

SISTEM MONITORING DAN KONTROL SANGKAR BURUNG MURAI BATU

¹Mohamad Khafil Hadi Mubaroq, ²M. Ibrahim Ashari, ³Irmalia Suryani Faradisa.

Institut Teknologi Nasional, Malang, Indonesia

Khafilhadi24@gmail.com

Abstract— Sistem monitoring sangkar burung otomatis merupakan sebuah sistem yang difungsikan agar mempermudah pemantauan sangkar burung otomatis. Yang mana membantu pengguna dalam memonitoring kondisi sangkar burung, seperti suhu sangkar, kelembapan sangkar, ketesediaan pakan, ketersediaan minum, kondisi saat atap terbuka atau tertutup, kondisi saat burung mandi atau tidak dan kondisi saat melatih suara burung yang dapat dilakukan secara jarak jauh. Kondisi pakan dapat diketahui dengan menggunakan sensor Loadcell, kondisi minum dapat diketahui dengan menggunakan sensor Water Level, kondisi saat atap terbuka atau tertutup dengan menggunakan motor Stepper, Kondisi saat burung mandi dengan menggunakan sensor DHT11, kondisi saat melatih suara burung dengan menggunakan Mp3Player yakni akan diintegrasikan dengan mikokontroler ESP32. Data yang diterima ESP32 dari sensor akan dikirimkan ke *we Thinger.io* sebagai display monitoring.

Kata Kunci— *Monitoring, Sangkar Burung Otomatis, ESP32, Thinger.io, Burung Murai Batu*

I. PENDAHULUAN

Sejak dulu masyarakat Indonesia telah mengenal dan melakukan sebuah hobi yaitu memelihara burung. Terdapat banyak sekali ragam dan jenis burung di Indonesia, terutama burung kicau yang banyak sekali dipelihara oleh masyarakat karena memiliki suara merdu dan indah yang menjadi hiburan tersendiri bagi pemeliharanya. Akan tetapi semakin lama burung kicau tidak hanya sekedar untuk hiburan saja, melainkan juga menjadi sebuah hobi, bisnis dan bahkan di perlombakan suara kicauannya.[1] Salah satu burung kicau yang paling populer di Indonesia adalah Burung Murai Batu (*Copsychus malabaricus*), burung ini termasuk ke keluarga Turdidae yaitu burung dengan ciri khas kemampuan berkicau yang sangat merdu dan indah serta suara kicauannya yang sangat bervariasi. Burung murai batu merupakan burung mudah untuk beradaptasi, pemalu dan relatif jinak apabila dalam pemeliharannya dilakukan dengan baik dan tepat.[2]

Dalam memelihara burung murai batu hal perlu dilakukan sangatlah mudah yaitu dengan menyediakan sangkar burung, memberi makan, minum, memandikan burung, dan melatih suara burung, namun

untuk menghasilkan burung murai batu dengan kualitas yang bagus dan suara kicauan yang indah maka harus memperhatikan dalam melakukan pewartannya.[3]

Selama ini sangkar burung untuk burung murai batu yang ada masih menggunakan metode manual dalam melakukan pemberian pakan, pemberian minum, memandikan burung, menjemur burung, dan melatih suara burung.[4] Sangkar burung harus dilakukan pengecekan kondisinya secara manual setiap hari seperti kondisi ketersediaan pakan, minum, dan memandikan burung, serta melatih suara burung. Namun terdapat permasalahan yaitu pemelihara tidak dapat melakukan pengecekan kondisi sangkar setiap karena terganggu dengan aktifitas lainnya atau saat sedang tidak berada di rumah dan sedang bepergian jauh yang membuat pemelihara burung murai batu tidak dapat melakukan pengecekan kondisi sangkar dan melakukan perawatan. Maka dari itu perlu dibuat sebuah sangkar burung yang dapat melakukan perawatan dan pemantauan serta kontrol terhadap sangkar burung murai batu yang dapat diakses pemelihara burung murai batu darimana saja dan kapan saja, saat tidak dapat melakukan perawatan dan pemantauan secara langsung.

Telah dilakukan penelitian sebelumnya oleh Mayda Waruni Kasrani pada tahun 2018 Melakukan pengembangan otomasi tempat makan dan minum menggunakan arduino dan GSM 900. Dimana pada penelitian ini sensor HCSR-04 digunakan untuk mendeteksi makan dan minum pada wadah, kemudian hasil pengukuran sensor dikirimkan ke sms menggunakan modul sim 800L. Namun pada penelitian ini hanya terbatas untuk otomasi makan dan minum saja dan untuk hasil otomasi alatnya hanya bisa dimunculkan melalui text saja.[5]

Penelitian selanjutnya oleh Mucthar pada tahun 2021 telah mengembangkan sebuah sangkar cerdas berbasis iot menggunakan mikrokontroler Arduino Mega 2560 dan Modul ESP8266-01, serta menggunakan sensor Infrared IR dan sensor ultrasonik HC-SR04 untuk mendeteksi pakan, sensor water level untuk mendeteksi minum. Monitoring yang di pakai berbasis web Namun penelitian ini hanya di khusukan untuk burung tiriep dan sangkar yang digunakan

berukuran besar karena di peruntukan untuk penangkaran burung tiriep.[6]

Juga pada penelitian selanjutnya oleh Darlis & Haryanti pada tahun 2021 dengan menggunakan sistem monitoring sangkar burung dengan NodeMCU ESP8266 dan sensor ultrasonik HC-SR04 untuk mendeteksi pakan dan minum. Sistem monitoringnya berbasis web server menggunakan PhpMyAdmin. Namun penelitian ini hanya dikhususkan untuk burung love bird saja dan hanya berfokus untuk melakukan otomasi sangkar burung pada pemberian pakan, minum, dan memandikan burung tidak adanya sistem otomasi untuk membuka dan menutup atap sangkar burung.[7]

Pada pemaparan penelitian terdahulu diatas terdapat beberapa kekurangan diantaranya tidak adanya sistem untuk memandikan burung secara otomatis dan membuka atap secara otomatis serta hanya dikhususkan untuk burung jenis tertentu saja dan belum ada sistem monitoring yang menggunakan web server platform thinger.io.

Berdasarkan masalah diatas, penulis ingin membuat Sistem Monitoring dan Kontrol Sangkar Burung Murai Batu untuk memudahkan pengguna dalam memantau dan mengawasi serta mengontrol sangkar burung peliharaannya secara jarak jauh. Dengan melakukan perancangan sistem untuk monitoring sangkar burung murai batu berbasis web server menggunakan platform Thinger.io. Konfigurasi sistem ini mudah disesuaikan untuk mengumpulkan informasi data seperti ketersediaan pakan, minum, suhu dan kelembapan, serta kontrol buka tutup atap sangkar burung, kontrol On/Off water pump saat mandi burung dan kontrol On/Off pada MP3DPlayermini saat latih suara burung.

II. TINJAUAN PUSTAKA

Telah banyak jurnal penelitian mengenai pembuatan ataupun pengembangan otomasi pada sangkar burung dengan tujuan membantu pemelihara burung dalam melakukan perawatan burung dan memantau kondisi sangkar burung. Dengan menggunakan sensor-sensor yang diintegrasikan dengan mikrontroller membuat perawatan burung pada sangkar burung di lakukan secara otomatis dan melakukan monitoring melalui website. Namun dari banyak penelitian perancangan otomasi sangkar burung memiliki kekurangan yang sama yaitu tidak adanya sistem membuka dan menutup atap saat menjemur burung, serta tidak adanya penelitian otomasi sangkar burung yang khusus untuk burung Murai Batu.

Berikut adalah beberapa penelitian yang telah dilakukan sebelumnya dan memiliki tema penelitian yang sama dengan penelitian yang akan dilakukan :

Mayda Waruni Kasrani pada tahun 2018, Melakukan pengembangan otomasi tempat makan dan minum menggunakan arduino dan GSM 900. Dimana

pada penelitian ini sensor HCSR-04 digunakan untuk mendeteksi makan dan minum pada wadah, kemudian hasil pengukuran sensor dikirimkan ke sms menggunakan modul sim 800L. Namun pada penelitian ini hanya terbatas untuk otomasi makan dan minum saja dan untuk hasil otomasi alatnya hanya bisa dimunculkan melalui text saja.[5]

Mucthar pada tahun 2021 telah mengembangkan sebuah sangkar cerdas berbasis iot menggunakan mikrokontroler Arduino Mega 2560 dan Modul ESP8266-01, serta menggunakan sensor Infrared IR dan sensor ultrasonik HC-SR04 untuk mendeteksi pakan, sensor water level untuk mendeteksi minum. Monitoring yang di pakai berbasis web Namun penelitian ini hanya di khususkan untuk burung tiriep dan sangkar yang digunakan berukuran besar karena di peruntukan untuk penangkaran burung tiriep.[6]

Penelitian selanjutnya oleh Darlis & Haryanti pada tahun 2021 dengan menggunakan sistem monitoring sangkar burung dengan NodeMCU ESP8266 dan sensor ultrasonik HC-SR04 untuk mendeteksi pakan dan minum. Sistem monitoringnya berbasis web server menggunakan PhpMyAdmin. Namun penelitian ini hanya dikhususkan untuk burung love bird saja dan hanya berfokus untuk melakukan otomasi sangkar burung pada pemberian pakan, minum, dan memandikan burung tidak adanya sistem otomasi untuk membuka dan menutup atap sangkar burung.[7]

Zamani pada tahun 2022 Telah mengembangkan otomasi pengisian makan dan minum burung menggunakan ESP32 dan web sever untuk monitoring. Sensor HCSR-04 digunakan untuk mendeteksi makan dan minum pada wadah, kemudian data hasil dari sensor dikirim ke web server lalu diolah menjadi tampilan monitoring pada android. Namun pada penelitian ini sistem otomasi hanya berfokus pada otomasi makan dan minum saja dan monitoring hanya terbatas untuk di android saja.[8]

Adi Wibowo pada tahun 2022 Juga telah melakukan penelitian dengan sebuah sistem smart farming dalam penangkaran burung kicau berbasis website, menggunakan mikrokontroler Arduino Mega 2560 dan modul ESP8266-01 serta sensor ultrasonik HCSR-04 untuk mendeteksi pakan pada penampungan, sensor infrared untuk mendeteksi pakan pada wadah pakan, sensor water level untuk mendeteksi air pada wadah minum dan sensor DHT11 untuk mendeteksi suhu sangkar burung. Namun pada penelitian ini dikhususkan untuk sangkar burung yang berukuran besar karena diperuntukkan sebagai penangkarang burung kicau.[9]

A. Burung Murai Batu

Burung Murai Batu (*Copsychus malabaricus*) adalah jenis burung kicau yang merupakan keluarga burung turdidae yang dimana pada dasarnya memiliki warna yang beragam dan pola yang unik. Selain itu burung murai batu juga memiliki kemampuan yang sangat baik dalam berkicau, kicauannya sangatlah indah, bermelodi dan memiliki berbagai variasi kicauan. Burung ini memiliki ciri yaitu ukuran tubuhnya yang rata-rata sedang, dengan bentuk kepala yang bulat, panjang kaki yang lumayan panjang, bentuk paruhnya runcing, serta bentuk tubuhnya ramping dan sayapnya lebar.

Umumnya pada habitatnya di alam, burung murai batu bisa ditemukan pada daerah dataran rendah hingga daerah yang memiliki ketinggian lebih dari 1.000 mdpl. Burung murai batu memiliki nama dan jenis yang sangat bervariasi sesuai tempat burung itu berasal, seperti dari Sumatera ada burung murai batu medan, burung murai batu Aceh, terdapat juga jenis lain diantaranya adalah burung murai batu Borneo, burung murai batu Irian dan burung murai batu Jawa.[10] Harga untuk burung murai Batu berkisar dari ratusan ribu hingga jutaan rupiah.

B. Monitoring

Pemantauan atau monitoring adalah proses pengumpulan dan analisis informasi secara sistematis dan terus menerus berdasarkan indikator yang telah ditetapkan tentang kegiatan program sehingga dapat diambil tindakan korektif untuk lebih meningkatkan kegiatan program. Pemantauan dapat digambarkan sebagai mengenali apa yang ingin Anda ketahui. Pemantauan tingkat tinggi dilakukan dengan melakukan pengukuran dari waktu ke waktu yang mengindikasikan pergerakan menuju atau menjauhi target. Tujuan dari pemantauan adalah untuk mengamati/mengetahui perkembangan dan kemajuan, mengidentifikasi masalah dan mengantisipasi atau menyelesaikannya.[11] Monitoring adalah evaluasi kegiatan yang dilakukan sesuai dengan rencana, identifikasi masalah yang muncul agar dapat segera diatasi, evaluasi pola kerja dan manajemen untuk mencapai tujuan, dan penentuan hubungan antara kegiatan dan tujuan. Ini adalah langkah untuk mengambil tindakan. kemajuan.

C. ESP32

ESP32 adalah mikrokontroler SoC (System on Chip) terintegrasi dengan WiFi 802.11 b/g/n, Bluetooth versi 4.2 dan berbagai periferal. Mikrokontroler ini menggunakan chip dengan mikroprosesor Xtensa LX6 32-bit dual-core. Mikrokontroler ini memiliki ADC 18-pin (12-bit), empat SPI, dan dua I2C. Keunggulan mikrokontroler ini adalah harganya yang relatif murah, mudah diprogram, memiliki jumlah pin I/O yang

memadai, dan memiliki adaptor WiFi built-in untuk mengakses jaringan Internet..[12]

D. Internet Of Things

Internet of Things (IoT) adalah jaringan yang menghubungkan objek-objek yang berbeda dengan identitas pengenal dan alamat IP sehingga mereka dapat berkomunikasi dan berbagi informasi tentang diri mereka sendiri dan lingkungan yang mereka rasakan. Objek-objek dalam IoT dapat mengonsumsi atau menghasilkan layanan dan bekerja sama satu sama lain untuk mencapai tujuan bersama. Dengan kemampuan ini, IoT telah menggeser definisi Internet sebagai komputasi kapan saja, di mana saja, menjadi apa saja, siapa saja, dan layanan apa saja. Salah satu implementasi sifat yang terkait dengan identifikasi objek .[13]

Sebuah makalah akademis McKinsey Global Institute menyatakan, "Internet of Things menghubungkan mesin, perangkat, dan objek fisik lainnya ke sensor dan aktuator jaringan untuk menerima data dan mengelola kinerjanya sendiri." Ini adalah teknologi yang memungkinkan mesin untuk diaktifkan." Mereka dapat bekerja sama satu sama lain dan bahkan bereaksi secara independen terhadap informasi baru. Internet of Things sebenarnya diperkenalkan oleh seorang pengusaha teknologi asal Inggris, Kevin Ashton. Istilah "Internet of Things" pertama kali digunakan sebagai judul presentasi di seminarnya untuk pemasok komoditas AS, Procter & Gamble (P&G). Kevin menjelaskan bahwa IoT adalah sebuah sistem yang memungkinkan benda-benda fisik terhubung ke internet melalui sensor yang ada di mana-mana.[14]

E. Thinger.IO

Thinger.io adalah platform Internet of Things (IoT) yang menyediakan kemampuan cloud untuk menghubungkan berbagai perangkat yang terhubung ke internet. Thinger.io juga dapat memvisualisasikan pembacaan sensor dalam bentuk nilai atau grafik. Thinger.io mendukung semua jenis board seperti Arduino, ESP 8266, Raspberry Pi, Intel Edison. Thinger.io juga menyediakan semua alat yang Anda butuhkan untuk membuat prototipe, skala, dan mengelola produk Anda dengan mudah. Thing bersifat open source.[15] Thinger.io memiliki beberapa protokol diantaranya adalah MQTT, HTTP dan IOTMP.

F. Wireless fidelity (Wi-Fi)

Wireless Fidelity (Wi-Fi) adalah teknologi yang memungkinkan beberapa komputer terhubung melalui jaringan area lokal nirkabel (WLAN). Wireless LAN (WLAN) adalah teknologi LAN yang menggunakan frekuensi radio dan transmisi sebagai media untuk menggantikan fungsi kabel di area

tertentu. WLAN umumnya digunakan sebagai titik distribusi tingkat pengguna akhir di satu atau lebih perangkat yang disebut titik akses (AP), yang berfungsi mirip dengan hub dalam terminologi jaringan kabel Ethernet. Pada tingkat backbone, sejumlah besar AP masih terhubung melalui media kabel. WiFi ditujukan sebagai solusi media alternatif untuk menjangkau pengguna yang tidak terlayani oleh jaringan kabel, serta untuk mendukung pengguna yang bergerak atau berpindah-pindah (mobilitas).[16] Wi-fi digunakan sebagai akses internet untuk menghubungkan antara device dan web server monitoring. Seperti pada penelitian ini yang membutuhkan Wi-Fi (akses internet) untuk menghubungkan ESP32 dengan web server Thingier.io sebagai tampilan monitoring.

G. Sensor DHT11

Sensor DHT11 terdiri dari banyak komponen sensor dan IC pengontrol yang dikemas dalam satu paket. Sebagian sensor ini memiliki 4 pin dan sebagian lagi memiliki 3 pin. Tetapi dalam aplikasi Anda tidak ada perbedaan sehingga tidak masalah. Di dalam badan sensor yang berwarna biru atau putih terdapat sebuah resistor tipe NTC (koefisien suhu negatif).[6]

H. Potensiometer

Potensiometer adalah jenis resistor yang resistansinya dapat diatur sesuai dengan kebutuhan rangkaian elektronik atau kebutuhan pengguna. Potensiometer adalah jenis resistor yang termasuk dalam kategori resistor variabel. Secara struktural, potensiometer terdiri dari tiga buah kabel dengan poros atau tuas yang berfungsi sebagai pengatur.[17]

I. Router

Router adalah perangkat jaringan yang dapat digunakan untuk menghubungkan dua jaringan area lokal yang memiliki protokol yang sama pada lapisan jaringan OSI tetapi protokol yang berbeda pada lapisan fisik dan link. Atau, Anda dapat mengartikannya sebagai perangkat keras yang dapat menghubungkan dua atau lebih jaringan dengan subnet yang berbeda. Router memiliki kemampuan untuk meneruskan data ketika ada perbedaan di beberapa bagian protokol.[18]

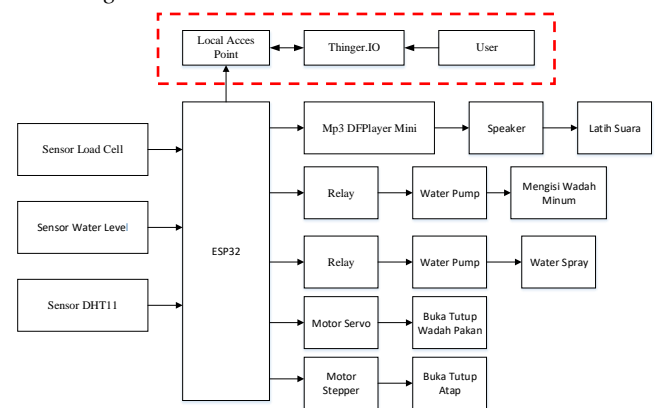
J. LED (Light Emitting Diode)

Untuk kontrol buka atau tutup atap, mandi burung dan latih suara dimisalkan menggunakan led. Website thingier.io akan melakukan kendali lampu LED yang akan mengirimkan sebuah data ke server kemudian server akan melanjutkan data tersebut ke arduino untuk melakukan pembacaan perintah yaitu ON atau OFF.

III. METODOLOGI PERANCANGAN SISTEM

Pada bagian ini akan membahas mengenai perancangan monitoing berupa perancangan software.

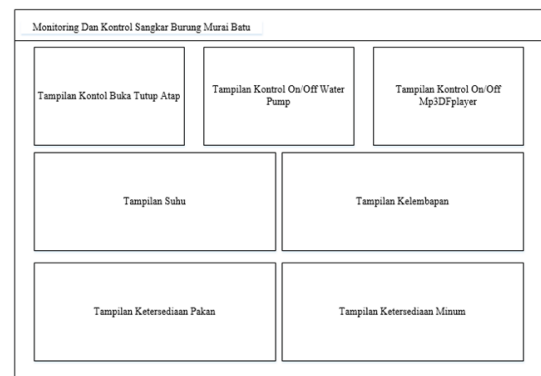
A. Diagram Blok Keseluruhan



Gambar 3.1 Diagram Blok

Pada blok diagram diatas saya hanya membahas untuk sistem monitoring dan kontrol yang mana sesuai dengan judul yaitu “Sistem Monitoring Dan Kontrol Sangkar Burung Murai Batu”. Pada diagram blok ditandai dengan kotak bergaris putus-putus yang menandakan sistem monitoring dan kontrol. Prinsip kerjanya setelah semua alat bekerja dan sensor membaca nilai sensor selanjutnya data sensor tersebut akan dikirimkan ke web server (Thingier.io) yang kemudian akan muncul di halaman tampilan monitoring dan kontrol.

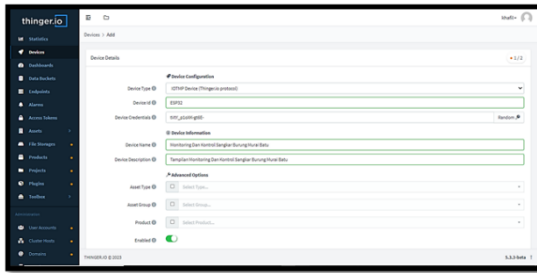
B. Perancangan Tampilan Monitoring Dan Kontrol



Gambar 3.2 Rancangan Tampilan Monitoring Dan Kontrol

Pada gambar 3.2 perancangan tampilan monitoring dan kontrol sangkar burung murai batu diatas di bagi menjadi dua bagian yaitu, bagian pertama adalah tampilan kontrol yang ditunjukkan pada tiga kotak di bagian atas yang terdiri dari tampilan kontrol buka tutup atap, tampilan kontrol On/Off Water Pump dan tampilan kontrol On/Off Mp3DFPlayer. Kemudian tampilan bagian kedua adalah tampilan monitoring suhu, kelembapan, ketersediaan pakan dan ketersediaan minum yang ditunjukkan pada gambar kotak dibagian tengah dan bawah.

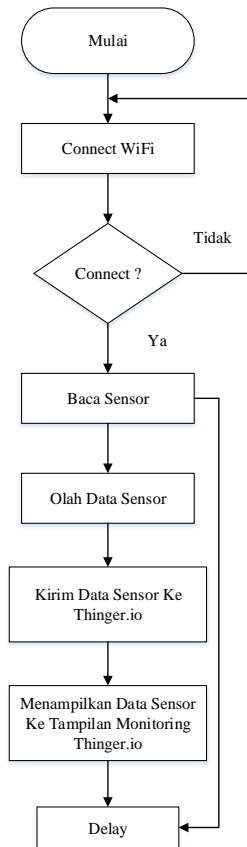
C. Perancangan Sistem Monitoring Dan Kontrol Via Thingier.IO D. Flowchart



Gambar 3.3 Konfigurasi Device ESP32 Pada Thingier.IO

Pada gambar 3.3 menunjukkan konfigurasi device ESP32 dengan Thingier.IO. Dimana terdapat beberapa settingan konfigurasi device diantaranya adalah :

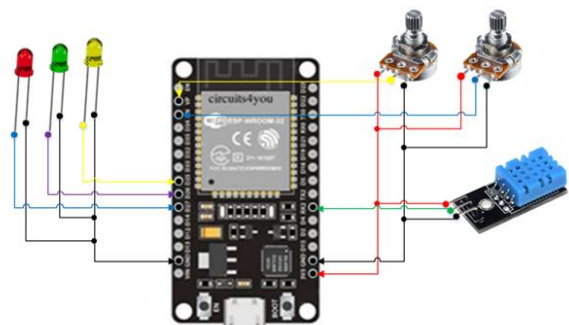
1. *Device Configuration*
 - a) *Device type* yang berfungsi untuk memilih jenis protokol, terdapat beberapa protokol yang dapat digunakan diantaranya adalah IOTMP Device (Thingier.IO), HTTP Device (Sigfox, Lora, cURL) dan MQTT Device.
 - b) *Device ID* berfungsi sebagai pengalamtan atau *adres* pada *device*.
 - c) *Device Credentials* berfungsi sebagai *password device* dalam komunikasi agar terkoneksi dengan baik.
2. *Device Information*
 - a) *Device Name* berfungsi sebagai penamaan pada *device*.
 - b) *Device Description* berfungsi sebagai penambahan deskripsi pada *device*.
3. *Thingier.IO Configuration*
 - a) *Device type* : IOTMP Device (Thingier.IO)
 - b) *Device ID* : ESP32
 - c) *Device Credentials* : tVty_p1oXK-gt6e-
 - d) *Device Name* : Monitoring dan kontrol sangkar burung murai batu
 - e) *Device Description* :Tampilan monitoring dan kontrol sangkar burung murai batu



Gambar 3.4 Flowchart Keseluruhan Sistem

Pada bagian connect wifi bertujuan untuk mengetahui apakah perangkat sensor telah terhubung ke web thingier.io, setelah perangkat sensor terhubung ke web thingier.io data yang telah dibaca oleh sensor akan dikirimkan ke thingier.io. Kemudian data tersebut akan ditampilkan pada dashboard monitoring pada thingier.io.

E. Perancangan Hardware Prototype



Gambar 3.5 Rangkaian Skematik Prototype

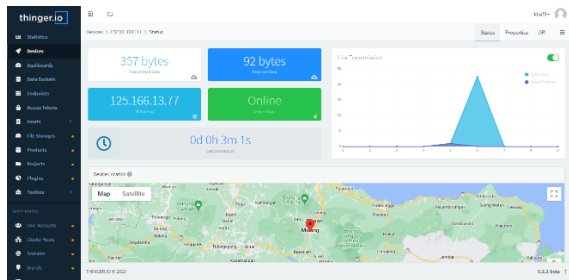
Pada gambar 3.5 menunjukkan rangkaian skematik dari prototype yang akan dibuat. Dimana terdapat esp32 sebagai mikrokontroler, sensor DHT11 sebagai pendeteksi suhu dan kelembapan, potensiometer sebagai indikator pendeteksi pakan dan

minum, dan LED sebagai indikator kontrol kondisi On/Off.

IV. HASIL DAN PEMBAHASAN

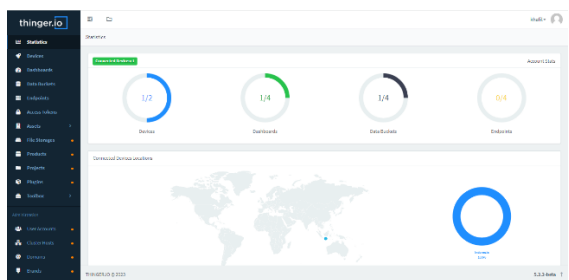
A. Pengujian Koneksi Antara Esp32 Dengan Thingier.Io

Pengujian ini dilakukan dengan melakukan konfigurasi antara perangkat dengan web thingier.io menggunakan ID address yang sudah diberikan secara random oleh thingier.io kemudian ID address tersebut akan di cantumkan ke dalam program pada ESP32 agar terhubung, seperti pada gambar 4.1 dan 4.2.



Gambar 4.1 Tampilan hasil komunikasi antara alat dan thingier.io

Pada gambar 4.1 menunjukkan status koneksi pada tampilan platform Thingier.io, status menunjukkan online yang berarti komunikasi antar ESP32 dengan web server Thingier.io telah berhasil terhubung. Pada gambar diatas juga terdapat transmited data dan received data, kemudian ip address dan status online ditunjukkan pada gambar berwarna hijau bertuliskan online serta terdapat grafik live transmission data pada bagian kanan. Pada bagian bawah adalah maps yang menampilkan dimana lokasi perangkat itu di aktifkan.



Gambar 4.2 Tampilan statistik komunikasi antara alat dan thingier.io

Pada gambar 4.2 juga menunjukkan bahwa ESP32 telah berhasil melakukan komunikasi dengan Thingier.io yang di tunjukkan pada statistik bahwa terdapat satu device yang terhubung.

B. Pengujian Pengiriman Data Dari ESP32 Ke Thingier.IO

Pengujian ini dilakukan untuk mengetahui kestabilan pengiriman data dari alat rancangan ke platform thingier.io. Kestabilan ini diuji dengan melihat waktu pengiriman dan tingkat error nilai data informasi ke thingier berdasarkan program yang telah dibuat, nilai yang didapat dari pengukuran sensor akan dikirim ke

database thingier setiap 1 detik, begitu juga pada tampilan display informasi user akan terupdate secara otomatis sesuai nilai yang dikirimkan dari sensor ke database thingier setiap 1 detik.

Namun, pada data bucket (data logger) pada thingier.IO dibuat untuk melakukan pencatatan data dari nilai sensor yang di terima sebanyak 60 sample data nilai per-1 menit. Ini dilakukan agar pada saat pengambilan data nilai sensor bisa jauh lebih akurat dengan menggunakan cara pengambilan 60 sample data nilai sensor per-60 detik, kemudian akan dirata-rata sehingga hasil pada pencatatan jauh lebih akurat. Kemudian data tersebut akan ditampilkan pada display dashboard seperti pada gambar 4.3, gambar 4.4 dan gambar 4.5.

Date	humidity	temperature
7/14/2023, 12:54:19 AM	73.69999694824219	28.399999618530273
7/14/2023, 12:53:18 AM	73.4000015258789	28.399999618530273
7/14/2023, 12:52:18 AM	73	28.5
7/14/2023, 12:51:17 AM	72.19999694824219	28.5
7/14/2023, 12:50:16 AM	70.19999694824219	28.600000381469727
7/14/2023, 12:49:15 AM	68.30000305175781	28.899999618530273
7/14/2023, 12:48:14 AM	65.5999984741211	29.5
7/14/2023, 12:47:13 AM	62.20000076293945	30.5
7/14/2023, 12:46:12 AM	61.900001525878906	32.20000076293945
7/14/2023, 12:45:12 AM	70.9000015258789	31.299999237060547
7/14/2023, 12:44:11 AM	75.19999694824219	28.100000381469727
7/14/2023, 12:43:11 AM	74.80000305175781	28.200000762939453
7/14/2023, 12:42:09 AM	75.30000305175781	28.200000762939453
7/14/2023, 12:41:08 AM	75	28.299999237060547
7/14/2023, 12:40:07 AM	75.5999984741211	28.299999237060547
7/14/2023, 12:39:06 AM	75.5999984741211	28.299999237060547

Gambar 4.3 Data Buckets Suhu dan Kelembapan via thingier.io

Date	Value
7/14/2023, 5:07:20 PM	61
7/14/2023, 5:06:19 PM	61
7/14/2023, 5:05:19 PM	74
7/14/2023, 5:04:19 PM	73
7/14/2023, 5:03:19 PM	95
7/14/2023, 5:02:18 PM	95
7/14/2023, 5:01:19 PM	95
7/14/2023, 5:00:18 PM	95
7/14/2023, 4:59:18 PM	94
7/14/2023, 4:58:18 PM	95

Gambar 4.4 Data buckets ketersediaan pakan via thingier.io

Date	Value
7/14/2023, 5:34:26 PM	61
7/14/2023, 5:33:26 PM	69
7/14/2023, 5:32:26 PM	70
7/14/2023, 5:31:25 PM	69
7/14/2023, 5:30:25 PM	69
7/14/2023, 5:29:25 PM	94
7/14/2023, 5:28:25 PM	94
7/14/2023, 5:27:26 PM	94
7/14/2023, 5:26:25 PM	94
7/14/2023, 5:25:24 PM	94

Gambar 4.5 Data Buckets Ketersediaan Minum Via Thinger.io

C. Pengujian Pengiriman Data Sensor DHT11

Pengujian dilakukan untuk mengetahui tingkat waktu delay dari sensor DHT11 sebagai pendeteksi suhu dan kelembapan ke thinger.io. Hasil pengujian bisa dilihat pada tabel 4.1.

Tabel 4.1 Tabel Pengujian Delay Pengiriman Data Sensor DHT11

No.	Time Stamp Sample dari Thinger.IO	Time Stamp Dari Serial monitor Arduino IDE
1.	12:39:06	12:39:06
2.	12:40:07	12:40:07
3.	12:41:08	12:41:08
4.	12:42:09	12:42:09
5.	12:43:11	12:43:11
6.	12:44:11	12:44:11
7.	12:45:12	12:45:12
8.	12:46:12	12:46:12
9.	12:47:13	12:47:13
10.	12:48:14	12:48:14
Rata-rata waktu delay		1.1 Detik

Pada tabel 4.1 dilakukan pengujian komunikasi pengiriman data sensor DHT11, dimana sample yang di bandingkan adalah sample time stamp yang tercatat pada thinger.io dengan sample time stamp yang tercatat pada serial monitor pada arduino IDE. Dari tabel 4.1 didapatkan waktu pengiriman setiap sampel dapat diambil rata-rata delay sebesar 1.1 detik.

D. Pengujian Pengiriman Data Sensor LoadCell

Pengujian dilakukan untuk mengetahui tingkat waktu delay dari sensor LoadCell untuk mendeteksi ketersediaan pakan yang dimisalkan dengan sebuah potensiometer untuk mengirimkan data ke Thinger.IO. Hasil pengujian bisa dilihat pada tabel 4.2.

Tabel 4.2 Tabel Pengujian Delay Pengiriman Data Sensor LoaCell

No.	Time Stamp Sample dari Thinger.IO	Time Stamp Dari Serial monitor Arduino IDE
1.	4:58:18	16:58:18
2.	4:59:18	16:59:18
3.	5:00:18	17:00:18
4.	5:01:19	17:01:19
5.	5:02:18	17:02:18
6.	5:03:19	17:03:19
7.	5:04:19	17:04:19
8.	5:05:19	17:05:19
9.	5:06:19	17:06:19
10.	5:07:20	17:07:20
Rata-rata waktu delay		0.4 Detik

Pada tabel 4.2 dilakukan pengujian komunikasi pengiriman data sensor loadcell yang dimisalkan menggunakan potensiometer, dimana sample yang di bandingkan adalah sample time stamp yang tercatat pada thinger.io dengan sample times tamp yang tercatat pada serial monitor pada Arduino IDE. Dari tabel 4.2 didapatkan bahwa waktu pengiriman setiap sample dapat diambil rata-rata delay sebesar 0.4 detik.

E. Pengujian Pengiriman Data Sensor Water Level

Pengujian dilakukan untuk mengetahui tingkat waktu delay dari sensor water level untuk mendeteksi ketersediaan minum yang dimisalkan dengan sebuah potensiometer untuk mengirimkan data ke thinger.io. Hasil pengujian bisa dilihat pada tabel 4.3.

Tabel 4.3 Tabel Pengujian Delay Pengiriman Data Water Sensor

No.	Time Stamp Sample dari Thinger.IO	Time Stamp Dari Serial monitor Arduino IDE
1.	5:25:24	17:25:24
2.	5:26:25	17:26:25
3.	5:27:26	17:27:26
4.	5:28:25	17:28:25
5.	5:29:25	17:29:25
6.	5:30:25	17:30:25
7.	5:31:25	17:31:25
8.	5:32:25	17:32:25
9.	5:33:26	17:33:26
10.	5:34:26	17:34:26
Rata-rata waktu delay		0.4 Detik

Pada tabel 4.3 dilakukan pengujian komunikasi pengiriman data sensor water sensor yang dimisalkan menggunakan potensiometer, dimana sample yang di bandingkan adalah sample time stamp yang tercatat pada thinger.io dengan sample time stamp yang tercatat pada serial monitor pada arduino IDE.

Dari tabel 4.3 didapatkan bahwa waktu pengiriman setiap sample dapat diambil rata-rata delay sebesar 0.4 detik.

F. Pengujian Kontrol Pada Kondisi Menyala (ON)

Pengujian dilakukan untuk mengetahui tingkat waktu delay dan tingkat nilai error data dari dashboard kontrol motor stepper, water pupm, mp3dfplayer mini yang dimisalkan dengan tiga buah led, untuk mengirimkan kondisi menyala (ON) ke led pada dashboard kontrol yang ada pada tampilan thinger.io sebagai kontrol led.

Tabel 4.4 Tabel Pengujian Fungsi Kontrol Kondisi Menyala (ON)

Pengujian	Kondisi Led	Waktu Delay (Detik)
1	ON	1.14
2	ON	1.48
3	ON	2.09
4	ON	1.12
5	ON	0.77
6	ON	0.86
7	ON	0.93
8	ON	0.83
9	ON	0.87
10	ON	0.91
11	ON	0.97
1	ON	1.59
13	ON	0.67
14	ON	1.06
15	ON	1.28
16	ON	0.85
17	ON	2.89
18	ON	0.99
19	ON	0.7
20	ON	1.15
21	ON	0.65
22	ON	0.61
23	ON	1.46
24	ON	0.75
25	ON	0.97
26	ON	1.36
27	ON	1.05
28	ON	1.16
29	ON	0.57
30	ON	0.92
Rata-Rata Waktu Delay		1.09

Pada tabel 4.4 dilakukan pengujian komunikasi dashboard kontrol yang dimisalkan menggunakan led, dimana pengujian dilakukan dengan menghitung waktu delay menggunakan stopwatch pada saat kondisi led dari kondisi mati (OFF) ke kondisi menyala (ON) setelah menekan tombol ON pada dashboard kontrol di thinger.io. Dari tabel 4.4 didapatkan bahwa dalam 30 kali percobaan led selalu berhasil menyala (ON) rata-rata waktu delay sebesar 1.09 detik.

G. Pengujian Kontrol Pada Kondisi Mati (OFF)

Pengujian dilakukan untuk mengetahui tingkat waktu delay dan tingkat nilai error data dari dashboard kontrol motor stepper, water pupm, mp3dfplayer mini yang dimisalkan dengan tiga buah led, untuk mengirimkan kondisi menyala (OFF) ke led pada dashboard kontrol yang ada pada tampilan thinger.io sebagai kontrol led.

Tabel 4.5 Tabel Pengujian Fungsi Kontrol Kondisi Mati (OFF)

Pengujian	Kondisi Led	Waktu Delay (Detik)
1	OFF	0.84
2	OFF	0.88
3	OFF	0.79
4	OFF	0.75
5	OFF	0.68
6	OFF	2.58
7	OFF	0.95
8	OFF	0.86
9	OFF	1.29
10	OFF	0.89
11	OFF	0.98
1	OFF	1.28
13	OFF	0.71
14	OFF	0.85
15	OFF	1.27
16	OFF	1.03
17	OFF	0.4
18	OFF	0.87
19	OFF	1.47
20	OFF	1.06
21	OFF	1.15
22	OFF	0.76
23	OFF	0.79
24	OFF	0.9
25	OFF	0.52
26	OFF	0.39
27	OFF	0.84
28	OFF	1.17
29	OFF	1
30	OFF	1.34
Rata-Rata Waktu Delay		0.98

Pada tabel 4.5 dilakukan pengujian komunikasi dashboard kontrol yang dimisalkan menggunakan led, dimana pengujian dilakukan dengan menghitung waktu delay menggunakan stopwatch pada saat kondisi led dari menyala (ON) ke kondisi mati (OFF) setelah menekan tombol OFF pada dashboard kontrol di thinger.io. Dari tabel 4.5 didapatkan bahwa dalam 30 kali percobaan led selalu berhasil mati (OFF) dengan rata-rata delay sebesar 0.98 detik.

H. Hasil Tampilan Monitoring



Gambar 4.6 Tampilan Umum Monitoring Dan Kontrol Sangkar Burung Murai Batu



Gambar 4.7 Tampilan Monitoring Dan Kontrol Sangkar Burung Murai Batu Saat Prototype Sudah Berjalan

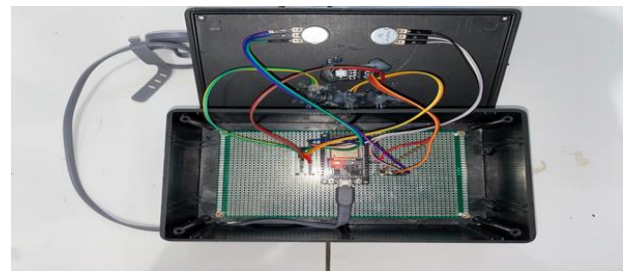
I. Pengujian Durabilitas Sistem Monitoring Dan Kontrol

Pengujian ini dilakukan untuk mengetahui apakah sistem monitoring dan kontrol sangkar burung murai batu via thinger.io yang telah dibuat berjalan dengan baik atau tidak. Pada pengujian ini dilakukan uji coba menjalankan sistem dengan durasi 1 jam, kemudian setelah 1 jam sistem di tinjau kembali untuk dilihat apakah sistem tetap berjalan dengan baik atau tidak. Hasil yang didapatkan adalah sistem tetap berjalan dengan baik, tampilan monitoring tidak terjadi error dan dashboard kontrol juga tetap berjalan dengan baik tidak ada error setelah di uji coba untuk terus dijalankan selama 1 jam.

J. Hasil Rancangan Prototype



Gambar 4.8 Hasil Rancangan Prototype



Gambar 4.9 Rangkaian pada prototype

V. PENUTUP

A. Kesimpulan

Berdasarkan dari hasil dan pembahasan yang telah dilakukan, dalam penelitian yang berjudul “Sistem Monitoring Dan Kontrol sangkar Burung Murai Batu” ini, didapatkan kesimpulan penelitian sebagai berikut :

1. Hasil pengujian pengiriman data sensor DHT11 memiliki rata-rata waktu delay 1.1 detik.
2. Hasil pengujian pengiriman data sensor Loadcell memiliki rata-rata waktu delay 0.4 detik.
3. Hasil pengujian pengiriman data sensor Water Level memiliki rata-rata waktu delay 0.4 detik.
4. Hasil pengujian Kontrol pada kondisi ON memiliki rata-rata waktu delay 1.9 detik.
5. Hasil pengujian Kontrol pada kondisi OFF memiliki rata-rata waktu delay 0.98 detik.
6. Seluruh pengujian waktu delay didapatkan rata-rata waktu delay sebesar 1 detik atau dapat dikatakan *real time*, karena menurut dari beberapa sumber dikatakan bahwa pada skala industri *real time* dapat di tentukan jika waktu respon antara kondisi sebenarnya dan waktu display sama dengan waktu yang diharapkan.[22]
7. Hasil pengujian pengiriman data dari ESP32 ke Thinger.IO yaitu data dapat ter-Update setiap 1 detik namun untuk pencatatan data pada data buckets via thinger.io diambil dalam 1 menit agar data yang diambil tetap akurat dan dapat diambil kesimpulan bahwa komunikasi antara ESP32 dan Thinger.IO berhasil.
8. Hasil pengujian pada tampilan monitoring berhasil menampilkan kondisi sangkar burung berupa kondisi suhu dan kelembapan, kondisi ketersediaan pakan, serta kondisi ketersediaan minum yang dapat diambil kesimpulan bahwa tampilan monitoring yang dibuat telah berhasil dan berjalan dengan baik.
9. Hasil pengujian pada dashboard kontrol berhasil melakukan kontrol terhadap kondisi buka atau tutup atap dengan kontrol ON/OFF pada motor stepper, kondisi mandi burung dengan kontrol ON/OFF water pump, kondisi latihan suara burung dengan kontrol ON/OFF

- Mp3dfPlayer dan dapat diambil kesimpulan bahwa fungsi kontrol berjalan dengan baik.
10. Hasil pada pengujian durabilitas sistem monitoring dan kontrol sangkar burung murai batu berhasil berjalan sesuai rancangan yang dibuat.
 11. Sistem monitoring dan kontrol sangkar burung murai batu dapat dimonitor dan dapat melakukan kontrol melalui jarak jauh dengan menggunakan akses internet.

B. Saran

Berdasarkan dengan hasil pengujian sistem monitoring dan kontrol sangkar burung murai batu ada saran untuk pengembangan di masa mendatang, yaitu Penambahan fitur kontrol untuk kondisi pakan dan kondisi minum.

DAFTAR PUSTAKA

- [1] G. Dananjoyo, F. Hermawan, dan B. Sungkwo, "Analisis Kelayakan Keuangan Bisnis Budidaya Burung Murai Batu Narogong," *Manajemen*, vol. 12, no. 2, hal. 190–198, 2020, [Daring]. Tersedia pada: <http://journal.feb.unmul.ac.id/index.php/JURNALMANAJEMEN>
- [2] J. Ilmiah, P. Terpadu, P. Ayam, P. Pada, S. Brooding, dan K. Dan, "Khaira Nova," *J. Ilm. Peternak. Terpadu*, vol. 4, no. 3, hal. 222–229, 2016.
- [3] A. Khumaidi, "Mikrokontroler Arduino," *J. Teknokris*, vol. 20, hal. 9, 2017.
- [4] Saverus, "No 主観的健康感を中心とした在宅高齢者における健康関連指標に関する共分散構造分析Title," *J. Kaji. Pendidik. Ekon. dan Ilmu Ekon.*, vol. 2, no. 1, hal. 1–19, 2019, [Daring]. Tersedia pada: http://www.scopus.com/inward/record.url?eid=2-s2.0-84865607390&partnerID=tZOtx3y1%0Ahttp://books.google.com/books?hl=en&lr=&id=2LIMMD9FVXkC&oi=fnd&pg=PR5&dq=Principles+of+Digital+Image+Processing+fundamental+techniques&ots=HjrHeuS_
- [5] A. F. S. R. dan A. D. P. P. Mayda Waruni Kasrani, "Otomatisasi Tempat Makan Dan Minum Burung Berbasis Mikrokontroler Board Arduino Dan Gsm 900," *Pros. Semin. Din. Inform. 2018 (Senadi 2018)*, vol. 2018, no. Senadi, hal. 27–32, 2018.
- [6] F. Muchtar, S. Adi Wibowo, dan A. Ariwibisono, "PENERAPAN IoT (Internet of Thing) TERHADAP RANCANG BANGUN SANGKAR BURUNG PINTAR UNTUK BURUNG TERIEP," *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 5, no. 1, hal. 162–170, 2021, doi: 10.36040/jati.v5i1.3219.
- [7] D. Darlis dan T. Haryanti, "RANCANG BANGUN SMART BIRD CAGE DENGAN PENGENDALIAN BERBASIS IoT Design And Build Smart Bird Cage With Control Based On IoT," vol. 7, no. 6, hal. 3308–3320, 2021.
- [8] I. Z. Zamani, R. H. Hardyanto, P. W. Ciptadi, dan R. Nadilah, "Pengisian Makanan Dan Minuman Burung Secara Otomatis Menggunakan Mikrokontroler Dan Android," *Appl. Sci. Technol. Reaserch J.*, vol. 1, no. 1, hal. 1–10, 2022, doi: 10.31316/astro.v1i1.3206.
- [9] S. Adi Wibowo, D. R. Rudhistiar, dan K. Ardi Widodo, "Implementasi Low Cost Smart Farming Dalam Penangkaran Burung Kicau," *J. Mnemon.*, vol. 5, no. 1, hal. 72–79, 2022, doi: 10.36040/mnemonic.v5i1.4435.
- [10] R. A. Taruna, "Strategi Pengembangan ABPN Asosiasi Penangkar Burung Nusantara) Usaha Peternakan Burung Murai Batu (*Copsychus malabaricus*) di Korwil Sumatera Utara," *J. Ilm. Mhs. Pertan. ...*, vol. 2, hal. 1–10, 2022, [Daring]. Tersedia pada: <http://jurnalmahasiswa.umsu.ac.id/index.php/jimtani/article/view/1381%0Ahttp://jurnalmahasiswa.umsu.ac.id/index.php/jimtani/article/download/1381/1418>
- [11] S. A. Hurrijal dan R. Gupitha, "Sistem Informasi Monitoring Sales Berbasis Web Pada PT. Arifindo Mandiri TDC Pamanukan," *J. FIKI (Jurnal Teknol. Inf. dan Komunikasi)*, vol. X, no. 2, hal. 63–67, 2020.
- [12] D. A. Megawaty, M. Bakri, dan E. Damayanti, "SISTEM MONITORING KEGIATAN AKADEMIK SISWA," vol. 14, no. 2, hal. 98–101, 2020.
- [13] F. Adani dan S. Salsabil, "Internet of Things: Sejarah Teknologi Dan Penerapannya," *Isu Teknol. Stt Mandala*, vol. 14, no. 2, hal. 92–99, 2019.
- [14] B. Artono dan R. G. Putra, "PENERAPAN INTERNET OF THINGS (IOT) UNTUK KONTROL LAMPU MENGGUNAKAN ARDUINO BERBASIS WEB," vol. 05, no. 01, hal. 9–16, 2018.
- [15] W. K. Raharja dan R. Ramadhon, "Purwarupa Alat Pendeteksi Kebakaran Jarak Jauh Menggunakan Platform Thinger.Io," *J. Elektro Luceat*, vol. 7, no. 2, hal. 188–206, 2021.
- [16] S. Dharma, "Analisis Kinerja Jaringan WIFI P- ISSN : 2302-3295," vol. 8, no. 2, 2020.
- [17] D. Srivastava, A. Kesarwani, dan S. Dubey, "Measurement of Temperature and Humidity by using Arduino Tool and DHT11," *Int. Res. J. Eng. Technol.*, vol. 5, no. 12, hal. 876–878, 2018, [Daring]. Tersedia pada: www.irjet.net
- [18] A. P. Zanofa, R. Arrahman, M. Bakri, dan A. Budiman, "Pintu Gerbang Otomatis Berbasis Mikrokontroler Arduino Uno R3," *J. Tek. dan Sist. Komput.*, vol. 1, no. 1, hal. 22–27, 2020, doi: 10.33365/jtikom.v1i1.76.
- [19] A. Khamdilah, Ali; Saleh, "Analisa kegagalan monitoring dan pengontrolan cpp (controllable pitch propeller) dalam sistem maneuvering berpengaruh pada pergerakan kapal 1&2," *J. Ilm. Gema Marit.*, hal. 116–119, 2020.



Penulis lahir di kabupaten Kutai Barat, Kalimantan Timur tanggal 24 Maret 2001, dan mulai bersekolah di SDN 004 Melak Ulu pada tahun 2007 dan lulus pada tahun 2013, kemudian lanjut SLTP di SMPN 1 Melak Ulu pada tahun 2013 dan lulus pada tahun 2016, kemudian melanjutkan pendidikan SLTA di SMAN 1 SENDAWAR pada tahun 2016 dengan memilih kejuruan MIPA hingga lulus pada tahun 2019. Selanjutnya melanjutkan pendidikan di perguruan tinggi Institut Teknologi Nasional Malang dengan program studi Teknik Elektro S-1 konsentrasi Elektronika.