value of
constants

```

```

MQ135.setRegressionMethod(1); //_PPM = a*ratio^b
MQ135.setA(102.2); MQ135.setB(-2.473); // Configure the equation to to
calculate Benzene concentration

```

```

/*

```

```

  Exponential regression:

```

Gas	a	b
LPG	44771	-3.245
CH4	$2 \cdot 10^{31}$	19.01
CO	521853	-3.821
Alcohol	0.3934	-1.504
Benzene	4.8387	-2.68
Hexane	7585.3	-2.849
NH4	102.2	-2.473

```

*/

```

```

MQ135.init();
Serial.print("Calibrating please wait.");
float calcR0 = 0;
for(int i = 1; i<=10; i++)
{
  MQ135.update(); // Update data, the arduino will read the voltage from the
analog pin
  calcR0 += MQ135.calibrate(RatioMQ135CleanAir);
  Serial.print(".");
}
MQ135.setR0(calcR0/10);
Serial.println(" done!");

```

```

if(isinf(calcR0)) {Serial.println("Warning: Conection issue, R0 is infinite (Open
circuit detected) please check your wiring and supply"); while(1);}
if(calcR0 == 0){Serial.println("Warning: Conection issue found, R0 is zero
(Analog pin shorts to ground) please check your wiring and supply"); while(1);}
/**** MQ CALibration *****/

```

```

// Membuat task untuk membaca sensor

```

```

xTaskCreatePinnedToCore(
    sensorTask, /* Fungsi yang menjalankan task */
    "SensorTask", /* Nama task */
    10000, /* Ukuran stack (dapat disesuaikan sesuai kebutuhan) */
    NULL, /* Parameter yang dilewatkan ke task */
    1, /* Prioritas task */
    NULL, /* Task handle */
    0 /* Core di mana task akan dijalankan */
);

```

```

// Membuat task untuk kontrol waktu/WiFi

```

```

xTaskCreatePinnedToCore(
    controlTask,
    "ControlTask",
    10000,
    NULL,
    1,
    NULL,
    1
);

```

```

}

```

```

void loop() {

```