

**BUKU TUGAS AKHIR
CAPSTONE DESIGN**



**OPTIMASI PENGERINGAN MAGGOT DENGAN
TEKNOLOGI IOT**

Oleh :

Daffa Naufal Putra / 1101200079

Muhammad Rinaldy Fathurrahman / 1101200290

Fadhillah Padmanaba Fauzi / 1101204057

Raymond Ibrahim / 1101204361

**PRODI S1 TEKNIK TELEKOMUNIKASI
FAKULTAS TEKNIK ELEKTRO
UNIVERSITAS TELKOM
BANDUNG
2024**

LEMBAR PENGESAHAN
BUKU CAPSTONE DESIGN

OPTIMASI PENGERINGAN MAGGOT DENGAN TEKNOLOGI IOT

OPTIMIZATION OF MAGGOT DRYING WITH IOT TECHNOLOGY

Telah disetujui dan disahkan sebagai bagian dari Capstone Design

Program S1 Teknik Telekomunikasi

Fakultas Teknik Elektro

Universitas Telkom

Bandung

Disusun oleh:

Daffa Naufal Putra / 1101200079

Muhammad Rinaldy Fathurrahman / 1101200290

Fadhillah Padmanaba Fauzi / 1101204057

Raymond Ibrahim / 1101204361

Bandung, 15 Juli 2024

Menyetujui,

Pembimbing 1

Pembimbing 2

Dr. Sofia Naning Hertiana, Ir.,M.T.
NIP. 99710011

Dr. Iman Hedi Santoso, S.T., M.T.
NIP. 08740008

LEMBAR PERNYATAAN ORISINALITAS

Saya, yang bertanda tangan di bawah ini

Nama : Daffa Naufal Putra

NIM : 1101200079

Alamat : Mutiara Sentul Blok I No. 14 RT.003/010, Kabupaten Bogor

No. Telepon : 081383272674

Email : daffanaufal142@gmail.com

Menyatakan bahwa Buku Capstone Design ini merupakan karya orisinal saya sendiri bersama dengan kelompok Capstone Design saya, dengan judul:

OPTIMASI PENGERINGAN MAGGOT DENGAN TEKNOLOGI IOT

OPTIMIZATION OF MAGGOT DRYING WITH IOT TECHNOLOGY

Atas pernyataan ini, saya siap menanggung resiko/sanksi yang dijatuhkan kepada saya apabila dikemudian ditemukan adanya pelanggaran terhadap kejujuran akademik atau etika keilmuan dalam karya ini, atau ditemukan bukti yang menunjukkan ketidak aslian karya ini.

Bandung, 15 Juli 2024



Daffa Naufal Putra
1101200079

LEMBAR PERNYATAAN ORISINALITAS

Saya, yang bertanda tangan di bawah ini

Nama : Fadhillah Padmanaba Fauzi
NIM : 1101204057
Alamat : Losari No. 14 RT.003/005, Ampelgading, Kabupaten Pemalang
No. Telepon : 083865810571
Email : padmanaba1710@gmail.com

Menyatakan bahwa Buku Capstone Design ini merupakan karya orisinal saya sendiri bersama dengan kelompok Capstone Design saya, dengan judul:

OPTIMASI PENGERINGAN MAGGOT DENGAN TEKNOLOGI IOT

OPTIMIZATION OF MAGGOT DRYING WITH IOT TECHNOLOGY

Atas pernyataan ini, saya siap menanggung resiko/sanksi yang dijatuhkan kepada saya apabila dikemudian ditemukan adanya pelanggaran terhadap kejujuran akademik atau etika keilmuan dalam karya ini, atau ditemukan bukti yang menunjukkan ketidak aslian karya ini.

Bandung, 15 Juli 2024



Fadhillah Padmanaba Fauzi
1101204057

LEMBAR PERNYATAAN ORISINALITAS

Saya, yang bertanda tangan di bawah ini

Nama : Muhammad Rinaldy Fathurrahman
NIM : 1101200290
Alamat : Jl. Cipinang Lontar III no. 27 RT 08/09, Pulogadung, Jakarta Timur
No. Telepon : 081370601317
Email : aldy.muhammad311@gmail.com

Menyatakan bahwa Buku Capstone Design ini merupakan karya orisinal saya sendiri bersama dengan kelompok Capstone Design saya, dengan judul:

OPTIMASI PENGERINGAN MAGGOT DENGAN TEKNOLOGI IOT

OPTIMIZATION OF MAGGOT DRYING WITH IOT TECHNOLOGY

Atas pernyataan ini, saya siap menanggung resiko/sanksi yang dijatuhkan kepada saya apabila dikemudian ditemukan adanya pelanggaran terhadap kejujuran akademik atau etika keilmuan dalam karya ini, atau ditemukan bukti yang menunjukkan ketidak aslian karya ini.

Bandung, 15 Juli 2024



Muhammad Rinaldy Fathurrahman
1101200290

LEMBAR PERNYATAAN ORISINALITAS

Saya, yang bertanda tangan di bawah ini

Nama : Raymond Ibrahim
NIM : 1101204361
Alamat : Jl Suryani Dalam 1 no 202
No. Telepon : 0895361327154
Email : raymondibrahim12@gmail.com

Menyatakan bahwa Buku Capstone Design ini merupakan karya orisinal saya sendiri bersama dengan kelompok Capstone Design saya, dengan judul:

OPTIMASI PENGERINGAN MAGGOT DENGAN TEKNOLOGI IOT

OPTIMIZATION OF MAGGOT DRYING WITH IOT TECHNOLOGY

Atas pernyataan ini, saya siap menanggung resiko/sanksi yang dijatuhkan kepada saya apabila dikemudian ditemukan adanya pelanggaran terhadap kejujuran akademik atau etika keilmuan dalam karya ini, atau ditemukan bukti yang menunjukkan ketidak aslian karya ini.

Bandung, 15 Juli 2024



Raymond Ibrahim
1101204361

ABSTRAK

Budidaya maggot semakin banyak diminati karena banyak peternak yang membutuhkan maggot sebagai pakan untuk hewan ternak mereka. Maggot merupakan sumber pakan yang sangat berguna untuk ikan dan unggas. Maggot dapat diberikan dalam keadaan hidup (basah) atau diolah menjadi bentuk kering. Maggot kering memiliki daya tahan simpan yang lebih lama dibandingkan dengan maggot basah. Maggot kering dihasilkan melalui proses pengeringan yang dapat secara signifikan memperpanjang umur simpan maggot, menjaga konsistensi kualitasnya, dan mengurangi risiko pembusukan. Namun saat ini mesin pengering maggot yang tersedia masih konvensional, sehingga memerlukan kehadiran dan pengawasan langsung oleh pekerja. Hal ini mengakibatkan waktu yang digunakan menjadi kurang efektif, karena proses pengeringan harus diawasi secara konstan dan pekerja harus menunggu hingga maggot benar-benar kering sebelum dapat melanjutkan ke tahap berikutnya.

Solusi yang kami tawarkan adalah dengan mengaplikasikan *Internet of Things* dalam *monitoring* proses pengeringan untuk menjaga kualitas proses pengeringan. Pengaplikasian *Internet of Things* dilakukan dengan menggunakan mikrokontroler yang berfungsi untuk mengontrol alat pengering maggot. Setelah itu membuat sistem *monitoring* berupa aplikasi *monitoring* alat pengering maggot yang memiliki kemampuan pemantauan jarak jauh yang memberikan fleksibilitas. Aplikasi yang kami buat memiliki fitur untuk memasukkan berapa berat maggot yang akan dikeringkan dalam satuan kilogram, selama proses pengeringan berjalan, aplikasi *monitoring* akan menampilkan estimasi waktu yang diperlukan untuk mengeringkan maggot dan akan memberikan notifikasi bagi peternak maggot apabila proses pengeringan telah selesai.

Alat pengering maggot yang kami buat dapat bekerja dengan optimal dan menghasilkan maggot kering dengan kualitas yang baik dan konsisten. Sensor dan komponen yang terdapat pada alat pengering maggot juga memiliki akurasi dan fungsi yang memuaskan, begitu juga dengan aplikasi yang mampu mengontrol dan *monitoring* alat pengering maggot. Pembuatan alat pengering maggot dengan teknologi *Internet of Things* ini diharapkan dapat memberikan solusi efektif dan bermanfaat bagi peternak maggot, khususnya dalam proses produksi maggot kering. Peternak dapat mengelola pengeringan maggot dengan lebih efisien, meningkatkan produktivitas, dan menjaga kualitas maggot kering yang dihasilkan, sehingga memberikan nilai tambah yang signifikan bagi usaha mereka.

Kata kunci : *Internet of Things*, Maggot, Pengeringan, *Monitoring*.

ABSTRACT

Maggot cultivation is increasingly in demand as many farmers need maggots as feed for their livestock. Maggots are a very useful source of feed for fish and poultry. Maggots can be given in a live (wet) state or processed into a dry form. Dry maggots have a longer shelf life compared to wet maggots. Dry maggots are produced through a drying process that can significantly extend their shelf life, maintain quality consistency, and reduce the risk of spoilage. However, the current maggot drying machines are still conventional, requiring direct presence and supervision by workers. This results in less effective time usage because the drying process must be constantly monitored, and workers have to wait until the maggots are completely dry before proceeding to the next stage.

The solution we offer is to apply the Internet of Things (IoT) in monitoring the drying process to maintain the quality of the drying process. The application of the Internet of Things is done by using a microcontroller that functions to control the maggot drying machine. After that, we create a monitoring system in the form of a maggot dryer monitoring application that has remote monitoring capabilities, providing flexibility. The application we created features the ability to input the weight of maggots to be dried in kilograms. During the drying process, the monitoring application will display an estimated time required to dry the maggots and will notify the maggot farmers when the drying process is complete.

The maggot dryer we created can work optimally and produce dry maggots with good and consistent quality. The sensors and components in the maggot dryer also have satisfactory accuracy and function, as does the application that can control and monitor the maggot dryer. The creation of this maggot dryer with Internet of Things technology is expected to provide effective and beneficial solutions for maggot farmers, especially in the production process of dry maggots. Farmers can manage maggot drying more efficiently, increase productivity, and maintain the quality of the dried maggots produced, thus providing significant added value to their business.

Keywords: Internet of Things, Maggot, Dryer, Monitoring.

KATA PENGANTAR

Puji syukur kami panjatkan ke hadirat Tuhan Yang Maha Esa atas segala rahmat-Nya. Shalawat serta salam senantiasa tercurah kepada Nabi Muhammad SAW, yang telah membawa umat manusia dari kegelapan menuju cahaya ilmu pengetahuan.

Dengan penuh rasa syukur, kami sebagai mahasiswa Program Studi Teknik Telekomunikasi merasa terhormat dapat menyelesaikan tugas akhir *capstone design* berjudul "Optimasi Pengerinan Maggot dengan Teknologi IoT". Penelitian ini merupakan salah satu wujud dari perjalanan akademik kami di Universitas Telkom, yang tidak terlepas dari bimbingan, dorongan, serta dukungan dari berbagai pihak.

Tugas akhir ini bertujuan untuk menjawab tantangan dalam pengerinan maggot supaya menjadi lebih efisien dengan menggunakan teknologi *Internet of Things* (IoT). Kami berharap hasil penelitian ini dapat memberikan kontribusi nyata dalam bidang pengolahan maggot menjadi produk yang memiliki nilai jual lebih dengan diolah menjadi maggot kering.

Kami mengucapkan terima kasih yang tak terhingga kepada semua pihak yang telah memberikan dukungan, bimbingan, serta motivasi bagi kami dalam menyelesaikan tugas akhir ini. Semoga tugas akhir *capstone design* yang kami buat dapat bermanfaat bagi perkembangan ilmu pengetahuan dan teknologi di masa depan.

Akhir kata, kami mohon maaf apabila terdapat kekurangan dalam penyusunan laporan ini. Kritik dan saran yang membangun sangat kami harapkan untuk perbaikan di masa mendatang.

UCAPAN TERIMA KASIH

Kepada Allah SWT yang Maha Kuasa, penulis memanjatkan puji syukur atas segala segala rahmat dan hidayah-Nya yang telah memberikan petunjuk dan kesehatan sehingga penulis dapat menyelesaikan penelitian ini dengan judul “Optimasi Pengeringan Maggot dengan Teknologi IoT”. Capstone design ini dibuat untuk syarat menyelesaikan pendidikan pada Program Studi S1 Teknik Telekomunikasi Universitas Telkom.

Penulis mengucapkan terima kasih sebanyak-banyaknya kepada:

1. Keluarga tersayang yang selalu memberikan dukungan dan doa dalam setiap proses perjalanan penulisan capstone design.
2. Dosen Pembimbing Capstone Design, yaitu Ibu Dr. Sofia Naning Hertiana, Ir., M.T. dan Bapak Dr. Iman Hedi Santoso, S.T., M.T., yang telah memberikan bimbingan dan masukkan yang membantu dalam proses penyusunan *capstone design* ini.
3. Teman seperjuangan kelompok penyusunan capstone design, yaitu Daffa Naufal Putra, Muhammad Rinaldy Fathurrahman, Fadhillah Padmanaba Fauzi, Raymond Ibrahim, telah saling memberikan dukungan dan doa dalam setiap proses penyusunan *capstone design* ini.
4. Bapak Wali Dosen kelas TT-44-01, yaitu Bapak Dr. Bambang Setia Nugroho, S.T., M.T., yang telah menjadi orang tua di masa perkuliahan.
5. Pada Dosen yang telah memberikan ilmu dan pengalaman yang mendukung penelitian pada *capstone design* ini.
6. Semua teman penulis yang selalu memberikan dukungan dan doa dalam setiap langkah perjalanan dari awal perkuliahan hingga saat ini.

Kepada semua pihak yang disebutkan, terima kasih atas kontribusi dan kerja sama yang diberikan dalam penyelesaian capstone design ini. Capstone design ini berhasil karena adanya kontribusi dari semua pihak yang telah disebutkan. Semoga capstone design ini memberikan banyak manfaat terhadap pengembangan ilmu pengetahuan di bidangnya. Penulis memohon maaf atas segala kekurangan dan kesalahan dalam capstone design ini.

Bandung, 15 Juli 2024

Penulis

DAFTAR ISI

LEMBAR PENGESAHAN	i
BUKU CAPSTONE DESIGN	i
LEMBAR PERNYATAAN ORISINALITAS	ii
ABSTRAK.....	vi
ABSTRACT.....	vii
KATA PENGANTAR	viii
UCAPAN TERIMA KASIH	ix
DAFTAR ISI.....	x
DAFTAR GAMBAR	xiii
DAFTAR TABEL.....	xv
BAB 1 USULAN GAGASAN	1
1.1 Deskripsi Umum Masalah.....	1
1.1.1 Latar Belakang Masalah	1
1.1.2 Analisa Masalah.....	2
1.1.3 Tujuan Capstone	2
1.2 Analisa Solusi yang Ada.....	3
1.2.1 Rancang Bangun Sistem Pengering Maggot BSF Sebagai Alternatif Pakan Ternak Berbasis Iot (<i>Internet of Things</i>).....	3
1.2.2 Proteksi Motor Induksi Satu Fasa Terhadap Kenaikan Suhu Pada Pengering Maggot Berbasis Panel Surya	3
BAB 2 SPESIFIKASI DAN BATASAN SOLUSI	4
2.1 Dasar Penentuan Spesifikasi	4
2.2 Batasan dan Spesifikasi.....	4
2.3 Pengukuran/Verifikasi Spesifikasi.....	5
2.3.1 Verifikasi spesifikasi Sensor Suhu.....	5

2.3.2	Verifikasi spesifikasi Motor Servo	6
2.3.3	Verifikasi Spesifikasi Pemantik Api dan <i>Valve solenoid</i>	6
2.3.4	Verifikasi Spesifikasi Aplikasi	7
BAB 3 DESAIN RANCANGAN SOLUSI		8
3.1	Alternatif Usulan Solusi.....	8
3.1.1	Alternatif Solusi 1	8
3.1.2	Alternatif Solusi 2	8
3.1.3	Alternatif Solusi 3	9
3.2	Analisis dan Pemilihan Solusi	9
3.3	Desain Solusi Terpilih.....	10
3.3.1	Diagram Alur Sistem	10
3.3.2	<i>Flowchart</i> Proses Pengeringan Maggot.....	12
3.3.3	<i>Flowchart</i> Aplikasi	13
3.3.4	UI Desain Aplikasi.....	14
3.4	Jadwal dan Anggaran.....	15
BAB 4 IMPLEMENTASI		17
4.1	Deskripsi Umum Implementasi	17
4.2	Detil Implementasi.....	18
4.2.1	Software	18
4.2.2	<i>Hardware</i>	26
4.3	Prosedur Pengoperasian	28
BAB 5 PENGUJIAN DAN KESIMPULAN.....		33
5.1	Skenario Umum Pengujian	33
5.1.1	Tujuan Pengujian	33
5.1.2	Daftar Pengujian	34
5.1.3	Lokasi dan Waktu Pengujian	34
5.1.4	Pihak yang Terlibat	34

5.2	Detil Pengujian.....	34
5.2.1	Pengujian Keandalan Koneksi	34
5.2.2	Pengujian Sensor Suhu	36
5.2.3	Pengujian Motor Servo	39
5.2.4	Pengujian Pemantik Otomatis.....	42
5.2.5	Pengujian <i>Valve Solenoid</i>	45
5.2.6	Pengujian <i>Relay 5V DC</i>	46
5.2.7	Pengujian Usabilitas Aplikasi	48
5.2.8	Pengujian Hasil Maggot Kering.....	50
5.3	Analisis Hasil Pengujian	53
5.3.1	Analisa Hasil Pengujian Keandalan Koneksi	53
5.3.2	Analisa Hasil Pengujian Sensor Suhu.....	53
5.3.3	Analisa Hasil Pengujian Motor Servo.....	54
5.3.4	Analisa Hasil Pengujian Pemantik Otomatis dan <i>Valve solenoid</i>	54
5.3.5	Analisa Hasil Pengujian <i>Relay 5V</i>	54
5.3.6	Analisa Hasil Pengujian Usabilitas Aplikasi	55
5.3.7	Analisa Pengujian Hasil Maggot Kering	55
5.4	Kesimpulan	56
	DAFTAR PUSTAKA	57
	LAMPIRAN I	59
	LAMPIRAN II	69
	LAMPIRAN III.....	70
	LAMPIRAN IV	80

DAFTAR GAMBAR

Gambar 3. 1 Diagram Alur Sistem	10
Gambar 3. 2 <i>Flowchart</i> Proses Pengeringan Maggot	12
Gambar 3. 3 <i>Flowchart</i> Aplikasi.....	13
Gambar 3. 4 Tampilan Awal Aplikasi	14
Gambar 3. 5 Tampilan Aplikasi Saat Proses Pengeringan Berlangsung	14
Gambar 3. 6 Tampilan Aplikasi Saat Proses Pengeringan Selesai	15
Gambar 4. 1 Realtime <i>Database</i> Firebase.....	25
Gambar 4. 2 Desain Perangkat Keras	26
Gambar 4. 3 Alat Pengering Maggot	26
Gambar 4. 4 ESP8266.....	27
Gambar 4. 5 MAX6675 <i>Thermocouple</i> Sensor	27
Gambar 4. 6 <i>Valve Solenoid</i>	28
Gambar 4. 7 Pemantik Otomatis.....	28
Gambar 4. 8 Motor Servo	28
Gambar 4. 9 Komponen Terhubung ke Power Source	29
Gambar 4. 10 ESP8266 dan Aplikasi Monitoring Terkoneksi Internet	29
Gambar 4. 11 Gas Terpasang dan Saklar Baterai Hidup.....	29
Gambar 4. 12 Tampilan Utama Aplikasi	30
Gambar 4. 13 Tampilan Saat Proses Pengeringan Berlangsung	30
Gambar 4. 14 Pengeringan Maggot Berlangsung	31
Gambar 4. 15 Tampilan Saat Proses Pengeringan Selesai	31
Gambar 4. 16 Proses Pengeringan Selesai.....	32
Gambar 5. 1 Alat Pengering Maggot	33
Gambar 5. 2 Topologi Pengujian Koneksi	34
Gambar 5. 3 Capture Data Delay Wireshark	35
Gambar 5. 4 Capture Data Throughput Wireshark	36
Gambar 5. 5 Thermometer Digital.....	37
Gambar 5. 6 Motor Servo	39
Gambar 5. 7 Tachometer	40
Gambar 5. 8 Chart Kestabilan Motor Servo.....	42
Gambar 5. 9 Pemantik Otomatis.....	42
Gambar 5. 10 Valve Solenoid	45

Gambar 5. 11 Relay 5V DC	47
Gambar 5. 12 Menu Awal	48
Gambar 5. 13 Prose Pengeringan	49
Gambar 5. 14 Pengeringan Selesai	49
Gambar 5. 15 Sesi Wawancara	50
Gambar 5. 16 Maggot Gosong	50
Gambar 5. 17 Maggot Optimal	51
Gambar 5. 18 Maggot Belum Matang	51
Gambar 5. 19 Penyusutan Berat Maggot	52
Gambar 5. 20 <i>Survey</i> Kualitas Maggot	53

DAFTAR TABEL

Tabel 2. 1 Tabel Verifikasi Spesifikasi Sensor Suhu	5
Tabel 2. 2 Verifikasi Spesifikasi Motor Servo	6
Tabel 2. 3 Verifikasi Spesifikasi Pemantik Api dan <i>Valve solenoid</i>	6
Tabel 2. 4 Verifikasi Spesifikasi Aplikasi	7
Tabel 2. 5 Perbandingan Antar Solusi	10
Tabel 3. 1 Jadwal Kegiatan Selama 6 Bulan	16
Tabel 3. 2 Anggaran Dana	16
Tabel 5. 1 Pengujian Sensor Suhu	37
Tabel 5. 2 Pengujian Kestabilan Motor Servo	40
Tabel 5. 3 Pengujian Pemantik Otomatis	43
Tabel 5. 4 Pengujian <i>Valve solenoid</i>	46
Tabel 5. 5 Pengujian <i>Relay 5V DC</i>	47
Tabel 5. 6 Pengujian Hasil Maggot Kering	52

BAB 1

USULAN GAGASAN

1.1 Deskripsi Umum Masalah

1.1.1 Latar Belakang Masalah

Budidaya maggot semakin banyak diminati karena banyak peternak yang membutuhkan maggot sebagai pakan untuk hewan ternak mereka. Selama siklus hidupnya, lalat *Black Soldier Fly* (BSF) menghasilkan telur yang kemudian dapat bertransformasi menjadi maggot yang merupakan sumber pakan yang sangat berguna untuk ikan dan unggas. Maggot dapat diberikan dalam keadaan hidup (basah) atau diolah menjadi bentuk kering dan pelet. Maggot mampu menjadi sumber protein pakan yang efektif dalam bentuk kering, sehingga membantu mengurangi biaya tinggi yang diperlukan untuk protein pakan bagi para peternak ikan dan unggas.

Para pembudidaya maggot selama ini mengalami kesulitan dalam mempertahankan kualitas maggot yang mereka panen karena teknik pasca panen maggot belum optimal. Maggot yang telah mati biasanya masih memiliki tingkat kelembaban yang tinggi, sehingga tidak tahan lama untuk disimpan dan berisiko mengalami penurunan kualitas yang signifikan. Oleh sebab itu, untuk mempertahankan produk menjadi tahan lama dan menjaga kualitasnya tetap terjaga, satu solusi yang efektif adalah dengan menghilangkan kadar air di dalam maggot melalui proses pengeringan. Proses pengeringan ini dapat secara signifikan memperpanjang umur simpan maggot, menjaga konsistensi kualitasnya, dan mengurangi risiko pembusukan atau kerusakan akibat kelembaban berlebihan. Cara mengeringkan maggot melibatkan tahap-tahap tertentu, terutama dengan penggunaan alat pengering seperti sangrai berputar. Namun, saat ini, mesin sangrai yang tersedia masih umumnya bersifat manual, sehingga memerlukan kehadiran dan pengawasan langsung oleh pekerja. Hal ini mengakibatkan waktu yang digunakan menjadi kurang efektif, karena proses pengeringan harus diawasi secara konstan dan pekerja harus menunggu hingga maggot benar-benar kering sebelum dapat melanjutkan ke tahap berikutnya.

Pengaplikasian *Internet of Things* dalam *monitoring* pengeringan maggot dapat diaplikasikan untuk menjaga kualitas proses pengeringan. Pengaplikasian *Internet of Things* dapat dilakukan dengan cara menggunakan sensor suhu sebagai parameter dalam alat pengering maggot dan menggunakan mikrokontroler yang berfungsi mengontrol semua komponen yang ada pada alat pengering melalui *realtime database*. Setelah itu membuat sistem

monitoring berupa aplikasi *monitoring* alat pengering maggot yang menggunakan *realtime database* sebagai referensinya dan memiliki kemampuan pemantauan jarak jauh yang memberikan fleksibilitas.

1.1.2 Analisa Masalah

Permasalahan utama dalam proses pengeringan maggot adalah penggunaan metode manual yang mengakibatkan penurunan efisiensi dan efektivitas kerja. Proses manual seringkali menghadapi beragam tantangan yang mengganggu konsistensi kualitas pengeringan maggot, hal ini dapat terjadi jika pengawasan kurang ketat. Selain itu, proses manual memakan waktu dan memerlukan banyak tenaga kerja, terutama pada skala produksi yang besar. *Monitoring* saat ini tergantung pada pemantauan langsung oleh pekerja. Masalah yang ada saat ini juga berdampak pada beberapa aspek yang menyebabkan tidak optimalnya proses pengeringan maggot, contohnya sebagai berikut:

1.1.2.1 Aspek Ekonomi

Dari segi ekonomi, adanya biaya tambahan yang harus dikeluarkan untuk membayar gaji dan upah pekerja, yang dapat meningkatkan beban finansial dalam operasional produksi maggot kering.

1.1.2.2 Aspek Manufakturabilitas

Proses pengeringan maggot dapat ditingkatkan kualitasnya dengan menambahkan teknologi *Internet of Things* (IoT) pada alat sehingga proses produksi maggot kering lebih optimal, yang pada akhirnya dapat mendukung efisiensi operasional secara keseluruhan.

1.1.2.3 Aspek Teknis

Secara teknis, proses pengeringan maggot yang ada saat ini masih dilakukan secara manual. Tanpa adanya sistem otomatisasi dengan memanfaatkan teknologi *Internet of Things* (IoT), sangat riskan diperoleh hasil pengeringan maggot yang kurang maksimal.

1.1.2.4 Aspek Produksi

Secara produksi, proses pembuatan maggot kering memerlukan pengawasan yang baik selama proses pengeringan dilakukan dengan memperhatikan waktu yang sesuai untuk memperoleh hasil maggot kering berkualitas tinggi.

1.1.3 Tujuan Capstone

Capstone ini dibuat dengan tujuan untuk mengoptimalkan proses pengeringan maggot dengan mengaplikasikan teknologi *Internet of Things* (IoT) supaya alat pengering maggot

memberikan keuntungan bagi peternak maggot pada pengoperasian alat dan lebih efisien, serta untuk menghasilkan maggot kering dengan kualitas yang bagus dan konsisten.

1.2 Analisa Solusi yang Ada

Solusi yang sudah ada sebagai berikut:

1.2.1 Rancang Bangun Sistem Pengering Maggot BSF Sebagai Alternatif Pakan Ternak Berbasis Iot (*Internet of Things*)

Pengeringan maggot secara otomatis dilakukan untuk menghemat tenaga, waktu produksi, dan menjaga kualitas dari maggot kering agar kualitasnya terjaga. Sistemnya adalah saat maggot sudah kering, maka akan muncul notifikasi di smartphone pengguna bahwa proses pengeringan maggot telah selesai dilakukan. Sistem pengeringan maggot secara otomatis ini terintegrasi dengan aplikasi telegram. Jadi notifikasi dikirimkan ke aplikasi telegram pengguna. Untuk proses pengeringannya diatur dengan waktu kurang lebih 3 jam dengan suhu 65°C serta kelembaban berada pada kisaran 16 persen[1]. Namun pada solusi ini, masih menggunakan aplikasi telegram untuk *monitoring* proses pengeringan maggotnya, sehingga apabila aplikasi telegram sedang mengalami gangguan maka tidak bisa melakukan *monitoring* pada proses pengeringan maggot.

1.2.2 Proteksi Motor Induksi Satu Fasa Terhadap Kenaikan Suhu Pada Pengering Maggot Berbasis Panel Surya

Alat pengering maggot yang menggunakan motor induksi satu fasa memiliki sistem perawatan yang sangat mudah. Oleh karena itu pada penelitian ini membuat sebuah sistem *monitoring* untuk motor induksi satu fasa yang bertujuan untuk melindungi motor induksi fasa dari kenaikan suhu yang berlebih dan menjaga supaya motor induksi satu fasa dalam kondisi tetap stabil pada saat proses pengeringan maggot dengan menggunakan sensor DS18B20[2]. Namun pada solusi ini bertujuan untuk *monitoring* motor induksi fasa, bukan *monitoring* proses pengeringan maggot yang bertujuan untuk mendapatkan hasil maggot kering yang maksimal.

BAB 2

SPESIFIKASI DAN BATASAN SOLUSI

2.1 Dasar Penentuan Spesifikasi

Alat pengering maggot otomatis berbasis *Internet of Things* (IoT) menggunakan beberapa sensor seperti sensor suhu dan menggunakan mikrokontroler untuk mengumpulkan data dari sensor yang kemudian diolah di *database*. Pembuatan alat ini bertujuan untuk mendapatkan hasil maggot kering yang maksimal sesuai dengan keputusan menteri pertanian No.240/Kpts/OT.210/4/2003 tentang cara pembuatan pakan yang baik yang mengatur tentang alat yang digunakan untuk memproduksi pakan harus dibuat dengan memenuhi persyaratan teknik, dan alat yang dibuat mudah untuk dioperasikan[3].

2.2 Batasan dan Spesifikasi

Beberapa solusi yang ada menggunakan mikrokontroler, menggunakan sensor suhu pada alat pengering maggot dengan suhu selama pengeringan di bawah 65°C. Alat pengering maggot otomatis berbasis *Internet of Things* (IoT) yang kita buat memiliki tujuan untuk mempermudah peternak maggot dalam *monitoring* proses pengeringan maggot melalui aplikasi *monitoring* maggot untuk mendapatkan hasil maggot kering yang maksimal. Pada aplikasi *monitoring* pengeringan maggot menggunakan *realtime database* untuk mengolah data yang kemudian ditampilkan pada aplikasi *monitoring*. Aplikasi *monitoring* akan memberikan notifikasi kepada pengguna saat proses pengeringan maggot telah selesai dilakukan.

Adapun batasan dan spesifikasi berdasarkan beberapa aspek sebagai berikut:

- Aspek Teknis

Pada aspek teknis, berfokus pada *monitoring* proses pengeringan maggot dengan menggunakan parameter suhu untuk *monitoring* kondisi di dalam alat pengering maggot dan membuat ketentuan waktu pengeringan untuk setiap berat maggot yang akan dikeringkan sehingga mampu memberikan informasi yang akurat mengenai tingkat kekeringan maggot. Ketika waktu untuk pengeringan pada setiap berat maggot yang dikeringkan telah habis maka alat akan secara otomatis menghentikan proses pengeringan. Penggunaan mikrokontroler bertugas untuk mengumpulkan data yang relevan dan mengirimkannya ke server dan data tersebut akan dianalisis lebih lanjut.

- Aspek Ekonomi

Pada aspek ekonomi, berfokus pada bagaimana alat ini bisa meningkatkan kualitas maggot kering serta dapat mengurangi beban biaya operasional yang dikeluarkan untuk membayar upah pekerja dalam proses pengeringan maggot. Dengan adanya alat ini proses pengeringan maggot tidak lagi memerlukan pekerja untuk mengawasi jalannya proses pengeringan maggot karena bisa dilakukan *monitoring* dengan hanya melalui aplikasi *monitoring* saja.

- Aspek Manufakturabilitas

Pada aspek Manufakturabilitas, berfokus pada perancangan alat. Perancangan alat dipermudah dengan ketersediaan komponen yang mudah didapatkan.

2.3 Pengukuran/Verifikasi Spesifikasi

Berdasarkan bab sebelumnya tentang batas dan dasar penentuan spesifikasi yang ingin dicapai, alat ini dibuat untuk mempermudah peternak maggot untuk *monitoring* proses pengeringan maggot. Berikut merupakan spesifikasi dan pengukuran dari alat yang akan digunakan dalam penelitian:

2.3.1 Verifikasi spesifikasi Sensor Suhu

Tabel 2. 1 Tabel Verifikasi Spesifikasi Sensor Suhu

Parameter	Penjelasan
Rincian	Output yang dihasilkan oleh sensor suhu antara 0°C - 200°C.
Mekanisme Pengujian	Untuk mengukur akurasi hasil dari sensor suhu mahasiswa melakukan perbandingan dengan alat ukur yang sudah ada di pasaran yaitu thermometer digital.
Parameter Keberhasilan	Akurasi yang didapat dari sensor suhu saat dibandingkan dengan thermometer digital mendekati 100%

2.3.2 Verifikasi spesifikasi Motor Servo

Tabel 2. 2 Verifikasi Spesifikasi Motor Servo

Parameter	Penjelasan
Rincian	Motor Servo dapat berputar dengan stabil selama proses pengeringan berlangsung.
Mekanisme Pengujian	Pengukuran motor servo dilakukan dengan menggunakan alat tachometer yang akan mengukur kecepatan putaran dari motor servo dan melihat kecepatan motor servo stabil atau tidak.
Parameter Keberhasilan	Proses pengeringan dapat dikatakan berhasil jika motor servo dapat berputar dengan kecepatan yang stabil selama proses pengeringan berlangsung dan saat diukur menggunakan tachometer.

2.3.3 Verifikasi Spesifikasi Pemantik Api dan *Valve solenoid*

Tabel 2. 3 Verifikasi Spesifikasi Pemantik Api dan *Valve solenoid*

Parameter	Penjelasan
Rincian	<i>Valve solenoid</i> dapat mengalirkan gas saat diberikan tegangan dan pemantik api dapat memercikan aliran listrik juga saat diberikan tegangan.
Mekanisme Pengujian	<i>Valve solenoid</i> diuji dengan mengalirkan gas ke <i>valve solenoid</i> untuk menguji fungsinya yaitu sebagai perantara aliran gas yang dialirkan ke pemantik dan kompor. Pemantik diuji dengan cara memberikan tegangan listrik ke pemantik untuk menguji fungsinya yaitu dapat memercikan aliran listrik yang dapat memantik api.
Parameter Keberhasilan	Api dapat menyala saat <i>valve solenoid</i> mengalirkan gas ke pemantik dan pemantik dapat memercikan listrik yang dapat memantik api.

2.3.4 Verifikasi Spesifikasi Aplikasi

Tabel 2. 4 Verifikasi Spesifikasi Aplikasi

Parameter	Penjelasan
Rincian	Aplikasi dapat mengoperasikan alat pengering maggot dan <i>monitoring</i> proses pengeringan yang sedang terjadi.
Mekanisme Pengujian	Pengukuran aplikasi dilakukan menggunakan Wireshark untuk menghitung <i>quality of service</i> dari aplikasi.
Parameter Keberhasilan	Aplikasi memiliki <i>quality of service</i> yang baik berupa <i>delay</i> yang minim dan <i>throughput</i> yang besar.

BAB 3

DESAIN RANCANGAN SOLUSI

3.1 Alternatif Usulan Solusi

Proses pengeringan maggot berbasis *Internet of things* (IoT) dapat ditingkatkan dengan mengintegrasikan teknologi dan otomatisasi untuk meningkatkan efisiensi, kontrol, dan pemantauan. Berikut adalah beberapa alternatif usulan solusi untuk meningkatkan kualitas hasil pengeringan maggot. Pada perancangan sistem yang digunakan sebagai media monitoring proses pengeringan maggot menggunakan platform web berbasis aplikasi atau aplikasi *mobile*. Sedangkan untuk mikrokontroler menggunakan ESP8266 atau Raspberry Pi dan untuk koneksi menggunakan Wi-Fi. Alternatif solusi terkait masalah yang akan diselesaikan ada tiga yaitu *mobile app* dengan ESP8266, *web app* dengan ESP8266, *mobile app* dengan Raspberry Pi.

3.1.1 Alternatif Solusi 1

Pada alternatif usulan solusi yang pertama fitur utama yang digunakan adalah *monitoring* pengeringan maggot berbasis IoT secara *realtime* menggunakan mikrokontroler ESP8266 yang diintegrasikan dengan *mobile application* dengan koneksi Wi-Fi. *Mobile application* (aplikasi seluler) diunggulkan dalam konteks *Internet of things* (IoT) karena berbagai alasan, termasuk keunggulannya dalam hal aksesibilitas, antarmuka pengguna yang ramah, dan kemampuannya untuk menyediakan pengalaman yang terintegrasi dan mudah digunakan[4]. Berikut merupakan prosedur pengoperasian pada solusi ini:

1. Hubungkan ESP8266 dengan sensor suhu dan juga ESP8266 diprogram untuk dapat membaca data dari sensor suhu dan juga mengontrol motor servo.
2. Hubungkan *mobile application* dengan *platform cloud* atau *server* untuk menyimpan data.
3. Data suhu yang ada pada *database* dapat diakses melalui aplikasi seluler.
4. Desain antarmuka aplikasi berisi informasi suhu, kolom untuk memasukkan informasi berat maggot, serta status pengeringan.

3.1.2 Alternatif Solusi 2

Pada alternatif usulan solusi yang kedua fitur utama yang digunakan adalah *monitoring* pengeringan maggot berbasis *Internet of Things* (IoT) secara *realtime* menggunakan mikrokontroler ESP8266 yang diintegrasikan dengan *web application* dengan koneksi Wi-Fi. Aplikasi berbasis web atau *web application* memiliki beberapa keunggulan yang menjadikannya pilihan yang diunggulkan dalam beberapa konteks. Alasan utama mengapa

aplikasi berbasis web seringkali lebih diunggulkan karena *web application* tidak memerlukan instalasi dan juga dapat diakses diberbagai perangkat tanpa harus melihat komabilitas perangkat. Berikut merupakan prosedur pengoperasian pada solusi ini:

1. Hubungkan ESP8266 dengan sensor suhu dan juga ESP8266 diprogram untuk dapat membaca data dari sensor suhu dan juga mengontrol motor servo.
2. Hubungkan *web application* dengan *platform cloud* atau server untuk menyimpan data.
3. Data suhu yang ada pada *database* dapat diakses melalui *web application*.
4. Desain antarmuka web yang mencakup informasi suhu, serta status pengeringan.

3.1.3 Alternatif Solusi 3

Pada alternatif usulan solusi yang pertama fitur utama yang digunakan adalah *monitoring* pengeringan maggot berbasis *Internet of Things* (IoT) secara *realtime* menggunakan mikrokontroler Raspberry Pi yang diintegrasikan dengan *mobile application* dengan koneksi Wi-Fi. Raspberry Pi mampu bekerja layaknya computer dengan menjalankan sistem operasi Linux. Raspberry Pi memiliki sistem operasi Raspbian yang memiliki beberapa program standar dan program pembantu untuk menjalankan perangkat keras dari Raspberry Pi[5]. Berikut merupakan prosedur pengoperasian pada solusi ini:

1. Hubungkan sensor suhu dengan Raspberry Pi dan juga diprogram untuk dapat membaca data dari sensor suhu dan juga mengontrol motor servo.
2. Hubungkan aplikasi seluler dengan platform cloud atau server untuk menyimpan data.
3. Data suhu yang ada pada *database* dapat diakses melalui aplikasi seluler.
4. Desain antarmuka aplikasi berisi informasi suhu, kolom untuk memasukkan informasi berat maggot, serta status pengeringan.

3.2 Analisis dan Pemilihan Solusi

Metode pemilihan solusi yang digunakan adalah matriks pemilihan. Parameter yang digunakan untuk menentukan solusi adalah biaya (terjangkaunya harga), kemampuan komputasi, komabilitas perangkat, kemudahan akses, *delay*. Berikut adalah matriks pada solusi yang terpilih:

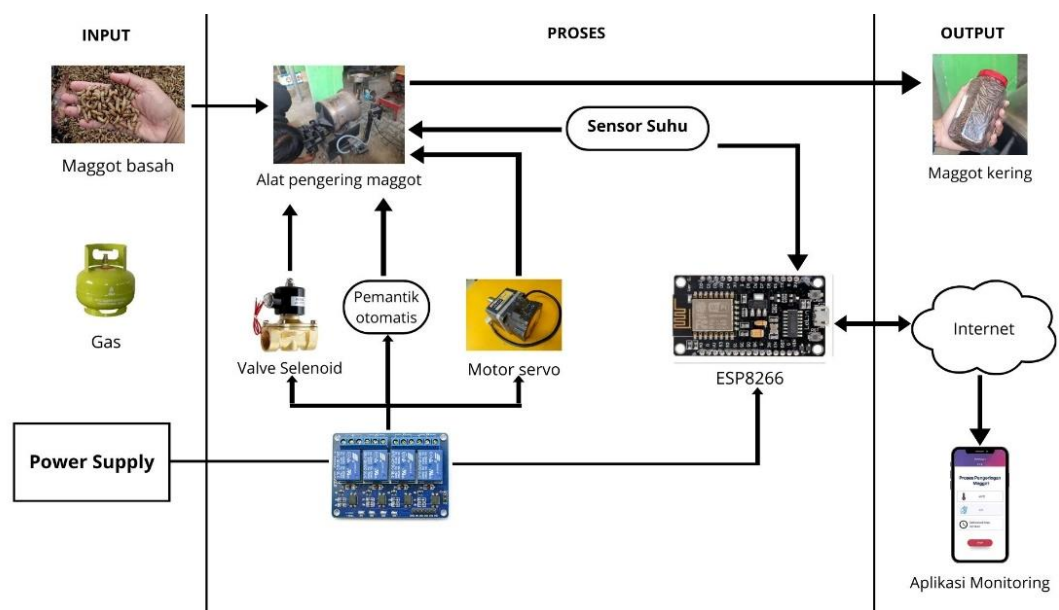
Tabel 2. 5 Perbandingan Antar Solusi

Kriteria Seleksi		Bobot	ESP8266 dan Mobile App		ESP8266 dan Web App		Raspberry Pi dan Mobile App	
			Rating	Nilai Bobot	Rating	Nilai Bobot	Rating	Nilai Bobot
Perbandingan Mikrokontroler	Harga	50%	5	2,5	5	2,5	2	1
	Kemampuan Komputasi	25%	3	0,75	3	0,75	5	1,25
	Kompatibilitas Perangkat	25%	5	1,25	5	1,25	5	1,25
Media Monitoring	Kompatibilitas Prangkat	25%	3	0,75	5	1,25	3	0,75
	Delay	30%	4	1,2	3	0,9	4	1,2
	Kemudahan Akses	40%	5	2,25	3	1,35	5	2,25
Total Nilai			8,7		8		7,7	
Peringkat			1		2		3	
Lanjutkan?			Ya		Tidak		Tidak	

3.3 Desain Solusi Terpilih

Berdasarkan alternatif solusi yang ada, desain solusi yang terpilih berdasarkan matriks perbandingan adalah menggunakan ESP8266 dan menggunakan *mobile application*. Berikut merupakan penjelasan dari solusi yang terpilih.

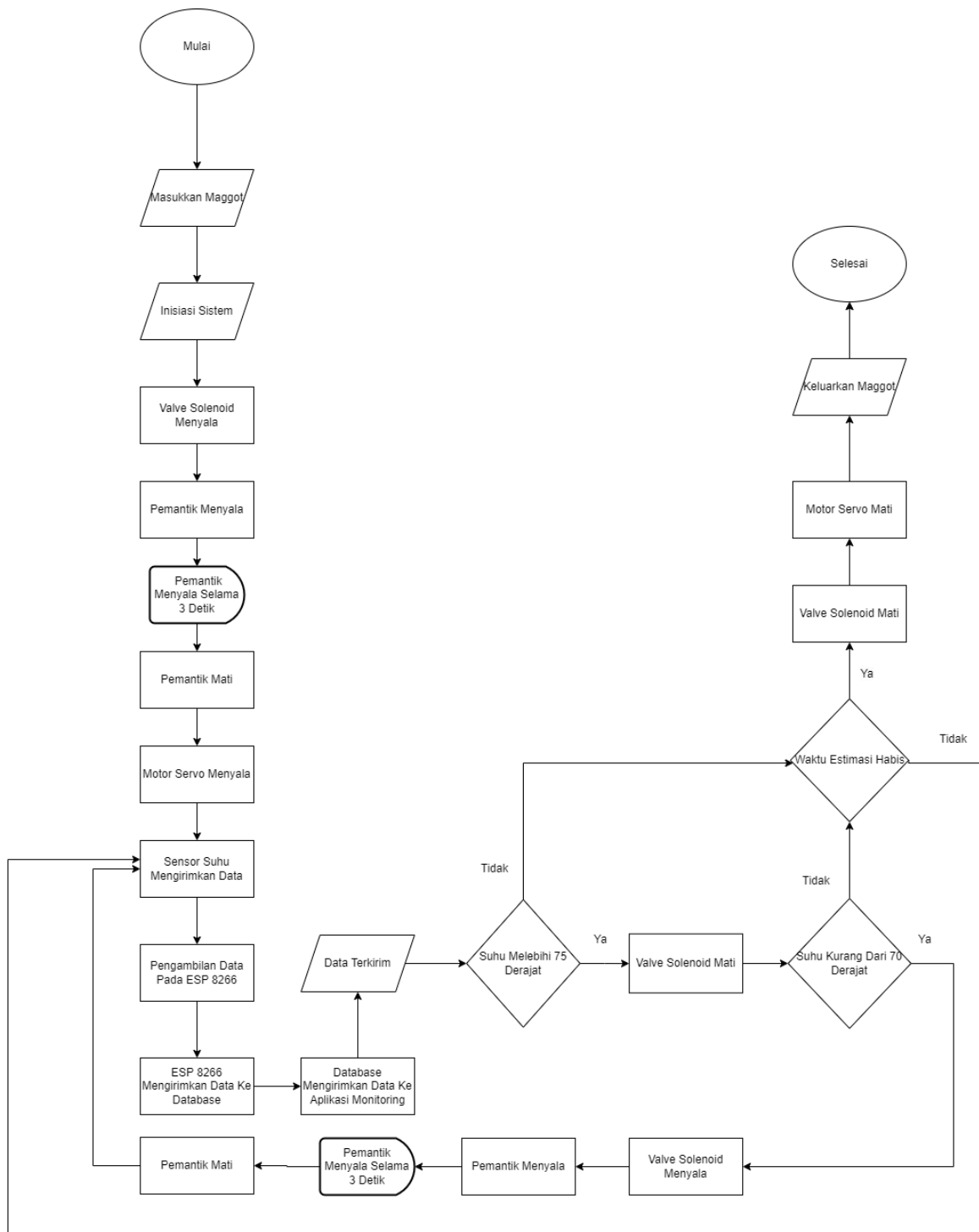
3.3.1 Diagram Alur Sistem



Gambar 3.1 Diagram Alur Sistem

Pada Gambar 3.1 terdapat diagram alur sistem yang memiliki tiga bagian yaitu *input*, proses dan *output*. Bagian pertama adalah input maggot basah yang akan di proses menjadi maggot kering, yang kedua ada *power supply* yang berfungsi sebagai input daya untuk motor servo dan ESP8266, yang ketiga adalah gas lpg untuk kompor yang dihubungkan melalui *valve solenoid* dan pemantik otomatis. Pada bagian proses, *valve solenoid* akan menerima arus listrik dan akan membuka aliran gas kemudian pemantik akan menyalakan kompor, setelah itu motor servo akan berputar dan proses pengeringan akan berlangsung. Proses ini semua dikendalikan oleh mikrokontroler ESP8266 yang terhubung melalui aplikasi. Pada saat proses pengeringan sensor suhu yang ada pada alat pengering akan mengirimkan data ke mikrokontroler ESP8266. Berat maggot dan waktu merupakan parameter-parameter yang dibutuhkan untuk proses pengeringan ini. Pada bagian output terdapat maggot kering, *monitoring* secara *realtime* proses pengeringan dan juga notifikasi pada aplikasi. *Valve solenoid* akan menghentikan aliran gas dan motor servo akan berhenti berputar beberapa saat setelah itu dan maggot kering sudah siap untuk diambil. Beberapa menit sebelum itu ESP8266 mengirimkan data ke aplikasi dan aplikasi mengirimkan notifikasi. Hal ini bertujuan agar maggot diambil pada saat yang tepat sehingga kualitas maggot kering maksimal.

3.3.2 Flowchart Proses Pengeringan Maggot

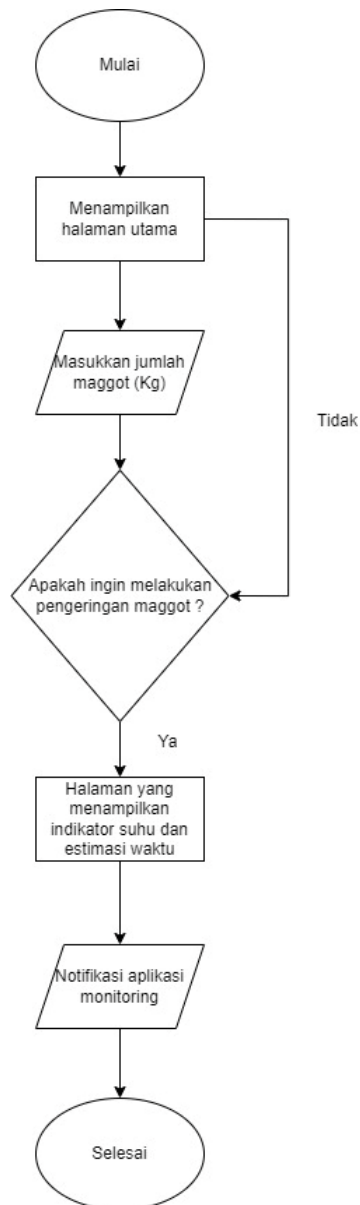


Gambar 3.2 Flowchart Proses Pengeringan Maggot

Pada Gambar 3.2 terdapat *flowchart* yang menjelaskan alur kerja dari alat pengering maggot mulai dari awal sampai akhir proses pengeringan maggot. Alur pada *Flowchart* tersebut dimulai dengan inisiasi sistem, jika proses inisiasi sudah dilakukan maka proses pengeringan akan dimulai. Ketika proses pengeringan dimulai ESP8266 akan mengambil data

dan akan mengirimkan data ke *database* yang dilanjutkan ke aplikasi *monitoring*. Pada alat pengering, sensor suhu juga berfungsi sebagai acuan jika suhu saat pengeringan di atas 75°C maka api akan mati dan akan nyala kembali ketika suhu 70°C, hal ini bertujuan untuk menjaga suhu agar stabil dibawah 70°C. Jika waktu pengeringan maggot belum selesai maka proses pengeringan akan terus berlangsung, tetapi jika waktu pengeringan selesai maka aplikasi akan mengirimkan notifikasi dan alat pengering maggot akan berhenti beroperasi.

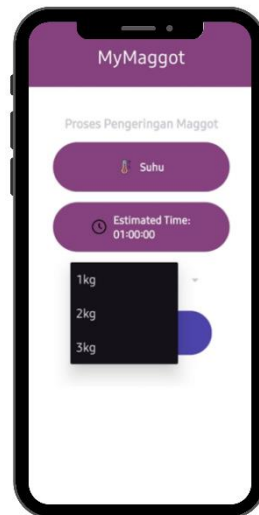
3.3.3 Flowchart Aplikasi



Gambar 3. 3 Flowchart Aplikasi

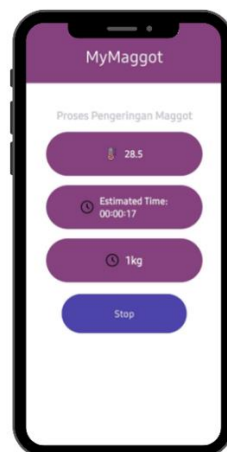
Pada Gambar 3.3 terdapat *Flowchart* aplikasi dengan alur kerja yang pertama adalah memulai aplikasi tersebut, setelah itu maka akan muncul halaman utama yang dapat diisi dengan jumlah maggot yang kita inginkan dalam jumlah kilogram, setelah itu lanjut ke proses pengeringan. Selama proses pengeringan, halaman utama aplikasi akan menampilkan indikator suhu dan estimasi waktu pengeringan, setelah itu aplikasi akan memberi notifikasi saat proses pengeringan telah selesai.

3.3.4 UI Desain Aplikasi



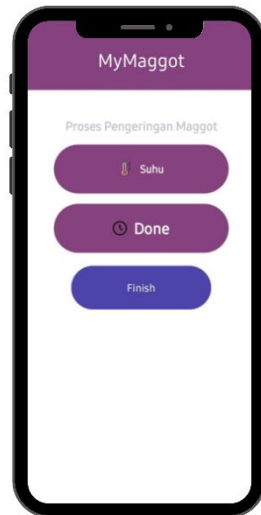
Gambar 3. 4 Tampilan Awal Aplikasi

Pada Gambar 3.4 merupakan tampilan awal saat dibuka aplikasi MyMaggot. Pada halaman utama aplikasi terdapat tiga bagian yang menampilkan data suhu yang belum ada nilainya karena proses pengeringan belum berjalan. Bagian berikutnya adalah data kelembaban maggot BSF. Pada bagian berikutnya adalah untuk *input* berat maggot dalam skala kilogram untuk dikeringkan. Tombol start digunakan untuk memulai proses pengeringan.



Gambar 3. 5 Tampilan Aplikasi Saat Proses Pengeringan Berlangsung

Gambar 3.5 adalah tampilan ketika proses pengeringan sedang berjalan. Terlihat bahwa pada gambar diatas, nilai suhu telah ditampilkan berdasarkan data *realtime* selama proses pengeringan maggot. Pada bagian berat maggot berganti menjadi estimasi waktu lamanya proses pengeringan maggot akan selesai. Pada bagian paling bawah terdapat tombol *stop* untuk menghentikan proses pengeringan saat masih berjalan.



Gambar 3. 6 Tampilan Aplikasi Saat Proses Pengeringan Selesai

Gambar 3.6 adalah tampilan terakhir pada aplikasi MyMaggot. Pada tampilan ini yang berbeda terletak pada estimasi waktu yang berganti menjadi tampilan *DONE* yang menunjukkan bahwa proses pengeringan maggot sudah selesai. Pada tombol *finish* berfungsi untuk kembali ke halaman utama untuk memulai pengeringan maggot berikutnya.

3.4 Jadwal dan Anggaran

Berikut adalah jadwal kegiatan yang akan dilakukan selama 6 bulan kedepan. Terdapat berbagai kegiatan seperti:

1. Survey alat pengering maggot
2. Pemilihan komponen alat pengering
3. Perancangan sistem pengoperasian alat
4. Perancangan solenoid valve & pemantik gas
5. Perancangan motor servo
6. Perancangan sensor suhu
7. *Wiring* alat, komponen dan pemograman
8. Pembuatan UI Design Aplikasi
9. Pembuatan program & *database* aplikasi

- 10. Pengujian aplikasi
- 11. Pengujian aplikasi yang sudah dihubungkan dengan alat
- 12. Pengoptimalan proses pengeringan.

Berikut ini adalah tabel kegiatan selama 6 bulan kedepan :

Tabel 3. 1 Jadwal Kegiatan Selama 6 Bulan

	Desember				Januari				Februari				Maret				April				Mei				Juni			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Survey Alat Pengerig Maggot																												
Pemilihan Komponen Alat Pengerig																												
Perancangan Sistem Pengoperasian Alat																												
Perancangan Valve & Pemantik Gas																												
Perancangan Motor Servo																												
Perancangan Sensor Suhu																												
Wiring Alat, Komponen dan Pemograman																												
Pembuatan UI Design Aplikasi																												
Pembuatan Program & Database Aplikasi																												
Pengujian Aplikasi																												
Pngujian Aplikasi yang Sudah Dihubungkan dengan Alat																												
Pengoptimalan Proses Pengeringan																												

Dalam proses pembuatan alat ini terdapat beberapa alat yang kami beli. Berikut adalah tabel anggaran dari alat yang kami beli yaitu :

Tabel 3. 2 Anggaran Dana

Alat dan Komponen	Harga
Alat Pengerig Maggot	Rp. 1.500.000
Valve Solenoid	Rp. 150.000
ESP8266	Rp. 60.000
Sensor Suhu	Rp. 100.000
Motor Servo	Rp. 900.000
Total	Rp. 2.710.000

BAB 4

IMPLEMENTASI

4.1 Deskripsi Umum Implementasi

Alat pengering maggot berbasis *Internet of Things* (IoT) digunakan untuk mengeringkan maggot dengan proses pengeringan maggot yang dapat dilakukan secara otomatis dan efisien, sehingga menghasilkan maggot kering yang berkualitas tinggi dengan memanfaatkan teknologi *Internet of things* (IoT). Selain itu, dilengkapi juga dengan aplikasi *monitoring* yang memungkinkan untuk kontrol proses pengeringan maggot menjadi lebih mudah. Implementasi alat pengering maggot berbasis *Internet of Things* (IoT) melibatkan beberapa tahapan prosedur untuk mewujudkan rancangan solusi yang sudah dibuat. Berikut adalah penjelasannya:

1. Perencanaan Alat

Membuat perencanaan alat pengering maggot berbasis *Internet of Things* (IoT) dengan mengidentifikasi kebutuhan komponen seperti motor servo, pemantik otomatis, *valve solenoid*, dan sensor suhu, serta membuat skema untuk menghubungkan antar komponen

2. Pembuatan Prototipe Alat

Langkah selanjutnya adalah pembuatan prototipe alat pengering maggot dengan menggunakan komponen-komponen sesuai yang dibutuhkan.

3. Uji Kelayakan Alat

Prototipe yang telah dibuat selanjutnya diuji untuk memastikan fungsionalitasnya sesuai dengan yang direncanakan. Pengujian dilakukan untuk mengecek semua dapat berjalan dengan normal seperti motor servo, pengukuran suhu oleh sensor suhu yang akurat, pengendalian pemantik otomatis dan *valve solenoid*.

4. Pengembangan Aplikasi *Monitoring*

Selain *hardware*, implementasi juga melibatkan pengembangan aplikasi *monitoring* untuk mengontrol alat pengering maggot melalui gadget.

5. Wiring Komponen dan Integrasi Alat dengan Aplikasi *Monitoring*

Setelah semua komponen *hardware* dan aplikasi telah siap, selanjutnya adalah mengintegrasikan semua komponen tersebut supaya dapat bekerja secara bersama-sama sesuai dengan rancangan.

6. Uji Coba Keseluruhan dan Evaluasi

Alat pengering maggot dan aplikasi *monitoring* yang sudah jadi kemudian diuji coba untuk memastikan bahwa berfungsi dengan baik. Evaluasi dilakukan untuk menilai kinerja dari alat pengering maggot serta aplikasi *monitoring* sudah berfungsi sesuai yang diharapkan atau masih terdapat kesalahan pada kinerja alat dan aplikasi. Dengan mengikuti tata cara tersebut, pembuatan alat pengering maggot berbasis *Internet of Things* (IoT) dapat diimplementasikan dengan sukses dan memberikan manfaat yang maksimal dalam proses pembuatan maggot kering bagi petani maggot.

4.2 Detil Implementasi

Implementasi ini merupakan integrasi antara perangkat keras (mikrokontroler Arduino, sensor MAX6675 dan alat pengering maggot) dengan perangkat lunak (Firebase dan juga aplikasi) untuk *monitoring* dan mengendalikan sistem pengering maggot. Mikrokontroler berfungsi untuk mengumpulkan data dari sensor dan komponen yang ada pada alat pengering maggot yang selanjutnya dikirim ke databse. Pada Arduino IDE dibuat *source code* untuk memberikan perintah sensor dan komponen yang ingin dikirim ke database yaitu sensor suhu, valve solenoid, pemantik otomatis, dan motor servo. Berikut adalah penjelasan detail terkait implementasi ini:

4.2.1 Software

Berikut adalah penjelasan *source code* pada arduino IDE dan Android Studio untuk pembuatan aplikasi *monitoring* serta menggunakan *realtime database* Firebase.

4.2.1.1 Arduino IDE

Berikut adalah inklusi *library* dan deklarasi variabel:

```
#include "max6675.h"
#include <Arduino.h>
#if defined(ESP32)
    #include <WiFi.h>
#elif defined(ESP8266)
    #include <ESP8266WiFi.h>
#endif
#include <Firebase_ESP_Client.h>
#include "addons/TokenHelper.h"
#include "addons/RTDBHelper.h"
// Insert your network credentials
```

```

#define WIFI_SSID "ale"
#define WIFI_PASSWORD "icadbronze"
// Insert Firebase project API Key
#define API_KEY "AIzaSyBehW88DGQOht6-o8WBSRSVazDk4MFi46Y"
// Insert RTDB URL define the RTDB URL
#define DATABASE_URL "https://mesin-pengering-maggot-default-
rtdb.asia-southeast1.firebaseio.com/"

```

Program dimulai dengan meng-include beberapa *library* yang diperlukan, termasuk max6675.h untuk sensor MAX6675, *library* Wi-Fi untuk koneksi jaringan, dan *library* Firebase untuk koneksi dengan Firebase RTDB. Terdapat kredensial jaringan Wi-Fi dan konfigurasi Firebase seperti SSID dan password Wi-Fi, API Key Firebase, serta URL Firebase RTDB.

```

// Thermocouple MAX6675 pins
int SO = D7;    // Data Out
int CS = D6;    // Chip Select
int SCLK = D5; // Clock
int pin1 = D1; // Valve solenoid
int pin2 = D2; // Pemantik Elektrik
int pin3 = D3; // Motor Servo

MAX6675 thermocouple(SCLK, CS, SO);

bool emergencyStop = false;
bool signupOK = false;
unsigned long previousMillis = 0; // Waktu sebelumnya
const long interval1 = 90 * 60000; // Durasi proses pengeringan
jika berat maggot 1 kg
const long interval2 = 110 * 60000; // Durasi proses pengeringan
jika berat maggot 2 kg
const long interval3 = 140 * 60000; // Durasi proses pengeringan
jika berat maggot 3 kg

// Firebase Data object
FirebaseData fbdo;

```

```

FirebaseAuth auth;
FirebaseConfig config;

String sValue, sValue2, sValue3, sValue4, sValue5, sValue6, sValue7;

```

Pada *source code* diatas mendefinisikan pin untuk sensor termokopel MAX6675 (SO, CS, SCLK) dan pin-penyalakan (*valve solenoid*, pemantik otomatis, motor servo) serta variabel-variabel penting untuk terkoneksi dengan Firebase.

Berikut adalah void setup :

```

void setup() {
  Serial.begin(9600);

  pinMode(pin1, OUTPUT); // set pin 1 sebagai output
  pinMode(pin2, OUTPUT); // set pin 2 sebagai output
  pinMode(pin3, OUTPUT); // set pin 3 sebagai output

  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("Connecting to Wi-Fi");
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(300);
  }
  Serial.println();
  Serial.print("Connected with IP: ");
  Serial.println(WiFi.localIP());
  Serial.println();

  /* Assign the api key (required) */
  config.api_key = API_KEY;

  /* Assign the RTDB URL (required) */
  config.database_url = DATABASE_URL;

  /* Sign up */
  if (Firebase.signUp(&config, &auth, "", "")) {

```

```

Serial.println("ok");

signupOK = true;

}

else {

    Serial.printf("%s\n", config.signer.signupError.message.c_str());

}

/* Assign the callback function for the long running token
generation task */

config.token_status_callback = tokenStatusCallback; // see
addons/TokenHelper.h

Firebase.begin(&config, &auth);

Firebase.reconnectWiFi(true);

}

```

Di dalam *setup function* dijelaskan set pin 1,2 dan 3 sebagai *output* dan koneksi Wi-Fi ditetapkan serta konfigurasi Firebase dilakukan. Proses ini juga mencakup proses *sign-up* untuk otentikasi.

Berikut adalah void loop :

```

// Membaca suhu dari MAX6675

float temperatureC = thermocouple.readCelsius();

if (temperatureC != -1.0) {

    if (Firebase.ready() && signupOK) {

        if (Firebase.RTDB.setFloat(&fbdo,
"Temperature",temperatureC)){ // mengirim data suhu ke firebase

            Serial.print("Temperature: ");

            Serial.print(temperatureC);

            Serial.println(" °C");

        }

    }

    else {

        Serial.println("FAILED");

        Serial.println("REASON: " + fbdo.errorReason());

    }

}

```

```
}
```

Source code untuk membaca suhu dari MAX6675 dan mengirim data suhu ke Firebase.

```
if (Firebase.RTDB.getString(&fbdo, "valve")) { // Proses membuka
dan menutup valve solenoid

    if (fbdo.dataType() == "string") {
        sValue = fbdo.stringData();
        int a = sValue.toInt();
        Serial.println(a);
        if (a == 1){
            digitalWrite(pin1, HIGH);
        }else{
            digitalWrite(pin1, LOW);
        }
    }
}
else {
    Serial.println(fbdo.errorReason());
}
```

Source code di atas berfungsi untuk mengontrol *valve solenoid*.

```
if (Firebase.RTDB.getString(&fbdo, "pemanitik")) { // proses
menghidupkan dan mematikan pemanitik otomatis

    if (fbdo.dataType() == "string") {
        sValue2 = fbdo.stringData();
        int b = sValue2.toInt();
        Serial.println(b);
        if (b == 1){
            digitalWrite(pin2, HIGH);
            delay(3000);
            digitalWrite(pin2, LOW);
            delay(2000);
        }else{
            digitalWrite(pin2, LOW);
        }
    }
}
```

```

    }
  }
}
else {
  Serial.println(fbdo.errorReason());
}

```

Source code di atas berfungsi untuk mengontrol pemantik otomatis.

```

if (Firebase.RTDB.getString(&fbdo, "motor servo")) { // proses
menyalakan motor servo

  if (fbdo.dataType() == "string") {
    sValue3 = fbdo.stringData();
    int c = sValue3.toInt();
    Serial.println(c);
    if (c == 1){
      digitalWrite(pin3, HIGH);
    }else{
      digitalWrite(pin3, LOW);
    }
  }
}
else {
  Serial.println(fbdo.errorReason());
}

```

Source code di atas berfungsi untuk mengontrol motor servo.

```

if (temperatureC > 75) {
  digitalWrite(pin1, LOW);
}
if (temperatureC < 70 && valve == LOW) {
  digitalWrite(pin1, HIGH);
  delay(750);
  digitalWrite(pin2, HIGH);
  delay(3000);
  digitalWrite(pin2, LOW);
  delay(1500);
}

```

Source code di atas berfungsi untuk mengontrol suhu agar stabil pada saat pengeringan.

4.2.1.2 Android Studio untuk Pembuatan Aplikasi *Monitoring*

Pada pembuatan aplikasi monitoring menggunakan android studio dan terdapat tiga class yaitu *main activity*, *proses activity*, dan *proses selesai activity*. Pada *main activity* terdapat “ref1kg”, “ref2kg”, dan “ref3kg” yang merupakan variabel yang digunakan untuk merujuk ke *database* Firebase yang diinisialisasi dengan referensi ke tempat di mana data berat disimpan dalam *database* Firebase. Pada *main activity* terdapat source code untuk mengelola data berat dan waktu estimasi proses. *Main activity* memiliki komponen utama seperti *dropdown spinner* untuk memilih berat, *text view* untuk menampilkan waktu estimasi, dan tombol “Start” untuk memulai proses. Data dari Firebase secara *realtime* akan dikirimkan untuk diisi ke dalam *spinner*, dan waktu estimasi diatur berdasarkan berat yang dipilih. Ketika tombol “Start” ditekan, nilai di Firebase disesuaikan dengan berat yang dipilih sebelum memulai proses *activity* untuk memproses data lebih lanjut.

Pada *proses activity* terdapat sebuah *switch case* yang mengecek nilai dari “selectedBerat” dengan penjelasan sebagai berikut :

- Jika “selectedBerat” adalah “1kg”, maka nilai dari *node* “1kg” di Firebase diatur menjadi “0”.
- Jika “selectedBerat” adalah “2kg”, maka nilai dari *node* “2kg” di Firebase diatur menjadi “0”.
- Jika “selectedBerat” adalah “3kg”, maka nilai dari *node* “3kg” di Firebase diatur menjadi “0”.

Proses activity menampilkan waktu estimasi berdasarkan berat yang dipilih dari *main activity*, dan memulai sebuah “*CountdownTimer*” untuk menghitung mundur waktu estimasi pengeringan yang sedang dilakukan. Selama proses pengeringan dan hitung mundur, aplikasi secara *realtime* membaca data suhu dari Firebase dan menampilkannya. Ketika waktu estimasi habis, *proses activity* akan mengubah *value* di Firebase menjadi 0 dan pada aplikasi menampilkan notifikasi bahwa proses selesai, dan membunyikan *alarm* untuk memberi tanda bahwa proses pengeringan telah selesai dilakukan. Pada class ini juga diatur saat tombol stop ditekan maka akan memproses “refemergency” yang akan merujuk pada *database* Firebase dan mengatur *value* pada Firebase menjadi “1”.

Selesai *activity* adalah class yang dibuat untuk menyelesaikan proses dengan menghentikan status *emergency* yang sebelumnya diatur pada proses *activity* ketika tombol *stop* ditekan, serta mengirimkan *intent broadcast* untuk menghentikan *alarm* yang sedang berbunyi. Setelah itu, pengguna dapat kembali ke *main activity* dengan menekan tombol *finsih*.

4.2.1.3 Realtime Database Firebase



```
https://mesin-pengering-maggot-default-rtdb.asia-southeast1.firebaseio.com/  
— 1kg: "0"  
— 2kg: "0"  
— 3kg: "0"  
— Temperature: 28.5  
— berat_1: "1kg"  
— berat_2: "2kg"  
— berat_3: "3kg"  
— emergency: "1"
```

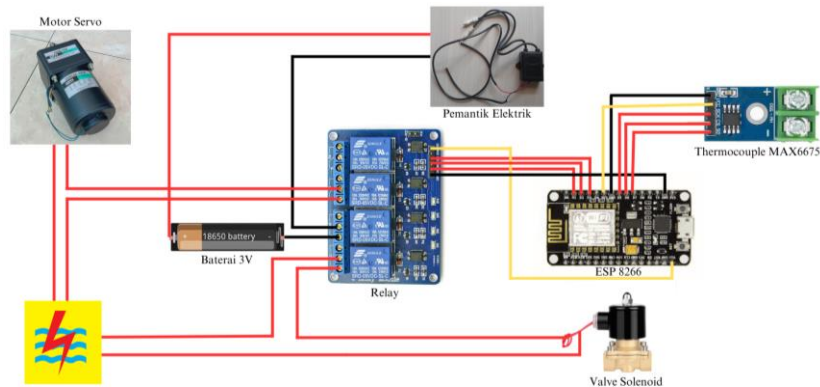
Gambar 4.1 Realtime Database Firebase

Gambar 4.1 merupakan tampilan pada Firebase yang merupakan *realtime database* yang kami gunakan untuk aplikasi *monitoring*. Berikut adalah penjelasannya :

1. “1kg”, “2kg”, “3kg” : Setelah kita memilih berat maggot yang akan dikeringkan dan klik *start* pada aplikasi, maka pada firebase *value* dari “1kg” atau “2kg” atau “3kg” akan berubah dari 0 menjadi 1, yang berarti bahwa alat pengering akan hidup dan memulai proses pengeringan, tetapi tidak berselang lama akan berubah kembali menjadi 0, hal ini supaya proses pengeringan tidak terjadi berulang atau *looping*.
2. “Temperature” : Data temperatur pada firebase akan ditampilkan pada aplikasi *monitoring*.
3. “berat_1”, “berat_2”, “berat_3” : *Database* ini digunakan pada aplikasi sebagai dropdown pilihan berat maggot yang akan dikeringkan. Pada aplikasi yang ditampilkan adalah value dari berat_1, berat_2, berat_3 yaitu 1kg, 2kg, 3kg.
4. “emergency” : Pada aplikasi apabila kita klik *stop* maka pada firebase *value* dari “emergency” akan berubah dari 0 menjadi 1, yang berarti bahwa proses pengeringan akan berhenti walaupun proses pengeringan belum selesai atau batal untuk dilakukan.

4.2.2 Hardware

Perangkat keras didesain dengan menghubungkan NodeMCU ESP8266, *relay*, sensor suhu *thermocouple* MAX6675, *valve solenoid*, pemantik otomatis, baterai dan motor servo. Kemudian dihubungkan ke Wi-Fi sebagai konektivitasnya. Gambar di bawah merupakan desain dari perangkat keras yang akan digunakan.



Gambar 4. 2 Desain Perangkat Keras

Berikut adalah tampilan dari alat pengering maggot yang akan berputar saat proses pengeringan dimulai.



Gambar 4. 3 Alat Pengering Maggot

Seperti terlihat pada Gambar 4.3, alat pengering memiliki komponen yang saling terintegrasi untuk menjalankan proses pengeringan. Komponen yang digunakan pada alat ini sebagai berikut:

4.2.2.1 ESP8266

Pada alat pengering yang kami buat menggunakan ESP8266 sebagai mikrokontroler untuk mengontrol semua perangkat keras. ESP8266 mempunyai 11 *port digital* dan 1 *port analog input*. Untuk logikanya menggunakan 3.3V dengan sumber tegangan 5V [6]. Fitur yang terdapat dalam NodeMCU ESP8266 antara lain *flash* sebesar 4MB, *clock* 80 / 160 MHz,

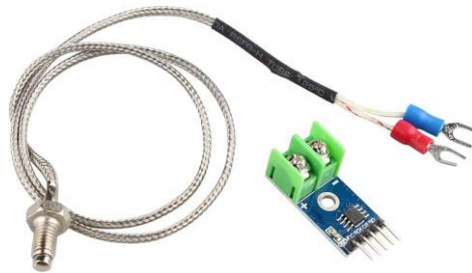
mempunyai 11 *port digital* dan 1 *port analog input*, untuk logikanya ESP8266 menggunakan 3.3V dengan sumber tegangan 5V. Sumber tegangan tersebut dikoneksikan ke *port Vin*. Sedangkan untuk sumber tegangan *external* bisa menggunakan VU atau sumber tegangan 5V dari NodeMCU. ESP8266 juga sudah memiliki modul Wi-Fi yang memungkinkan untuk melakukan konektivitas secara nirkabel. Berikut adalah bentuk dari ESP8266 yang digunakan.



Gambar 4. 4 ESP8266

4.2.2.2 MAX6675 Thermocouple Sensor

Sensor ini digunakan untuk membaca suhu dalam aplikasi pengering. *Thermocouple* MAX6675 mampu mengukur suhu sampai 1024°C[7]. Berikut adalah bentuk dari thermocouple MAX6675.



Gambar 4. 5 MAX6675 Thermocouple Sensor

4.2.2.3 Valve Solenoid

Valve solenoid digunakan untuk mengatur aliran gas yang menuju ke kompor. *Valve solenoid* bekerja pada tegangan 220VAC dan memiliki dimensi sebesar 75 x 40 x 40 mm dengan ukuran pipa sebesar 1/4 inci. Berikut adalah bentuk dari *valve solenoid*.



Gambar 4. 6 Valve solenoid

4.2.2.4 Pemantik Otomatis

Pemantik otomatis memiliki input sebesar 1,5V dan output sebesar 12KV dan berfungsi sebagai pemantik untuk menyalakan atau mematikan api dalam pengering. Berikut adalah bentuk dari *valve solenoid*.



Gambar 4. 7 Pemantik Otomatis

4.2.2.5 Motor Servo

Motor Servo berfungsi mengendalikan pergerakan mekanis alat pengering agar dapat berputar. Berikut adalah bentuk dari motor servo yang kami gunakan.



Gambar 4. 8 Motor Servo

4.3 Prosedur Pengoperasian

Berikut adalah prosedur pemakaian alat pengering maggot dengan aplikasi *monitoring* sebagai media pemantauan proses pengeringan maggot:

1. Hubungkan ESP8266, motor servo dan *valve solenoid* ke *power source*.



Gambar 4. 9 Komponen Terhubung ke *Power Source*

2. Pastikan ESP8266 dan aplikasi *monitoring* telah terhubung dengan koneksi internet.



Gambar 4. 10 ESP8266 dan Aplikasi Monitoring Terkoneksi Internet

3. Pastikan gas sudah terpasang pada selang gas yang terhubung ke *valve solenoid* dan nyalakan saklar pada baterai.



Gambar 4. 11 Gas Terpasang dan Saklar Baterai Hidup

4. Buka aplikasi *monitoring* kemudian pilih berat maggot yang akan dikeringkan.



Gambar 4. 12 Tampilan Utama Aplikasi

5. Setelah memasukkan berapa berat maggot yang akan dikeringkan, maka setelah itu proses pengeringan maggot akan langsung dimulai dengan ditampilkannya estimasi waktu pengeringan maggot.

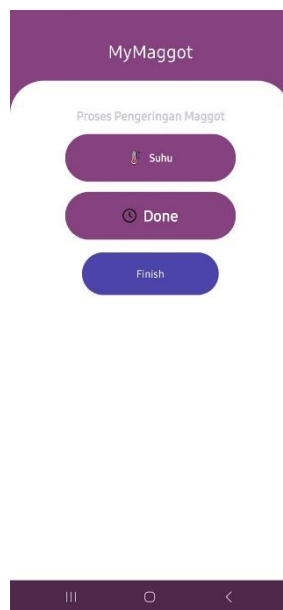


Gambar 4. 13 Tampilan Saat Proses Pengeringan Berlangsung



Gambar 4. 14 Pengeringan Maggot Berlangung

6. Setelah proses pengeringan telah selesai, maka pada aplikasi *monitoring* akan memberikan notifikasi.



Gambar 4. 15 Tampilan Saat Proses Pengeringan Selesai

7. Setelah proses pengeringan selesai, Tuangkan maggot yang telah kering pada wadah.



Gambar 4. 16 Proses Pengeringan Selesai

BAB 5

PENGUJIAN DAN KESIMPULAN

5.1 Skenario Umum Pengujian

Alat pengering maggot dengan teknologi *Internet of Things* (IoT) mampu menjadi solusi inovatif bagi peternak maggot dengan kemampuannya yang dapat dikendalikan secara otomatis melalui aplikasi monitoring serta dapat menghasilkan kualitas maggot kering yang bagus dan konsisten. Dalam upaya untuk memastikan keandalan dari alat pengering maggot yang kami buat, serangkaian pengujian yang komprehensif diperlukan. Skenario pengujian yang kami lakukan dirancang dengan mempertimbangkan berbagai aspek yang relevan untuk memvalidasi fungsionalitas dan memastikan kinerja optimal dari alat pengering maggot dengan teknologi *Internet of Things* (IoT) ini. Pada gambar di bawah ini merupakan bentuk dari alat pengering maggot.



Gambar 5. 1 Alat Pengering Maggot

5.1.1 Tujuan Pengujian

Pengujian alat pengering maggot dengan teknologi *Internet of Things* (IoT) bertujuan untuk memastikan bahwa komponen – komponen yang terdapat pada alat pengering maggot yang saling terintegrasi dan terhubung ke *database* serta aplikasi monitoring dapat bekerja dengan baik sesuai dengan yang diinginkan sehingga alat pengering maggot tersebut mampu memudahkan peternak maggot dalam menghasilkan maggot kering yang berkualitas.

5.1.2 Daftar Pengujian

Berikut adalah daftar pengujian pada alat pengering maggot:

1. Pengujian keandalan koneksi
2. Pengujian sensor suhu
3. Pengujian motor servo
4. Pengujian pemantik
5. Pengujian *valve solenoid*
6. Pengujian usabilitas aplikasi

5.1.3 Lokasi dan Waktu Pengujian

Pengujian dilakukan di kontrakan perumahan palem Dayeuhkolot pada tanggal 10 juni 2024.

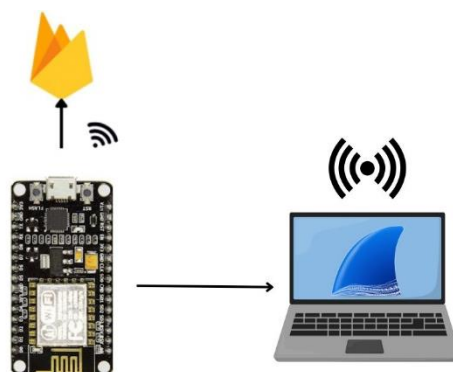
5.1.4 Pihak yang Terlibat

Penanggung jawab produksi maggot Green House Telkom University.

5.2 Detil Pengujian

5.2.1 Pengujian Keandalan Koneksi

Pada pengujian keandalan koneksi, penulis menggunakan laptop sebagai *access point* dan juga Wireshark sebagai *software* untuk menangkap data-data yang dikirim ke Firebase seperti terlihat pada Gambar 5.2 di bawah yang merupakan topologi pengujian koneksi. Pengujian menggunakan Wireshark dilakukan selama 3 menit dengan memfilter *ip adres* dari *access point* dan *ip adres* dari Firebase.



Gambar 5. 2 Topologi Pengujian Koneksi

5.2.1.1 Delay

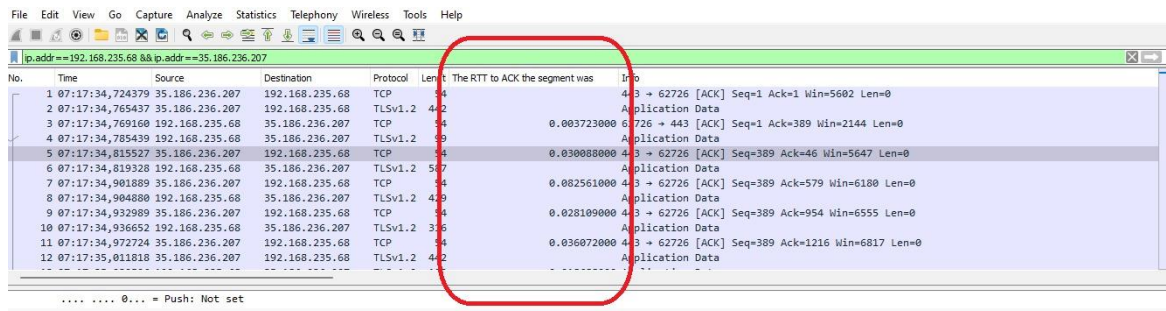
Delay merupakan periode tambahan yang diperlukan saat mentransfer data dari sumber ke tujuan dan juga merupakan parameter keberhasilan dalam proses transmisi paket data[8]. Penulis melakukan pengujian *delay* pada alat pengering maggot berteknologi *Internet of Things*

(IoT) melibatkan analisis beberapa waktu tunda yang terjadi selama proses pengiriman dan penerimaan data. Metode pengambilan data dilakukan menggunakan Wireshark, dengan koneksi dari laptop yang digunakan sebagai *access point*, ESP8266.

Perhitungan untuk mendapatkan *delay* menggunakan rumus di bawah ini:

$$Delay = \frac{Total\ Delay}{Jumlah\ Data} \tag{5.1}$$

$$Delay = \frac{76,49958}{1683} = 0,045\ s$$



Gambar 5.3 Capture Data Delay Wireshark

Pada pengujian *delay* antara ESP8266 dan Firebase penulis menggunakan perangkat lunak yaitu Wireshark yang akan mencatat setiap data yang masuk ke Firebase selama 2 menit 30 detik dan melihat apakah rata-rata *delay* memiliki nilai yang rendah atau tinggi. Data *delay* yang digunakan adalah rata-rata waktu dari data *Round Trip Time* (RTT) ke segmen *Acknowledgment* (ACK) seperti terlihat pada Gambar 5.3. Hasil dalam pengujian *delay* pada alat pengering maggot berteknologi *Internet of Things* (IoT) memiliki *delay* yang sangat bagus untuk kategori *Internet of Things* (IoT) yaitu 45 ms.

5.2.1.2 Throughput

Throughput adalah jumlah data yang berhasil diproses dalam suatu ukuran tertentu saat mentransmisikan paket[9]. Penulis menguji *throughput* pada pengiriman dan penerimaan data saat proses pengeringan berlangsung, pada pengambilan data *throughput* penulis menggunakan koneksi berupa *hotspot* dari laptop yang berfungsi sebagai *access point*.

$$Throughput = \frac{jumlah\ paket\ yang\ diterima}{lama\ pengiriman\ paket} \tag{5.2}$$

$$\text{Throughput} = \frac{713756}{150,472} \times 8 = 37.947 \text{ bps}$$

Measurement	Captured	Displayed
Packets	3347	3288 (98.2%)
Time span, s	150.472	150.472
Average pps	22.2	21.9
Average packet size, B	213	216
Bytes	713756	709652 (99.4%)
Average bytes/s	4743	4716
Average bits/s	37 k	37 k

Gambar 5. 4 Capture Data Throughput Wireshark

Pada Gambar 5.4 merupakan *capture* data pada Wireshark yang digunakan dalam perhitungan untuk mendapatkan *throughput* dilakukan menggunakan perangkat lunak Wireshark yang mencatat setiap data yang masuk ke Firebase selama 2 menit 30 detik serta menganalisis *throughput* yang dihasilkan rendah atau tinggi. Hasil dari pengujian *throughput* pada alat pengering ini memiliki nilai yaitu 37.947 bps atau 37, 947 Kbps.

5.2.2 Pengujian Sensor Suhu

Sensor suhu *thermocouple* MAX6675 digunakan untuk mengukur suhu pada alat pengering maggot selama proses pengeringan dilakukan. Sensor suhu nantinya akan membaca nilai suhu pada alat pengering dan datanya akan dikirimkan melalui *database*.

a. Cara kerja sensor suhu

Sensor suhu akan mendeteksi suhu pada udara ketika sensor diletakkan pada alat pengering. Sensor suhu berfungsi sebagai acuan jika suhu pengeringan melebihi 75°C maka api akan mati, selanjutnya apabila suhu kurang dari 70°C maka api akan menyala kembali. Sensor suhu yang dipakai adalah *thermocouple* MAX6675.

b. Implementasi sensor suhu ke ESP8266

- Sensor suhu dihubungkan ke ESP8266 pada pin out SCK pada D5, CS pada D6, SO pada D7, VCC pada 3V, dan Ground pada G.
- ESP8266 menerima data dari sensor suhu yang akan dihubungkan dengan Firebase sebagai *realtime database* yang terkoneksi dengan jaringan internet.

c. Tahap pengujian akurasi sensor suhu

Pengujian akurasi sensor suhu dilakukan menggunakan thermometer digital yang sudah ada di pasaran. *Thermocouple* MAX6675 akan dibandingkan dengan

thermometer digital seperti pada Gambar 5.5 di bawah untuk mengetahui tingkat akurasinya sama atau tidak.



Gambar 5. 5 Thermometer Digital

d. Pengujian sensor suhu

Akurasi yang optimal terjadi saat nilainya hanya memiliki selisih sebesar maksimal $2,2^{\circ}\text{C}$ [10]. Pengujian kali ini dilakukan dengan memanaskan sensor suhu dengan korek api pada jarak yang sama, pengujian dilakukan sebanyak 5 kali dan pengambilan data dilakukan sebanyak 5 kali dari setiap pengujiannya. Tujuannya adalah untuk mengetahui apakah sensor suhu *thermocouple* MAX6675 dapat menghasilkan data yang akurat. Jika sensor suhu *thermocouple* MAX6675 memberikan tingkat akurasi yang tinggi maka alat tersebut dapat diimplementasikan pada alat pengering maggot ini. Namun, jika akurasi rendah maka bisa diperkirakan terdapat komponen yang rusak atau kesalahan pada *source code*.

Tabel 5. 1 Pengujian Sensor Suhu

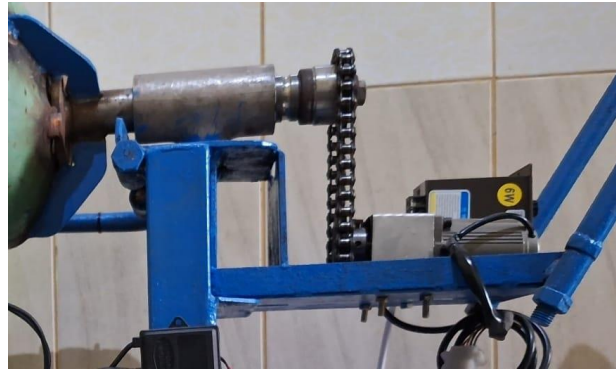
Pengujian Ke-	Detik Ke-	Sensor Suhu	Thermometer	Selisih ($^{\circ}\text{C}$)	Akurasi (%)
1	2	28,25	29,00	0,75 $^{\circ}\text{C}$	97 %
	4	30,70	30	0,70 $^{\circ}\text{C}$	98 %
	6	35,00	34,75	0,25 $^{\circ}\text{C}$	99 %
	8	34,50	34,25	0,25 $^{\circ}\text{C}$	99 %
	10	34,00	33,75	0,25 $^{\circ}\text{C}$	99 %
2	2	35,25	36,20	0,95 $^{\circ}\text{C}$	97 %
	4	38,20	37,75	0,45 $^{\circ}\text{C}$	99 %

	6	40,75	40,51	0,24°C	99 %
	8	39,50	40,00	0,50°C	98 %
	10	38,25	37,90	0,35°C	99 %
3	2	30	29,50	0,50°C	98 %
	4	31,28	30,72	0,56°C	98 %
	6	34,80	34,58	0,22°C	98 %
	8	35,51	35,38	0,13°C	99 %
	10	36,60	36,37	0,27°C	99 %
4	10	35,23	34,92	0,31°C	99 %
	20	46,53	46,28	0,25°C	99 %
	30	55,75	54,52	1,23°C	97 %
	40	69,46	68,20	1,26°C	98 %
	50	76,68	75,85	0,83°C	99 %
5	10	35,61	35,10	0,51°C	98 %
	20	47	46,72	0,28°C	99 %
	30	55,80	55,13	0,67°C	98 %
	40	70,25	69,81	0,44°C	99 %
	50	77,93	77,28	0,65°C	99 %
Rata-rata				0,51°C	98 %

Hasil perbandingan sensor suhu *Thermocouple* MAX6675 dengan thermometer digital menunjukkan bahwa sensor tersebut memiliki akurasi yang tinggi, dengan selisih nilai suhu dari setiap percobaan tidak lebih dari 2,2°C dan rata-rata akurasi mencapai 98%. Hal ini menunjukkan bahwa sensor suhu *thermocouple* MAX6675 layak digunakan pada alat pengering maggot.

5.2.3 Pengujian Motor Servo

Motor Servo digunakan untuk memutar alat atau tong sangrai supaya panas yang dihasilkan merata selama pengeringan akan optimal dan menghasilkan hasil maggot kering sesuai yang diinginkan. Berikut adalah bentuk dari motor servo yang ditunjukkan pada Gambar 5.6 di bawah.



Gambar 5. 6 Motor Servo

a. Cara kerja motor servo

Motor servo terdiri dari motor AC dan gear yang bekerja setelah mendapatkan tegangan 220 VAC dan mengkonversinya dalam bentuk pulsa untuk menggerakkan dan menentukan arah gerak dari motor servo[11].

b. Implementasi motor servo ke ESP8266

- Motor servo dihubungkan ke arus listrik melalui *relay* pada channel ke 3. *Relay* channel 3 dihubungkan ke ESP8266 pada pin D4.
- ESP8266 menerima data dari Firebase pada *realtime database* yang terkoneksi dengan jaringan internet untuk mengontrol *relay* sehingga motor servo dapat menyala.

c. Tahap pengujian kestabilan motor servo

Pengujian kestabilan motor servo dilakukan menggunakan tachometer seperti terlihat pada Gambar 5.7 di bawah. Tachometer akan mengukur kecepatan dari motor servo selama durasi tertentu untuk mengetahui apakah kecepatan motor servo stabil atau tidak.



Gambar 5. 7 Tachometer

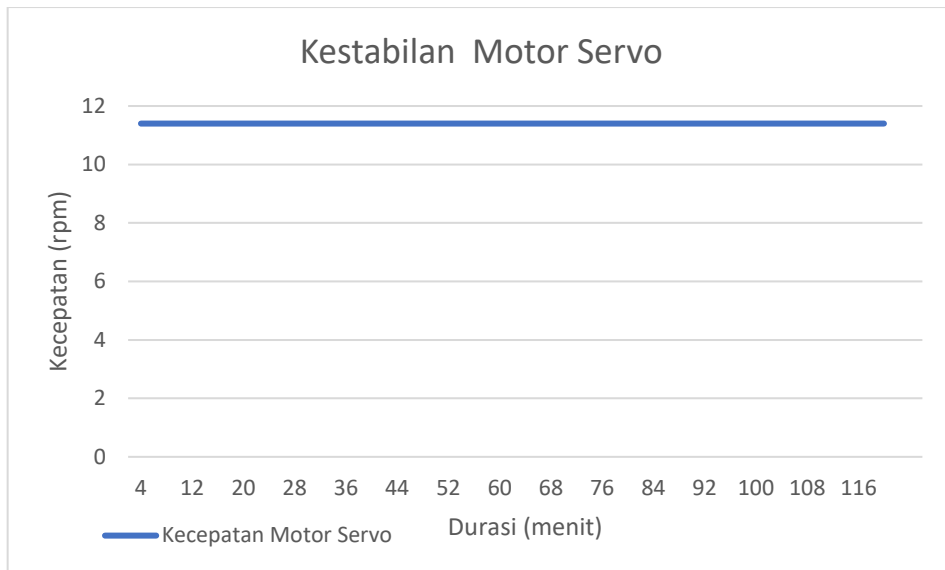
d. Pengujian kestabilan motor servo

Pengujian kestabilan motor servo bertujuan untuk mengetahui rpm pada tong pengering yang diputar dengan motor servo. Mengetahui rpm pada putaran tong bertujuan untuk mengetahui apakah motor servo tetap stabil berputar selama proses pengeringan dan layak untuk diimplementasikan.

Tabel 5. 2 Pengujian Kestabilan Motor Servo

Percobaan ke-	Durasi	Kecepatan
1	4 menit	11,4 rpm
2	8 menit	11,4 rpm
3	12 menit	11,4 rpm
4	16 menit	11,4 rpm
5	20 menit	11,4 rpm
6	24 menit	11,4 rpm
7	28 menit	11,4 rpm
8	32 menit	11,4 rpm
9	36 menit	11,4 rpm
10	40 menit	11,4 rpm
11	44 menit	11,4 rpm
12	48 menit	11,4 rpm

13	52 menit	11,4 rpm
14	56 menit	11,4 rpm
15	60 menit	11,4 rpm
16	64 menit	11,4 rpm
17	68 menit	11,4 rpm
18	72 menit	11,4 rpm
19	76 menit	11,4 rpm
20	80 menit	11,4 rpm
21	84 menit	11,4 rpm
22	88 menit	11,4 rpm
23	92 menit	11,4 rpm
24	96 menit	11,4 rpm
25	100 menit	11,4 rpm
26	104 menit	11,4 rpm
27	108 menit	11,4 rpm
28	112 menit	11,4 rpm
29	116 menit	11,4 rpm
30	120 menit	11,4 rpm



Gambar 5. 8 Chart Kestabilan Motor Servo

Pengujian kestabilan motor servo dilakukan dengan cara diambil sampel sebanyak 30 kali percobaan dengan rentang waktu setiap 4 menit sekali sampai 120 menit. Seperti terlihat pada Gambar 5.8, tong berputar dengan kecepatan yang stabil yaitu 11,4 rpm selama 30 kali percobaan yang dilakukan. Selama pengujian yang dilakukan, tidak ada perlambatan atau kesalahan pada putaran tong yang digerakan motor servo.

5.2.4 Pengujian Pemantik Otomatis

Pemantik pada alat pengeringan maggot ini digunakan untuk menyalakan api ketika adanya aliran gas berdasarkan perintah dari ESP8266. Pemantik otomatis memiliki peran yang sangat penting pada proses pengeringan. Pengujian pemantik otomatis dilakukan untuk menghindari adanya kesalahan ketika proses pengeringan.



Gambar 5. 9 Pemantik Otomatis

a. Cara kerja pemantik

Pemantik yang diberikan daya oleh baterai dapat menyala melalui aliran listrik buka tutup melalui logika dari ESP8266. Pada gambar 5. 9 terlihat bahwa pemantik diletakkan secara berdampingan.

b. Implementasi pemantik

- Pemantik dihubungkan ke baterai melalui *relay* pada channel ke 2. *Relay* channel 2 dihubungkan ke ESP8266 pada pin D3.
- ESP8266 menerima data dari Firebase pada *realtime database* yang terkoneksi dengan jaringan internet untuk mengontrol *relay* sehingga pemantik dapat menyala

c. Pengujian pemantik otomatis

Pengujian pemantik otomatis dilakukan untuk mencegah adanya kegagalan ketika proses pengeringan berlangsung, pengujian yang dilakukan sebanyak 10 kali. Pemantik diletakan 1 cm dari sumbu kompor, pemantik dinyalakan ketika gas mengalir untuk membuktikan apakah pemantik dapat menyalakan api pada kompor.

Tabel 5. 3 Pengujian Pemantik Otomatis

Percobaan Ke-	Durasi	Berfungsi	
		Ya	Tidak
1	3 detik	✓	
2	3 detik	✓	
3	3 detik	✓	
4	3 detik	✓	
5	3 detik	✓	
6	3 detik	✓	
7	3 detik	✓	
8	3 detik	✓	
9	3 detik	✓	

10	3 detik	✓	
11	3 detik	✓	
12	3 detik	✓	
13	3 detik	✓	
14	3 detik	✓	
15	3 detik	✓	
16	3 detik	✓	
17	3 detik	✓	
18	3 detik	✓	
19	3 detik	✓	
20	3 detik	✓	
21	3 detik	✓	
22	3 detik	✓	
23	3 detik	✓	
24	3 detik	✓	
25	3 detik	✓	
26	3 detik	✓	
27	3 detik	✓	
28	3 detik	✓	
29	3 detik	✓	
30	3 detik	✓	

Berdasarkan pengujian yang dilakukan pada pemantik sebanyak 30 kali, durasi pemantik menyala yaitu selama 3 detik untuk menyalakan api ketika gas dialirkan. Hal ini membuktikan bahwa pemantik dengan jarak 1,5 cm dari sumbu api layak untuk diimplementasikan pada alat pengering maggot.

5.2.5 Pengujian *Valve Solenoid*

Valve solenoid merupakan katup yang diaktifkan oleh energi listrik dan mempunyai kumparan yang berfungsi menggerakkan piston yang bergerak sesuai dengan arus AC. *Valve solenoid* memiliki lubang masuk untuk mengarahkan gas ke dalamnya dan lubang keluaran untuk mengeluarkan gas saat piston bergerak selama *valve solenoid* bekerja[12]. Berikut adalah bentuk dari *valve solenoid* yang ditunjukkan pada Gambar 5.10 dibawah.



Gambar 5. 10 Valve Solenoid

a. Cara kerja *valve solenoid*

Cara kerja dari *valve solenoid* adalah menggunakan koil listrik sebagai penggerakannya. Saat koil ini mendapat suplai tegangan, koil tersebut akan menghasilkan medan magnet yang menggerakkan piston di dalam katup. Ketika piston berpindah posisi, gas dari suplai akan keluar melalui lubang keluaran *valve solenoid*.

b. Implementasi *valve solenoid*

- *Valve solenoid* dihubungkan ke arus listrik melalui *relay* pada channel ke 1. *Relay* channel 1 dihubungkan ke ESP8266 pada pin D0.
- ESP8266 menerima data dari Firebase pada *realtime database* yang terkoneksi dengan jaringan internet untuk mengontrol *relay* sehingga *Valve solenoid* dapat terbuka.

c. Pengujian *valve solenoid*

Pengujian *valve solenoid* dilakukan untuk memastikan bahwa *valve solenoid* layak diimplementasikan dan mencegah adanya kegagalan ketika proses

pengeringan berlangsung, Pengujian yang dilakukan sebanyak 10 kali dan diletakan 1 cm dari sumbu kompor, pengujian *valve solenoid* berfungsi ketika gas mengalir untuk membuktikan apakah pemantik dapat menyalakan api pada kompor.

Tabel 5. 4 Pengujian *Valve solenoid*

Percobaan Ke-	Gas Mengalir	
	Ya	Tidak
1	✓	
2	✓	
3	✓	
4	✓	
5	✓	
6	✓	
7	✓	
8	✓	
9	✓	
10	✓	

Hasil dari pengujian *valve solenoid* yang dilakukan 10 kali menghasilkan bahwa *valve solenoid* dapat membuka dan menutup katup sesuai perintah yang diinginkan dan gas dapat mengalir. Dari 10 kali percobaan, tidak ada kesalahan pada *valve solenoid* yang mengganggu kinerja selama proses pengeringan.

5.2.6 Pengujian *Relay* 5V DC

Relay 5V DC adalah komponen yang berfungsi untuk memutus dan menghubungkan arus listrik yang dapat dikontrol melalui mikrokontroler dengan dialirkan tegangan sebesar 5V DC. Implementasi *relay* pada alat pengering berguna untuk mengatur aliran pada komponen pendukung lainnya di alat pengering maggot.



Gambar 5. 11 Relay 5V DC

a. Cara kerja *relay* 5V DC

Pada Gambar 5.11 ditunjukkan bahwa *relay* yang digunakan merupakan *relay* 4 *channel*. *Relay* 5V DC bekerja dengan menggunakan tegangan 5V yang diterapkan pada coil sehingga menciptakan medan magnet. jika komponen tersebut diberi tegangan 5V maka arus akan mengalir pada coil tersebut dan menyebabkan kontak pada *relay* tersebut bekerja.

b. Implementasi *relay* 5V DC

Relay 5V DC dihubungkan ke ESP8266, input 1 dihubungkan ke pin D0, input 2 dihubungkan ke pin D3, input 3 dihubungkan ke D4, VCC dihubungkan ke pin VV dan GND dihubungkan pin G.

c. Pengujian *relay* 5V DC

Pengujian pada komponen *relay* dilakukan untuk menguji *relay* dapat membuka dan menutup aliran listrik sesuai dengan perintah ES8266. Pengujian *relay* dilakukan sebanyak 5 kali tiap channelnya dan dengan tujuan mengetahui *relay* dapat berfungsi berdasarakan perintah.

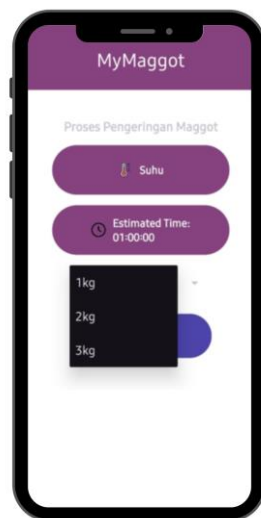
Tabel 5. 5 Pengujian Relay 5V DC

Percobaan ke-	Channel 1		Channel 2		Channel 3	
	Berfungsi	Tidak	Berfungsi	Tidak	Berfungsi	Tidak
1	✓		✓		✓	
2	✓		✓		✓	
3	✓		✓		✓	
4	✓		✓		✓	
5	✓		✓		✓	

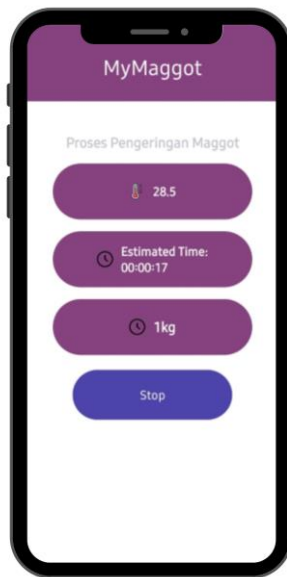
Pengujian *relay* yang dilakukan sebanyak 5 kali dari masing-masing channel menunjukkan bahwa *relay* dapat mengalirkan dan menutup aliran listrik sesuai dengan perintah ESP8266 yang di kendalikan lewat aplikasi. Pada pengujian yang dilakukan sebanyak 5 kali tiap channelnya tidak ada kesalahan ketika *relay* diberikan perintah melalui ESP8266.

5.2.7 Pengujian Usabilitas Aplikasi

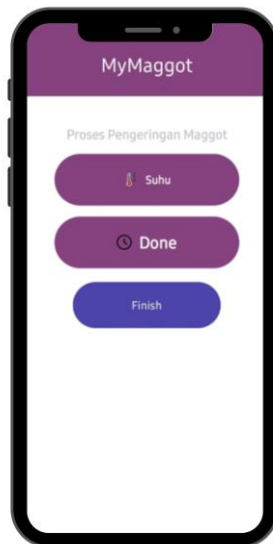
Alat pengering maggot dengan teknologi *Internet of Things* (IoT) tentunya menggunakan perangkat lunak untuk *monitoring* dan mengoperasikan alat pengering. Aplikasi “MyMaggot” telah berhasil menampilkan suhu untuk *dimonitoring* secara *realtime* dan juga fiturnya dapat memulai dan memberhentikan proses pengeringan.



Gambar 5. 12 Menu Awal



Gambar 5. 13 Prose Pengeringan



Gambar 5. 14 Pengeringan Selesai

Pada ketiga *page* berikut adalah berupa tahapan dari proses pembakaran, pada Gambar 5.12 yang merupakan tampilan menu awal terdapat pilihan berat untuk proses pengeringan yang akan dilakukan. Untuk Gambar 5.13 merupakan *page* ketika proses pengeringan berlangsung, di *page* berikut terdapat tampilan bahwa suhu akan berubah seiring dengan berjalannya proses berlangsung, dan juga ada waktu mundur berapa lama lagi pengeringan akan selesai. Selanjutnya pada Gambar 5.14 merupakan *page* terakhir adalah tampilan ketika pengeringan sudah selesai dan ada tombol *finish* untuk kembali ke halaman utama.



Gambar 5. 15 Sesi Wawancara

Setelah melakukan pengujian secara keseluruhan, penulis melakukan wawancara langsung dengan penanggung jawab produksi maggot yang bekerja di greenhouse Telkom university mengenai tampilan dan fungsionalitas aplikasi pada alat pengering maggot seperti terlihat pada gambar 5.15 saat sedang melakukan pengujian langsung. Menurut beliau selaku penanggung jawab produksi maggot, antar muka pengguna sudah baik secara keseluruhan dan mudah diakses oleh para petambak lainnya. Tampilan aplikasi yang sederhana memudahkan para petambak yang awam terhadap teknologi dalam mengoperasikan aplikasi dan *memonitoring* selama proses pengeringan.

5.2.8 Pengujian Hasil Maggot Kering

Pengeringan maggot merupakan proses penting untuk menghasilkan maggot kering yang berkualitas, terutama untuk digunakan sebagai pakan ternak.



Gambar 5. 16 Maggot Gosong



Gambar 5. 17 Maggot Optimal



Gambar 5. 18 Maggot Belum Matang

Pada Gambar 5.17 merupakan hasil maggot kering yang optimal memiliki ciri-ciri yaitu warna yang kuning keemasan, kadar air rendah yang ditandai dengan tekstur yang renyah. Untuk hasil yang tidak optimal terdapat dua kemungkinan yaitu maggot belum matang dan gosong. Pada Gambar 5.18 merupakan maggot yang belum matang ditandai dengan ciri-ciri warna maggot masih cenderung putih dengan tekstur yang masih kenyal dan pada Gambar 5.16 merupakan maggot yang gosong warnanya hitam dan saat dipegang mudah hancur.

Pengujian dilakukan dengan melakukan dengan berat maggot yang berbeda yaitu 1 kg, 2 kg dan 3 kg. Hal ini dilakukan untuk mendapatkan waktu yang ideal agar menghasilkan maggot kering yang optimal untuk setiap berat yang berbeda.

Tabel 5. 6 Pengujian Hasil Maggot Kering

Percobaan Ke-	Berat	Durasi	Hasil
1	1 kg	50 menit	Tidak optimal
2	1 kg	60 menit	Tidak optimal
3	1 kg	1 jam 10 menit	Tidak optimal
4	1 kg	1 jam 30 menit	Optimal
5	1 kg	1 jam 40 menit	Tidak optimal
1	2 kg	1 jam 35 menit	Tidak optimal
2	2 kg	1 jam 40 menit	Tidak optimal
3	2 kg	1 jam 45 menit	Tidak optimal
4	2 kg	1 jam 50 menit	Optimal
5	2 kg	2 jam	Tidak optimal
1	3 kg	2 jam	Tidak optimal
2	3 kg	2 jam 5 menit	Tidak optimal
3	3 kg	2 jam 10 menit	Tidak optimal
4	3 kg	2 jam 15 menit	Tidak optimal
5	3 kg	2 jam 20 menit	Optimal

Dari pengujian yang kita lakukan didapatkan hasil yang optimal dari setiap kilogramnya dengan waktu yang berbeda. Didapatkan waktu optimal selama 1 jam 30 menit untuk berat 1 kilogram, 1 jam 50 menit untuk berat 2 kilogram dan 2 jam 10 menit untuk berat 3 kilogram.



Gambar 5. 19 Penyusutan Berat Maggot

Pada Gambar 5.19 merupakan penyusutan berat maggot sebelum dikeringkan dan setelah dikeringkan seperti terlihat pada gambar berat sebelum pengeringan adalah 2009 gram dan setelah pengeringan berat maggot menjadi 749 gram, maka penyusutan berat maggot berkisar sekitar 62,6 %.



Gambar 5. 20 Survey Kualitas Maggot

Pada Gambar 5.20 penulis melakukan *survey* untuk menanyakan kualitas maggot kepada penjual pakan ikan dan unggas untuk menanyakan kelayakan maggot yang dikeringkan untuk dijual. *Survey* dilakukan dengan menanyakan sebanyak 10 penjual pakan ikan dan unggas.

5.3 Analisis Hasil Pengujian

5.3.1 Analisa Hasil Pengujian Keandalan Koneksi

Pengujian keandalan koneksi yang berupa *Quality of Service* (QoS) diperoleh dari nilai *delay* dan *throughput*. Nilai *delay* dikategorikan sangat bagus apabila memiliki nilai kurang dari 150 ms, sedangkan nilai *throughput* dikategorikan bagus apabila memiliki nilai lebih dari 100 Kbps dan dikategorikan sedang apabila memiliki nilai lebih dari 25 Kbps[13]. *Delay* yang didapat adalah 45 ms dan *throughput* sebesar 37, 947 Kbps. Hasil pengujian keandalan koneksi menunjukkan kinerja yang cukup baik dengan *delay* sebesar 45 ms bisa dikategorikan sangat bagus dan didapat *throughput* sebesar 37, 947 Kbps dapat dikategorikan sedang, tetapi untuk aplikasi monitoring dan pengontrolan *Internet of Things* (IoT), *throughput* ini sudah memadai. Dapat disimpulkan bahwa dengan koneksi yang baik dan stabil maka alat pengering dapat bekerja dengan optimal tanpa adanya gangguan.

5.3.2 Analisa Hasil Pengujian Sensor Suhu

Pada penelitian ini sensor suhu yang digunakan adalah *thermocouple* MAX6675. Sensor suhu tersebut dibandingkan dengan thermometer yang beredar di pasaran untuk memperoleh nilai akurasi. Data yang terkumpul kemudian dianalisis dan digunakan untuk menyesuaikan

pembacaan sensor suhu agar lebih akurat. Hasil yang didapat dari pengujian tersebut adalah sensor suhu *thermocouple* MAX6675 mendapatkan akurasi yang tinggi dengan selisih suhu kurang dari 2,2°C pada setiap percobaannya dan mendapatkan tingkat akurasi sebesar 98% sehingga layak untuk diimplementasikan pada alat pengering maggot. Sensor suhu berfungsi sebagai acuan jika suhu pengeringan melebihi 75°C maka api akan mati, selanjutnya apabila suhu kurang dari 70°C maka api akan menyala kembali. Sensor suhu yang dipakai adalah *thermocouple* MAX6675.

5.3.3 Analisa Hasil Pengujian Motor Servo

Setelah pengujian sebanyak 30 kali, motor servo yang bertugas menggerakkan alat pengering maggot telah berhasil mempertahankan kinerjanya dalam kondisi yang berbeda. Parameter kecepatan motor servo didapatkan dari hasil wawancara dan diukur menggunakan alat ukur yaitu tachometer. Dalam 30 kali pengujian dengan total waktu selama 120 menit, motor servo mampu menjalankan tugasnya secara konsisten dan mempertahankan kecepatan pada 11,4 rpm dengan berat maggot maksimal 3 kg. Selain itu, selama pengujian motor servo juga dapat dioperasikan melalui aplikasi monitoring tanpa kendala. Dari hasil ini, dapat disimpulkan bahwa motor servo menunjukkan tingkat keandalan yang tinggi dan dapat diandalkan selama periode pengujian yang dilakukan.

5.3.4 Analisa Hasil Pengujian Pemantik Otomatis dan *Valve solenoid*

Pengujian pemantik otomatis dan *valve solenoid* pada proses pengeringan maggot memiliki peran yang saling mendukung, yaitu menyalakan api untuk proses pengeringan. Katup pada valve dibuka lebih dahulu untuk mengalirkan gas dan kemudian pemantik dinyalakan untuk menyalakan api. Hasil pengujian yang dilakukan pada kedua komponen tersebut dilakukan secara bersamaan, dengan 10 kali percobaan, *valve solenoid* dan pemantik otomatis nyala secara normal dengan waktu yang sudah dibuat sesuai yang diinginkan melalui ESP8266.

5.3.5 Analisa Hasil Pengujian *Relay* 5V

Pengujian *Relay* 5V yang telah dilakukan, dapat disimpulkan bahwa *relay* yang dikendalikan melalui ESP8266 berhasil mengalirkan dan menutup aliran listrik sesuai dengan perintah yang diberikan. Pengujian telah dilakukan sebanyak 5 kali untuk setiap channel, dan hasilnya menunjukkan bahwa *relay* bekerja secara konsisten dan dapat diandalkan dalam mengendalikan aliran listrik. Pada pengujian *relay* tidak terdapat kesalahan saat *relay* menerima perintah dari ESP8266 melalui aplikasi. Ini menunjukkan bahwa respon *relay* terhadap perintah kontrol sangat baik dan tepat waktu.

5.3.6 Analisa Hasil Pengujian Usabilitas Aplikasi

Pengujian usabilitas aplikasi dilakukan dengan melakukan wawancara bersama penanggung jawab produksi maggot yang berada di Green House Telkom University yang sudah berpengalaman dalam memproduksi maggot kering. Berdasarkan hasil wawancara dan pengujian, berikut analisis yang telah didapatkan:

- Aplikasi mudah digunakan karena memiliki tampilan yang mudah dimengerti dan dioperasikan oleh pengguna yang awam akan teknologi. Fitur pada aplikasi sangat membantu bagi pengguna karena dapat monitoring waktu dan suhu selama proses pengeringan berlangsung.
- Saat proses pengujian, aplikasi memiliki kinerja *realtime* yang handal. Hal ini dibuktikan ketika penguji menggunakan aplikasi dan mengoperasikan secara langsung, aplikasi berhasil menjalankan perintah untuk mengoperasikan alat pengering sesuai dengan fitur yang tersedia dan tanpa ada *delay* yang dapat menghambat kinerja proses pengeringan.

Dengan demikian, berdasarkan hasil pengujian dan wawancara yang telah dilakukan, dapat disimpulkan bahwa aplikasi berhasil memenuhi ekspektasi sebagai media pengoperasian dan monitoring bagi petani maggot. MyMaggot mudah digunakan, data suhu dan waktu pengeringan yang ditampilkan akurat, serta dapat diandalkan bagi petani maggot.

5.3.7 Analisa Pengujian Hasil Maggot Kering

Pengujian ini dilakukan dengan berat 1 kilogram, 2 kilogram dan 3 kilogram untuk mendapatkan hasil yang optimal dari setiap kilonya dengan waktu yang berbeda. Hasil maggot kering dibagi menjadi 2 kategori, yaitu optimal dan tidak optimal. Maggot yang optimal memiliki ciri warna coklat keemasan, kadar air rendah yang ditandai dengan tekstur yang renyah, lalu maggot yang tidak optimal memiliki ciri warna masih cenderung putih dengan tekstur yang masih kenyal atau maggot berwarna hitam gosong dan ketika dipegang hancur. Maggot mengalami penyusutan berat dari sebelum dikeringkan sebesar 62,7 % sampai setelah dikeringkan.

Waktu yang optimal untuk pengeringan maggot berdasarkan percobaan yang dilakukan adalah selama 1 jam 30 menit untuk berat 1 kilogram, 1 jam 50 menit untuk berat 2 kilogram, 2 jam 10 menit untuk berat 3 kilogram. Berdasarkan pengujian tersebut, pengujian hasil maggot kering sudah dapat diimplementasikan sebagai acuan untuk proses pengeringan.

Untuk mengetahui kualitas maggot yang sudah dikeringkan apakah sudah layak untuk dipasarkan dan diberikan ke ikan dan unggas, kami melakukan survey ke beberapa penjual pakan ikan dan unggas. Setelah dilakukan survey, mayoritas dari penjual menyatakan bahwa maggot kering yang dihasilkan memiliki kualitas yang bagus dan layak untuk dipasarkan.

5.4 Kesimpulan

Penelitian kali ini menunjukkan bahwa solusi untuk *monitoring* dan mengontrol alat pengering maggot sudah baik. Sensor dan komponen memiliki akurasi dan fungsi yang memuaskan. Sensor suhu mampu bertahan pada suhu tinggi dan tetap mengirim data ke *database* dengan baik. Pengujian komponen seperti pemantik dan *valve solenoid* juga menunjukkan hasil yang baik, yang dapat dibuktikan dengan kompor yang menyala dengan sempurna dan tanpa kebocoran gas. Motor servo juga berputar dengan stabil selama durasi pengeringan. Keandalan koneksi yang diuji menghasilkan QoS yang cukup baik sehingga alat pengering dapat bekerja secara optimal. Pada pengujian usability, hasil yang didapatkan menunjukkan bahwa fitur dan tampilan pada aplikasi sangat mudah dipahami dan dioperasikan oleh pengguna.

Untuk pengembangan lanjutan, narasumber memberi evaluasi berupa dapat ditambahkan sensor gas yang digunakan untuk *memonitoring* tingkat kapasitas gas yang masih tersedia, sehingga pengguna dapat mengetahui ketika gas sudah habis saat proses pengeringan. Selain itu, dapat pula dikembangkan sistem identitas khusus pada alat, sehingga alat hanya bisa dioperasikan oleh aplikasi yang terotorisasi pada perangkat tertentu. Hal ini akan meningkatkan keamanan dan mencegah alat dioperasikan oleh aplikasi di perangkat lain yang tidak semestinya.

Dengan demikian, penelitian ini tidak hanya menunjukkan hasil yang positif, tetapi juga memberikan rekomendasi yang jelas untuk pengembangan lebih lanjut. Penelitian ini diharapkan dapat memberikan solusi efektif dan bermanfaat bagi peternak maggot, khususnya dalam proses produksi maggot kering. Dengan adanya penelitian ini, peternak dapat mengelola pengeringan maggot dengan lebih efisien, meningkatkan produktivitas, dan menjaga kualitas maggot kering yang dihasilkan, sehingga memberikan nilai tambah yang signifikan bagi usaha mereka.

DAFTAR PUSTAKA

- [1] S. A. M. Toza and D. Yendri, “RANCANG BANGUN SISTEM PENERING MAGGOT BSF SEBAGAI ALTERNATIF PAKAN TERNAK BERBASIS IOT (Internet Of Things),” 2022.
- [2] A. Indah Lestari and Y. Ragil Pamungkas, “Proteksi Motor Induksi Satu Fasa Terhadap Kenaikan Suhu Pada Pengering Maggot Berbasis Panel Surya,” vol. 8, no. 1, 2023, doi: 10.31851/ampere.
- [3] Menteri Pertanian, “Keputusan Menteri Pertanian No.240/Kpts/OT.210/4/2003 tentang Pedoman Cara Pembuatan Pakan yang Baik,” 2003.
- [4] H. Benouda and M. Lachgar, “IoT devices controlled using mobile apps,” 2021. [Online]. Available: <https://www.researchgate.net/publication/353196260>
- [5] H. Ghael, H. Dipak Ghael, L. Solanki, G. Sahu, and A. Professor, “A Review Paper on Raspberry Pi and its Applications,” *International Journal of Advances in Engineering and Management (IJAEM)*, vol. 2, p. 225, 2008, doi: 10.35629/5252-0212225227.
- [6] arduino.biz.id, “I/O Node MCU ESP8266 Lolin,” arduino.biz.id. Accessed: May 16, 2024. [Online]. Available: <https://www.arduino.biz.id/2022/08/io-node-mcu-esp8266-lolin.html?m=1>
- [7] R. Septiana, I. Roihan, and J. A. Karnadi dan Raldi Koestoer, “Calibration of K-Type Thermocouple and MAX6675 Module With Reference DS18B20 Thermistor Based on Arduino DAQ,” 2019.
- [8] F. H. Danufane and M. Di Renzo, “Analysis of the Delay Distribution in Cellular Networks by Using Stochastic Geometry,” *IEEE Open Journal of the Communications Society*, vol. 4, pp. 1728–1744, 2023, doi: 10.1109/OJCOMS.2023.3294791.
- [9] J. Xie and T. Murase, “Effective collaboration to maximize throughput based on multiuser cooperative mobility in social-physical Ad Hoc Networks,” *IEEE Open Journal of the Communications Society*, vol. 2, pp. 818–835, 2021, doi: 10.1109/OJCOMS.2021.3071853.

- [10] A. Setiyoko and D. E. Yuliana, “Kendali Suhu Minyak Goreng Pada Penggorengan Sosis Menggunakan Kontrol PID,” *JASEE Journal of Application and Science on Electrical Engineering*, vol. 3, no. 01, pp. 52–62, Mar. 2022, doi: 10.31328/jasee.v3i01.6.
- [11] S. Fakultas Matematika dan Ilmu Pengetahuan Alam, “PENGENDALI MOTOR SERVO BERBASIS MIKROKONTROLER BASIC STAMP 2SX UNTUK MENGEMBANGKAN SISTEM ROBOTIKA,” 2013.
- [12] A. Wibowo, “RANCANG BANGUN AKTUATOR SOLENOID VALVE PADA PENGENDALIAN PRESSURE REAKTOR OAW (OXYGEN ACETYLENE WELDING) DI BENGKEL LAS DIRAL MENUR SURABAYA,” 2017.
- [13] A. B. Aldiansyah, M. Hakimah, and D. T. Tukadi, “Sistem Monitoring dan Kontrol Rumah Berbasis Internet Of Things (IoT),” 2022.

LAMPIRAN I

Curriculum Vitae 1

PERSONAL INFORMATION

Full Name : Daffa Naufal Putra
Gender : Male
Birth Place and Date : Jakarta, November 26th 2002
Nationality : Indonesia
Religion : Islam
Phone Number : +6281383272674
Email : daffanaufal142@gmail.com



ACADEMIC STATUS

University : Telkom University
Major : Bachelor Degree of Telecommunication Engineering
Semester : 8

EDUCATION

Institutions	City and Province	Year
SMAN 8 Bogor	Bogor, West Java	July 2017 – May 2020
Universitas Telkom	Bandung, West Java	August 2020 – Present

PERSONAL ACHIEVEMENTS

Awards	Year	Description
-	-	-
-	-	-

SUPPORTING ACTIVITIES AND TRAINING

Activities and Trainings	Title	Place
-	-	-

ORGANIZATIONAL EXPERIENCE

Organizations	Title	Period	Descriptions
Grab it Fast FORCY	License and Equipment Staff	2020 - 2022	Activities to spread information about Telkom University which are carried out in the city of Bogor which are members of the Bogor Telkom University Student Community (FORCY)
Graduation & Thanksgiving FORCY	Logistic Staff & Inti Coordinator	2020 - 2022	Activities to celebrate the graduation of Telkom University students who are members of the FORCY
-	-	-	-

WORKING EXPERIENCE

Work	Year	Description
PT. TRANS NEWS CORPORA	2023	Study existing systems and procedures in the company and assist operational activities in the IT department. Perform tasks in activities in the transmission department to assist all operational activities of the IT department.
eternally.gift	2022 - Present	Running an online business focused on selling gift boxes or presents containing printed photos with various models. Eternally.gift has the advantage of photos that can be scanned using an augmented reality video feature.

SKILSS AND HOBBIES

Language Skills : Indonesian (Native), English (Advance).

Computer Skills : C++, Java, Microsoft Office.

Hobbies and interest : Learning about technology and physical exercise.

Others : Creative thinking, Public speaking.

Curriculum Vitae 2

PERSONAL INFORMATION

Full Name : Fadhillah Padmanaba Fauzi
Gender : Male
Birth Place and Date : Pemalang, October 17th 2002
Nationality : Indonesia
Religion : Islam
Phone Number : +6281327325698
Email : padmanaba1710@gmail.com



ACADEMIC STATUS

University : Telkom University
Major : Bachelor Degree of Telecommunication Engineering
Semester : 8

EDUCATION

Institutions	City and Province	Year
SMAN 1 Comal	Pemalang, Central Java	July 2017 – May 2020
Universitas Telkom	Bandung, West Java	August 2020 – Present

PERSONAL ACHIEVEMENTS

Awards	Year	Description
-	-	-
-	-	-

SUPPORTING ACTIVITIES AND TRAINING

Activities and Trainings	Title	Place
-	-	-

ORGANIZATIONAL EXPERIENCE

Organizations	Title	Period	Descriptions
Roadshow KMP Telkom Univerity	Public Relation	2020 - 2023	Activities to spread information about Telkom University which are carried out in the city of Pematang Laha which are members of the Pematang Laha Telkom University Student Community (KMP).
Webinar Campus Expansion KMP Telkom University	Public Relation	2022	Activities to spread information about Telkom University which are carried out in the city of Pematang Laha which are members of the Pematang Laha Telkom University Student Community (KMP).
-	-	-	-

WORKING EXPERIENCE

Work	Year	Description
PT. INFRACOM TELESARANA	2023	Work as Gateway operator at PT. INFRACOM TELESARANA and performed following tasks : Network configuration, network monitoring, troubleshooting, daily check IPSTAR, create report for shift transfer, Sysmon4
-	-	-

SKILSS AND HOBBIES

Language Skills : Indonesian (Native), English (Advance).

Computer Skills : C++, Phyton, Java, Microsoft Office.

Hobbies and interest : Have an interest about telecommunication development.

Others : Creative thinking, Problem solving.

Curriculum Vitae 3

PERSONAL INFORMATION

Full Name : Raymond Ibrahim
Gender : Male
Birth Place and Date : Jakarta, August 25th 2002
Nationality : Indonesia
Religion : Islam
Phone Number : +62895361327154
Email : raymondibrahim12@gmail.com



ACADEMIC STATUS

University : Telkom University
Major : Bachelor Degree of Telecommunication Engineering
Semester : 8

EDUCATION

Institutions	City and Province	Year
SMK Telkom Bandung	Bandung, West Java	July 2017 – May 2020
Universitas Telkom	Bandung, West Java	August 2020 – Present

PERSONAL ACHIEVEMENTS

Awards	Year	Description
-	-	-
-	-	-

SUPPORTING ACTIVITIES AND TRAINING

Activities and Trainings	Title	Place
-	-	-

ORGANIZATIONAL EXPERIENCE

Organizations	Title	Period	Descriptions
-	-	-	-

WORKING EXPERIENCE

Work	Year	Description
PT. BRIN	2023	Study existing LiFi for Vehicle to Vehicle communication.
-	-	-

SKILSS AND HOBBIES

Language Skills : Indonesian (Native), English (Advance)

Computer Skills : C++, Microsoft Office

Hobbies and interest : Swimming and interest about language

Others : Creative thinking, Public speak

Curriculum Vitae 4

PERSONAL INFORMATION

Full Name : Muhammad Rinaldy Fathurrahman

Gender : Male

Birth Place and Date : Jakarta, June 25th 2002

Nationality : Indonesia

Religion : Islam

Phone Number : +6281370601317

Email : aldymuhammad311@gmail.com



ACADEMIC STATUS

University : Telkom University

Major : Bachelor Degree of Telecommunication Engineering

Semester : 8

EDUCATION

Institutions	City and Province	Year
SMAN 31 Jakarta	Jakarta	July 2017 – May 2020
Universitas Telkom	Bandung, West Java	August 2020 – Present

PERSONAL ACHIEVEMENTS

Awards	Year	Description
-	-	-
-	-	-

SUPPORTING ACTIVITIES AND TRAINING

Activities and Trainings	Title	Place
-	-	-

ORGANIZATIONAL EXPERIENCE

Organizations	Title	Period	Descriptions
-	-	-	-

WORKING EXPERIENCE

Work	Year	Description
PT. KAI (Persero)	2023	Study existing systems and procedures in the company and works as an electrical engineer in the telecommunications department.
-	-	-

SKILSS AND HOBBIES

Language Skills : Indonesian (Native), English (Advance)

Computer Skills : C++, Java, Python and Microsoft Office

Hobbies and interest : Playing game and interest about future technology

Others : Creative thinking, Public speak, Cooperation

LAMPIRAN II

No	Komponen	jumlah	harga	total
1	Alat Pengering Maggot	1	Rp 1.500.000,00	Rp 1.500.000,00
2	Kompor Semawar	1	Rp 75.000,00	Rp 75.000,00
3	Selang Gas	2	Rp 15.000,00	Rp 30.000,00
4	<i>Valve solenoid</i>	1	Rp 70.000,00	Rp 70.000,00
5	Pemantik Elektrik	1	Rp 50.000,00	Rp 50.000,00
6	Motor Servo	1	Rp 900.000,00	Rp 900.000,00
7	<i>Relay 4 channel</i>	1	Rp 30.000,00	Rp 30.000,00
8	LCD 2 x 8	1	Rp 35.000,00	Rp 35.000,00
9	<i>Thermocouple</i> MAX 6675	1	Rp 40.000,00	Rp 40.000,00

LAMPIRAN III

Link *source code* google drive:

https://drive.google.com/drive/folders/1JNAP6m-prAT76-jx5_f8b3eiSrQCLmvx?usp=sharing

Source Code Android Studio:

- Source Code Main Activity

```
package com.mymaggot.mymaggot

import android.content.ContentValues.TAG
import android.content.Intent
import android.os.Bundle
import android.os.Handler
import android.util.Log
import android.view.View
import android.widget.AdapterView
import android.widget.Spinner
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import androidx.core.view.ViewCompat
import androidx.core.view.WindowInsetsCompat
import com.google.firebase.database.DataSnapshot
import com.google.firebase.database.DatabaseError
import com.google.firebase.database.FirebaseDatabase
import com.google.firebase.database.ValueEventListener
import com.mymaggot.mymaggot.databinding.ActivityMainBinding
import android.widget.ArrayAdapter
import android.widget.TextView
import com.google.firebase.database.DatabaseReference

class MainActivity : AppCompatActivity() {
    private lateinit var binding: ActivityMainBinding
    private lateinit var databaseReference:
com.google.firebase.database.DatabaseReference
    private lateinit var spinner: Spinner
    private lateinit var estimatedTimeTextView: TextView // Tambahkan
ini
    private var weight1kg: String? = null
    private var weight2kg: String? = null
    private var weight3kg: String? = null

    private lateinit var ref1kg: DatabaseReference
    private lateinit var ref2kg: DatabaseReference
    private lateinit var ref3kg: DatabaseReference

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        binding = ActivityMainBinding.inflate(layoutInflater)
        setContentView(binding.root)
```

```

        // Inisialisasi databaseReference sebelum pemanggilan
        readData() dan startButtonClickListener()
        databaseReference = FirebaseDatabase.getInstance().reference

        // Inisialisasi komponen lainnya
        spinner = findViewById(R.id.dropdown)
        estimatedTimeTextView = findViewById(R.id.estimated_time) //
        Ambil referensi ke TextView yang menampilkan waktu estimasi

        // Setting padding

        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) {
            v, insets ->
                val systemBars =
            insets.getInsets(WindowInsetsCompat.Type.systemBars())
                v.setPadding(systemBars.left, systemBars.top,
            systemBars.right, systemBars.bottom)
                insets
        }

        ref1kg = databaseReference.child("1kg")
        ref2kg = databaseReference.child("2kg")
        ref3kg = databaseReference.child("3kg")

        // Panggil fungsi readData() dan startButtonClickListener()
        setelah databaseReference diinisialisasi
        readData()
        startButtonClickListener()
    }

    private fun readData() {
        databaseReference.addValueEventListener(object :
        ValueEventListener {
            override fun onDataChange(dataSnapshot: DataSnapshot) {
                val weight1 =
            dataSnapshot.child("berat_1").value.toString()
                val weight2 =
            dataSnapshot.child("berat_2").value.toString()
                val weight3 =
            dataSnapshot.child("berat_3").value.toString()

                // Masukkan nilai berat ke dalam dropdown
                val beratList = listOf(weight1, weight2, weight3)
                val adapter = ArrayAdapter(this@MainActivity,
            android.R.layout.simple_spinner_item, beratList)

            adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropd
            own_item)

                spinner.adapter = adapter

                spinner.onItemSelectedListener = object :
            AdapterView.OnItemSelectedListener {
                    override fun onItemSelected(parent:
            AdapterView<*>, view: View?, position: Int, id: Long) {
                        val selectedBerat =
                    parent.getItemAtPosition(position).toString()
                        // Tentukan waktu estimasi berdasarkan nilai
                    berat yang dipilih

                        val estimatedTime = when (selectedBerat) {

```

```

        // UBAH DISINI SESUAI KEINGINAN
        weight1 -> "01:30:00"
        weight2 -> "01:50:00"
        weight3 -> "02:20:00"
        else -> "00:00:00"
    }

    // Tampilkan waktu estimasi pada TextView
    estimatedTimeTextView.text = estimatedTime
}

override fun onNothingSelected(parent:
AdapterView<*>) {
    // Kosongkan implementasi ini jika tidak
    diperlukan
}

}

override fun onCancelled(databaseError: DatabaseError) {
    // Handle any errors that occur during data retrieval
}

})
}

private fun startButtonClickListener() {
    binding.buttonStart.setOnClickListener {
        val selectedBerat = spinner.selectedItem.toString() //
Ambil berat yang dipilih dari dropdown

        // Atur nilai di Firebase sesuai dengan berat yang
dipilih
        when (selectedBerat) {
            "1kg" -> {
                ref1kg.setValue("1")
                Handler(mainLooper).postDelayed({
                    ref1kg.setValue("0")
                }, 7000)
            }
            "2kg" -> {
                ref2kg.setValue("1")
                Handler(mainLooper).postDelayed({
                    ref2kg.setValue("0")
                }, 7000)
            }
            "3kg" -> {
                ref3kg.setValue("1")
                Handler(mainLooper).postDelayed({
                    ref3kg.setValue("0")
                }, 7000)
            }
        }

        val intent = Intent(this@MainActivity,
ProsesActivity::class.java).apply {
            putExtra("selected_weight", selectedBerat) // Kirim
nilai berat yang dipilih ke ProsesActivity
        }
        startActivity(intent)
        finish()
    }
}

```



```

    }
}
}

```

- Source Code Proses Activity

```

package com.mymaggot.mymaggot

import android.app.AlarmManager
import android.app.NotificationChannel
import android.app.NotificationManager
import android.app.PendingIntent
import android.content.ContentValues.TAG
import android.content.Context
import android.content.Intent
import android.media.RingtoneManager
import android.os.Build
import android.os.Bundle
import android.os.CountDownTimer
import android.os.PowerManager
import android.provider.AlarmClock
import android.util.Log
import android.widget.TextView
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import androidx.core.app.NotificationCompat
import androidx.core.app.NotificationManagerCompat
import androidx.core.view.ViewCompat
import androidx.core.view.WindowInsetsCompat
import com.google.firebase.database.DataSnapshot
import com.google.firebase.database.DatabaseError
import com.google.firebase.database.DatabaseReference
import com.google.firebase.database.FirebaseDatabase
import com.google.firebase.database.ValueEventListener
import com.mymaggot.mymaggot.databinding.ActivityMainBinding
import com.mymaggot.mymaggot.databinding.ActivityProsesBinding
import android.os.Handler
import java.util.logging.Logger
import kotlin.math.log

class ProsesActivity : AppCompatActivity() {
    private lateinit var binding: ActivityProsesBinding
    private lateinit var databaseReference:
com.google.firebase.database.DatabaseReference
    private lateinit var countdownTimer: CountDownTimer
    private lateinit var estimatedTimeTextView: TextView
    private lateinit var selectedWeightTextView: TextView

    private lateinit var notificationManager: NotificationManager
    private lateinit var alarmManager: AlarmManager
    private lateinit var wakeLock: PowerManager.WakeLock
    private val NOTIFICATION_CHANNEL_ID =
    "mymaggot_notification_channel"

    private lateinit var ref1kg: DatabaseReference
    private lateinit var ref2kg: DatabaseReference
    private lateinit var ref3kg: DatabaseReference
    private lateinit var refemergency: DatabaseReference

```

```

companion object {

    const val ALARM_INTENT_KEY = "alarmIntentKey"

    fun createAlarmIntent(context: Context): Intent {
        return Intent(AlarmClock.ACTION_DISMISS_ALARM).apply {
            putExtra(AlarmClock.EXTRA_ALARM_SEARCH_MODE,
AlarmClock.ALARM_SEARCH_MODE_ALL)
        }
    }
}

private var isSelesaiActivityStarted = false

override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    enableEdgeToEdge()
    setContentView(R.layout.activity_proses)
    binding = ActivityProsesBinding.inflate(layoutInflater)
    setContentView(binding.root)

    ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) {
v, insets ->
        val systemBars =
insets.getInsets(WindowInsetsCompat.Type.systemBars())
        v.setPadding(systemBars.left, systemBars.top,
systemBars.right, systemBars.bottom)
        insets
    }

    databaseReference = FirebaseDatabase.getInstance().reference
    ref1kg = databaseReference.child("1kg")
    ref2kg = databaseReference.child("2kg")
    ref3kg = databaseReference.child("3kg")
    refemergency = databaseReference.child("emergency")

    val selectedBerat = intent.getStringExtra("selected_weight")
?: ""

    binding.buttonStop.setOnClickListener {
        val intent = Intent(this@ProsesActivity,
SelesaiActivity::class.java)
        startActivity(intent)

        // Atur nilai di Firebase sesuai dengan berat yang
dipilih
        when (selectedBerat) {
            "1kg" -> {
                refemergency.setValue("1")
                Handler(mainLooper).postDelayed({
                    refemergency.setValue("0")
                }, 7000)
            }

            "2kg" -> {
                ref2kg.setValue("0")
                Handler(mainLooper).postDelayed({

```

```

        refemergency.setValue("0")
    }, 7000)
    }

    "3kg" -> {
        ref3kg.setValue("0")
        Handler(mainLooper).postDelayed({
            refemergency.setValue("0")
        }, 7000)
    }
}

}

estimatedTimeTextView = findViewById(R.id.estimated_time)
selectedWeightTextView = binding.weight

databaseReference = FirebaseDatabase.getInstance().reference
readData()

notificationManager =
getSystemService(Context.NOTIFICATION_SERVICE) as NotificationManager
alarmManager = getSystemService(Context.ALARM_SERVICE) as
AlarmManager

val powerManager = getSystemService(Context.POWER_SERVICE) as
PowerManager
wakeLock =
powerManager.newWakeLock(PowerManager.PARTIAL_WAKE_LOCK,
"mymaggot:WakeLock")

createNotificationChannel()
updateUI()
}

private fun createNotificationChannel() {
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
        val channelName = "MyMaggot Notification"
        val channelDescription = "Channel for MyMaggot app
notifications"
        val channel = NotificationChannel(
            NOTIFICATION_CHANNEL_ID,
            channelName,
            NotificationManager.IMPORTANCE_DEFAULT
        ).apply {
            description = channelDescription
        }
        notificationManager.createNotificationChannel(channel)
    }
}

private fun readData() {
    databaseReference.addValueEventListener(object :
ValueEventListener {
        override fun onDataChange(dataSnapshot: DataSnapshot) {
            val temperature =
dataSnapshot.child("Temperature").value.toString()
            binding.suhu.text = temperature
        }

        override fun onCancelled(databaseError: DatabaseError) {

```

```

        // Handle any errors that occur during data retrieval
    }
    })
}

private fun updateUI() {
    // Ambil referensi ke TextView yang menampilkan waktu
    estimasi
    val selectedBerat = intent.getStringExtra("selected_weight")
    val estimatedTime = when (selectedBerat) {
        //UBAH WAKTU DISINI
        "1kg" -> 5400 * 1000L // 1 jam dalam milidetik
        "2kg" -> 6600 * 1000L // 2 jam dalam milidetik
        "3kg" -> 8400 * 1000L // 3 jam dalam milidetik
        else -> 0L // Jika berat tidak cocok, atur waktu menjadi
0
    }
    val weight = when (selectedBerat) {
        "1kg" -> "1 Kg"
        "2kg" -> "2 Kg"
        "3kg" -> "3 Kg"
        else -> "0 Kg"
    }

    selectedWeightTextView.text = selectedBerat

    // Mulai hitung mundur waktu
    countdownTimer = object : CountdownTimer(estimatedTime, 1000)
    {
        override fun onTick(millisUntilFinished: Long) {
            // Konversi waktu yang tersisa menjadi format
            jam:menit:detik
            val hours = (millisUntilFinished / (1000 * 60 * 60))
            % 24
            val minutes = (millisUntilFinished / (1000 * 60)) %
60
            val seconds = (millisUntilFinished / 1000) % 60
            val timeLeftFormatted =
String.format("%02d:%02d:%02d", hours, minutes, seconds)
            // Tampilkan waktu yang tersisa pada TextView
            estimatedTimeTextView.text = timeLeftFormatted
        }

        override fun onFinish() {
            if (!isSelesaiActivityStarted) {
                isSelesaiActivityStarted = true

                // Pindahkan kode untuk mengatur nilai Firebase,
                menampilkan notifikasi, dan membunyikan alarm ke sini
                when (selectedBerat) {
                    "1kg" -> {
                        ref1kg.setValue("0")
                    }

                    "2kg" -> {
                        ref2kg.setValue("0")
                    }

                    "3kg" -> {
                        ref3kg.setValue("0")
                    }
                }
            }
        }
    }
}

```

```

        }
    }

    // Tampilkan notifikasi
    val notificationIntent =
Intent(this@ProsesActivity, MainActivity::class.java)
    val pendingIntent = PendingIntent.getActivity(
        this@ProsesActivity,
        0,
        notificationIntent,
        PendingIntent.FLAG_UPDATE_CURRENT or
PendingIntent.FLAG_MUTABLE
    )

    val notification =
NotificationCompat.Builder(this@ProsesActivity,
NOTIFICATION_CHANNEL_ID)
        .setSmallIcon(R.drawable.ic_notification)
        .setContentTitle("MyMaggot")
        .setContentText("Proses selesai!")
        .setContentIntent(pendingIntent)
        .setAutoCancel(true)
        .build()

    notificationManager.notify(0, notification)

    // Bunyikan alarm
    wakeLock.acquire(10 * 60 * 1000L /*10 minutes*/)
    val alarmSound =
RingtoneManager.getDefaultUri(RingtoneManager.TYPE_ALARM)
    val ringtone =
RingtoneManager.getRingtone(applicationContext, alarmSound)
    ringtone.play()

    // Stop the alarm after a delay (adjust this
delay as needed)
    Handler(mainLooper).postDelayed({
        ringtone.stop()
    }, 7000) // Stop the alarm after 5 seconds

    val intent = Intent(this@ProsesActivity,
SelesaiActivity::class.java)
    startActivity(intent)
    }
}
}.start()
Log.d("TESTER", isSelesaiActivityStarted.toString())
Log.d("TIME", estimatedTimeTextView.toString())
}

override fun onDestroy() {
    super.onDestroy()
    countdownTimer.cancel()
}
}

```

- Source Code Selesai Activity

```

package com.mymaggot.mymaggot

import android.content.Intent
import android.net.Uri
import android.os.Bundle
import android.widget.Button
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import androidx.core.view.ViewCompat
import androidx.core.view.WindowInsetsCompat
import com.google.firebase.database.DatabaseReference
import com.google.firebase.database.FirebaseDatabase
import com.mymaggot.mymaggot.ProsesActivity
import com.mymaggot.mymaggot.databinding.ActivityMainBinding
import com.mymaggot.mymaggot.databinding.ProsesSelesaiBinding

class SelesaiActivity : AppCompatActivity() {
    private lateinit var binding: ProsesSelesaiBinding
    private lateinit var refemergency: DatabaseReference
    private lateinit var databaseReference:
com.google.firebase.database.DatabaseReference

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        binding = ProsesSelesaiBinding.inflate(layoutInflater)
        setContentView(binding.root)

        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v,
insets ->
            val systemBars =
insets.getInsets(WindowInsetsCompat.Type.systemBars())
            v.setPadding(systemBars.left, systemBars.top,
systemBars.right, systemBars.bottom)
            insets
        }

        databaseReference = FirebaseDatabase.getInstance().reference

        refemergency = databaseReference.child("emergency")

        binding.buttonStart.setOnClickListener {
            // Menerima intent dari aktivitas sebelumnya
            val alarmIntent =
intent.getParcelableExtra<Intent>("alarmIntent")
            alarmIntent?.let {
                // Hanya kirim broadcast jika alarmIntent tidak null
                sendBroadcast(it)
            }
            refemergency.setValue("0")

            val intent = Intent(this@SelesaiActivity,
MainActivity::class.java)
            startActivity(intent)
            // Jangan lupa untuk menutup aktivitas saat tombol ditekan

```




```
    finish()  
  }  
}  
}
```


LAMPIRAN IV

Link dokumentasi pengujian alat:

<https://drive.google.com/drive/folders/1StztCyPpwd8ZPL2tCjiFndV-wMLQG129?usp=sharing>

Dokumentasi pengujian alat:

Keterangan	Gambar
Pengujian Sensor Suhu	
Pengujian Motor Servo	
Pengujian <i>valve solenoid</i>	

<p>Pengujian Pemantik</p>	
<p>Pengujian <i>Relay</i></p>	