

LAPORAN AKHIR

**RANCANG BANGUN ALARM DENGAN AKSES SIDIK JARI DAN
APLIKASI WHATSAPP BERBASIS *INTERNET OF THINGS*
MENGUNAKAN ARDUINO UNO R3
DAN NODEMCU ESP8266 V3**



**Disusun Untuk Memenuhi Syarat Menyelesaikan Pendidikan Diploma III
Pada Jurusan Teknik Elektro Program Studi Teknik Telekomunikasi
Politeknik Negeri Sriwijaya**

Oleh :
M Raichan Muzakki
062130331092

POLITEKNIK NEGERI SRIWIJAYA

2024

LEMBAR PENGESAHAN

**RANCANG BANGUN ALARM DENGAN AKSES SIDIK JARI DAN
APLIKASI WHATSAPP BERBASIS INTERNET OF THINGS
MENGUNAKAN ARDUINO UNO R3
DAN NODEMCU ESP8266 V3**



Oleh :

M Raichan Muzakki
062130331093

Menyetujui,

Palembang, Juli 2024

Pembimbing I

Pembimbing II


Hi. I. Indrawati, S.T., M.T.I
NIP. 197105282006042001

Asriyadi, S.T., M.T
NIP. 198404272015041000

Mengetahui,

Ketua Jurusan Teknik Elektro

**Koordinator Prodi DIII
Teknik Telekomunikasi**


Ir. Iskandar Lutfi, M.T.
NIP.196501291991031002


Ciksadan, S.T., M.Kom
NIP.196809071993031003

DAFTAR ISI

HALAMAN SAMPUL	i
HALAMAN PENGESAHAN	ii
MOTTO.....	iii
ABSTRAK.....	iv
ABSTRACT.....	v
KATA PENGANTAR.....	vi
DAFTAR ISI	vii
DAFTAR GAMBAR	x
DAFTAR TABEL	xii
DAFTAR LAMPIRAN.....	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian	3
1.5 Urgensi Penelitian.....	3
1.6 Metode Penelitian.....	4
1.7 Sistematika Penulisan.....	4
BAB II TINJAUAN PUSTAKA	6
2.1 Karya-karya Rujukan.....	6
2.2 Alarm	7
2.2.1 Pengertian dan Fungsi Alarm	7
2.2.2 Alarm Pintu/Jendela.....	7
2.3 <i>Internet of Things</i> (IoT)	8
2.4 Arduino Uno R3 ATmega 328P.....	11
2.4.1 Fitur Arduino Uno	11
2.4.2 Spesifikasi Teknis.....	12

2.4.3 Connector Pinouts	13
2.5 NodeMCU ESP8266 V3	15
2.5.1 Spesifikasi	16
2.5.2 Pinouts	17
2.6 Arduino Integrated Development Environment (IDE)	18
2.7 Fingerprint Scanner AS608	19
2.7.1 Fitur	19
2.7.2 Spesifikasi	18
2.8 LCD I2C 16x2	20
2.9 Reed Switch MC-38	21
2.10 Modul Relay 5V DC	22
2.11 Buzzer	23
2.11.1 Fitur	23
2.12.2 Jenis-jenis Buzzer	23
2.12 ThingESP	25
2.13 Twilio	25
2.14 WhatsApp	26
BAB III RANCANG BANGUN PERANGKAT	28
3.1 Tujuan Perancangan	28
3.2 Diagram Blok	29
3.3 Prinsip Kerja Perangkat	29
3.4 Flowchart Kerja Perangkat	30
3.5 Skema Rangkaian	31
3.6 Perakitan Perangkat	31
3.6.1 Perangkaian Perangkat Keras	32
3.6.2 Coding Program Blok Induk (Arduino Uno)	34
3.6.3 Konfigurasi ThingESP, Twilio, WhatsApp	37
3.6.4 Coding Program Blok Alarm	45
3.6.5 Rangkaian Keseluruhan Perangkat	49
BAB IV PEMBAHASAN	50
4.1 Device Starts	50

4.2 Menambah Pengguna (Akses Kontrol dan Monitor <i>via</i> WhatsApp).....	51
4.3 Kontrol dan Monitor melalui WhatsApp.....	51
4.4 Respon Sensor	52
4.5 Kontrol dan Monitor <i>via</i> WhatsApp saat Terjadi Penerobosan.....	53
4.6 Analisis	53
4.7 Analisis FRR dan FAR.....	56
BAB V PENUTUP	58
5.1 Kesimpulan.....	58
5.2 Saran.....	59
DAFTAR PUSTAKA	
LAMPIRAN	

DAFTAR GAMBAR

Gambar 2.1	Ilustrasi <i>Internet of Things</i> (IoT).....	8
Gambar 2.2	Papan mikrokontroler Arduino Uno R3	11
Gambar 2.3	<i>Pinouts</i> Arduino Uno R3	15
Gambar 2.4	<i>Controller board</i> NodeMCU ESP8266 V3	15
Gambar 2.5	<i>Pinouts</i> NodeMCU ESP8266 V3	17
Gambar 2.6	Antarmuka <i>software</i> Arduino IDE	18
Gambar 2.7	Modul pemindai sidik jari AS608	19
Gambar 2.8	LCD I2C 16x2.....	20
Gambar 2.9	Sensor <i>reed</i> MC-38	21
Gambar 2.10	Modul <i>relay</i> 5V	22
Gambar 2.11	<i>Buzzer</i> aktif SFM-27	24
Gambar 2.12	<i>Buzzer</i> pasif 5V	24
Gambar 2.13	Tangkapan layar laman beranda Twilio	25
Gambar 2.14	<i>Icon</i> WhatsApp	26
Gambar 3.1	Diagram blok perangkat alarm.....	29
Gambar 3.2	<i>Flowchart</i> perangkat alarm	30
Gambar 3.3	Skema rangkaian	31
Gambar 3.4	Rangkaian awal perangkat alarm	33
Gambar 3.5	Nama <i>sketch</i>	34
Gambar 3.6	<i>Header</i> dan deklarasi variabel.....	35
Gambar 3.7	<i>Void setup</i>	35
Gambar 3.8	<i>Void loop</i>	36
Gambar 3.9	Kotak verifikasi <i>username</i> dan <i>password</i> ThingESP	37
Gambar 3.10	Beranda ThingESP	37
Gambar 3.11	Salinan nama proyek dan kredensial perangkat	38
Gambar 3.12	Nama proyek dan kredensial perangkat tugas Laporan Akhir ...	39
Gambar 3.13	<i>Endpoint</i> URL perangkat alarm <i>project</i> ThingESP.....	39

Gambar 3.14 Memasukkan kode otentifikasi ke kotak verifikasi.....	40
Gambar 3.15 Akun baru konsol Twilio	41
Gambar 3.16 Verifikasi nomor telepon yang juga nomor WhatsApp	41
Gambar 3.17 Kode verifikasi nomor telepon.....	42
Gambar 3.18 Memilih aplikasi perpesanan yang akan digunakan.....	42
Gambar 3.19 Pengaturan model perpesanan.....	43
Gambar 3.20 Mengaktifkan Twilio Sandbox.....	44
Gambar 3.21 Kode <i>request</i> dan notifikasi nomor WA telah dapat menggunakan layanan Sandbox.....	44
Gambar 3.22 Konfigurasi Sandbox.....	45
Gambar 3.23 Perubahan deklarasi variabel.....	46
Gambar 3.24 Perubahan <i>void setup</i>	46
Gambar 3.25 Perubahan <i>query</i> yang akan dikirim dan diterima melalui WhatsApp	47
Gambar 3.26 Penambahan <i>void loop</i>	48
Gambar 3.27 Uji coba dengan <i>query</i> “status”	48
Gambar 3.28 Rangkaian keseluruhan perangkat alarm	49

DAFTAR TABEL

Tabel 2.1 Spesifikasi teknis <i>controller board</i> Arduino Uno R3	12
Tabel 2.2 <i>Pinout</i> analog papan kendali Arduino Uno R3	13
Tabel 2.3 <i>Pinout</i> digital papan kendali Arduino Uno R3.....	14
Tabel 2.4 Nomor dan nama <i>pinouts</i> NodeMCU ESP8266 V3.....	17
Tabel 4.1 Kondisi perangkat setelah dinyalakan.....	50
Tabel 4.2 <i>Request</i> layanan Twilio Sandbox	51
Tabel 4.3 Kendali dan pemantauan melalui pesan WhatsApp.....	51
Tabel 4.4 Respon sensor	52
Tabel 4.5 Pengiriman perintah dan pemantauan saat terjadi penerobosan.....	53

BAB I

PENDAHULUAN

1.1 Latar Belakang

Alat yang berfungsi memberikan tanda peringatan ketika terjadi keadaan membahayakan, yang kemudian dikenal sebagai alarm, bukan merupakan salah satu alat baru dalam seperangkat sistem keamanan yang kerap digunakan. Seiring dengan perkembangan ilmu pengetahuan, perangkat terapan berupa teknologi alarm turut memperoleh berbagai inovasi guna memberikan nilai ekspektasi maksimal yang diharapkan oleh pengguna.

Ketangguhan dalam pembatasan pengguna dan kemudahan dalam pemakaian menjadi fitur penting dalam optimasi pemanfaatan alarm sebagai salah satu perangkat sistem keamanan. Selain itu, pencapaian dalam teknologi informasi dan komunikasi menjadi salah satu variabel yang memberikan nilai lebih dalam sebuah perangkat alarm.

Internet of Things (IoT) memberikan konsep lingkungan kerja yang memberikan kemudahan bagi pengguna untuk melakukan kendali terhadap berbagai macam perangkat elektronik. Sistem keamanan secara keseluruhan juga ikut tersentuh oleh konsep tersebut. Berbagai inovasi baru berbasis IoT bermunculan, dengan kekhasan dari masing-masing inovator.

Perangkat sistem keamanan tidak ketinggalan merasakan berbagai inovasi berbasis IoT. Prosedur kendali, pemantauan, maupun penerimaan notifikasi atas perangkat dapat dilakukan melalui jaringan internet. Pengguna semakin semakin merasa terjamin akan keamanan aset ataupun properti meskipun berada pada tempat yang tidak dapat diamati secara langsung.

Berbagai layanan *cloud* pendukung *platform* IoT memungkinkan akses ke perangkat keamanan, seperti alarm, dapat diperoleh melalui aplikasi yang terpasang pada *gadget* seperti *handphone*. Pilihan terbuka lebar, mulai dari

aplikasi yang cukup rumit, hingga ke aplikasi yang sudah sangat lazim dan mudah digunakan untuk berkomunikasi *via* internet. Salah satu aplikasi yang telah memperoleh popularitas di kalangan pengguna *smartphone* adalah WhatsApp. Aplikasi perpesanan satu ini bagi masyarakat Indonesia seolah menjadi media utama pertukaran informasi. Dengan trend popularitas yang sangat signifikan tersebut, WhatsApp pantas dipertimbangkan sebagai pilihan pertama untuk dijadikan layanan perpesanan dalam pengiriman instruksi ke berbagai perangkat berbasis IoT, tak ketinggalan perangkat alarm.

Dengan latar belakang tersebut, penulis menyusun Laporan Akhir membuat rancang bangun sebuah perangkat sederhana berupa alarm yang diharapkan memiliki keunggulan-keunggulan, seperti tingkat pembatasan dan kemudahan pemakaian yang benar-benar dapat diaplikasikan. Selanjutnya, gagasan untuk membangun perangkat alarm tersebut penulis tuangkan dalam Laporan Akhir “Rancang Bangun Alarm dengan Akses Sidik Jari dan Aplikasi WhatsApp berbasis *Internet of Things* (IoT) menggunakan Arduino UNO R3 dan NodeMCU ESP8266 V3”.

1.2 Rumusan Masalah

1. Bagaimana perakitan perangkat alarm berbasis *Internet of Things* dengan akses sidik jari dan aplikasi WhatsApp menggunakan *board* Arduino Uno R3 dan NodeMCU ESP8266.
2. Bagaimana cara kerja dan kinerja perangkat alarm berbasis IoT dengan akses sidik jari dan aplikasi WhatsApp menggunakan *board* Arduino Uno R3 dan NodeMCU ESP8266.

1.3 Batasan Masalah

Dengan perumusan masalah di atas, maka pembatasan masalah untuk rancang bangun perangkat alarm ini adalah:

1. Sensor sidik jari sebagai perangkat input melalui papan kendali Arduino Uno R3 sebagai akses kendali darurat sistem alarm.

2. Aplikasi WhatsApp sebagai media kendali dan pemantauan sistem alarm melalui jaringan internet.

1.4 Tujuan Penelitian

Laporan Akhir ini ditujukan untuk:

1. Merancang dan membangun perangkat alarm berbasis *Internet of Things* dengan mode kendali dan kontrol melalui WhatsApp dan akses darurat menggunakan sidik jari.
2. Memberikan rancangan perangkat alarm berbasis IoT untuk pengembangan lebih lanjut.

1.5 Urgensi Penelitian

Pemanfaatan WhatsApp memungkinkan kendali dan kontrol atas perangkat alarm berbasis IoT menjadi lebih mudah, tanpa mengharuskan pengguna memasang aplikasi ataupun pengaturan lainnya pada perangkat *smartphone*. Tak jarang, semua anggota dalam sebuah keluarga memiliki akun WhatsApp yang membuat kehadiran perangkat alarm tidak mengurangi keleluasaan setiap anggota keluarga untuk keluar dari ataupun masuk kembali ke dalam rumah.

Sebagai pengembangan dari model perangkat alarm yang telah ada, akses darurat menggunakan sidik jari tersedia jika berhadapan dengan kondisi-kondisi tertentu, seperti terjadi penerobosan ataupun pengguna ingin membuka pintu namun tidak dapat mengakses internet melalui *handphone*. Fitur otomatis dengan jeda waktu beberapa saat setelah perangkat alarm dinyalakan, selain memberikan kesempatan pengguna untuk dapat menutup dan mengunci pintu sebelum meninggalkan rumah, juga menghilangkan kekhawatiran apakah alarm telah dinyalakan atau belum.

1.6 Metode Penulisan

Dalam penyusunan Laporan Akhir ini, beberapa metode yang digunakan yaitu:

1. Studi pustaka

Sebagai rangkaian aktifitas membaca, mencatat dan mengumpulkan bahan penelitian. Studi terutama ditujukan untuk menetapkan dan menelaah landasan teori melalui berbagai sumber literatur, baik yang berhubungan langsung maupun terkait dengan topik penelitian.

2. Metode cyber

Metode siber diterapkan sebagai metode mengumpulkan informasi serta rujukan dengan memanfaatkan berbagai media yang dapat diakses melalui internet. Media-media tersebut dapat berupa konten dokumen, audio, maupun audio- visual dari berbagai sumber.

3. Konsultasi dan diskusi

Konsultasi dan diskusi terutama dilakukan dengan Dosen Pembimbing I dan II. Selain Dosen Pembimbing I dan II, bentuk-bentuk konsultasi dan diskusi juga terbuka untuk dilakukan dengan berbagai pihak yang memiliki kapabilitas sesuai topik penelitian, baik di dalam maupun di luar lingkungan kampus.

4. Observasi

Observasi yang dimaksud lebih ditekankan kepada pengamatan terhadap lingkungan yang akan dijadikan sebagai ruang pelaksanaan uji coba dan pengaplikasian perangkat alarm.

1.7 Sistematika Penulisan

Tugas Laporan Akir ini disusun dengan sistematika sebagai berikut;

BAB I PENDAHULUAN

Berisi uraian latar belakang, rumusan masalah, batasan masalah, tujuan,

manfaat, metode penulisan, dan sistematika penulisan laporan.

BAB II TINJAUAN PUSTAKA

Membahas rujukan-rujukan dan teori-teori tentang pembuatan perangkat alarm dan atau hal-hal lain yang berhubungan dengan objek yang diangkat pada Tugas Akhir.

BAB III RANCANG BANGUN ALAT

Dijelaskan proses pembuatan perangkat alarm dari mulai diagram, *flowchart*, skema rangkaian, dan perakitan perangkat.

BAB IV PEMBAHASAN

Berisi laporan terhadap kinerja alat yang akan disajikan dalam bentuk data dan analisis.

BAB V PENUTUP

Kesimpulan dan saran.

DAFTAR PUSTAKA

BAB II

TINJAUAN PUSTAKA

2.1 Karya-karya Rujukan

Perangkat yang menjadi karya pada Tugas Laporan Akhir ini dibangun dengan merujuk kepada beberapa karya yang telah ada sebelumnya, baik dari aspek perangkat keras maupun pemrograman instruksi yang ditanamkan ke *chip* papan pengendali. Karya-karya tersebut adalah

1. *Pengaman Pintu Rumah Berbasis Sensor Sidik Jari dan Magnetic Sensor* [1], karya Ade Septryanti dan Erlangga Satria Permana yang ditampilkan pada CESS (*Journal of Computer Engineering, System and Science*), tahun 2020.
2. Penggunaan modul sensor sidik jari AS608 yang ditampilkan dalam halaman web berjudul *How To Use The AS608 Fingerprint Sensor with Arduino* dan dapat diakses pada URL <https://srituhobby.com/how-to-use-the-as608-fingerprint-sensor-with-arduino/>, beserta kode program.
3. *Control LED Using WhatsApp* yang ditampilkan pada kanal YouTube <https://www.youtube.com/watch?v=noeT7KGudpM> (2022), dan kode yang diunggah pada situs web <https://github.com/Tech-Trends-Shameer/Esp-8266-Projects/tree/main/Control-LED-Using-WhatsApp>.

Sebagai kebaruan yang penulis aplikasikan pada rancang bangun perangkat Tugas Laporan Akhir, penulis menggabungkan unsur-unsur penting dari tiga karya tersebut di atas ke dalam satu perangkat alarm berbasis IoT. Langkah tersebut tidak lain adalah ikhtiar penulis mencoba melakukan pengembangan atas karya-karya yang bukan hanya telah sangat baik dari sudut perancangan, namun juga karya-karya tersebut pantas untuk dipelajari lebih lanjut.

2.2 Alarm

2.2.1 Pengertian dan Fungsi Alarm

Menurut Kamus Besar Bahasa Indonesia (KBBI), alarm diartikan sebagai tanda bahaya berupa sinyal, bunyi, sinar, dan sebagainya; alat mekanik yang dirancang untuk memperingatkan akan adanya bahaya atau kerusakan [2]. Pengertian tersebut sangat jelas sekali, dan penulis tidak menemukan referensi lain yang mendefinisikan alarm jauh berbeda dari pembatasan menurut KBBI.

Dengan pengertian di atas dipahami bahwa segala yang mungkin untuk digunakan sebagai pemberi tanda atau peringatan akan terjadinya bahaya atau ancaman dapat dikategorikan sebagai alarm. Pengertian ke-dua dari KBBI menunjukkan bahwa “sesuatu” tersebut dapat berupa sebuah alat atau perangkat yang secara sengaja dirancang memenuhi fungsi memberikan peringatan akan kehadiran bahaya.

2.2.2 Alarm Pintu/Jendela

Alarm pintu atau jendela adalah salah satu komponen utama dalam sistem keamanan rumah yang modern. Mereka berfungsi untuk mendeteksi pembukaan atau penutupan pintu secara otomatis. Ketika pintu dibuka tanpa izin, sensor akan memberikan peringatan kepada pemilik rumah atau sistem keamanan untuk mengambil tindakan selanjutnya. Ini memberikan deteksi awal yang penting terhadap upaya perampokan atau kegiatan kriminal lainnya [3].

Seiring perkembangan ilmu pengetahuan dan teknologi, perangkat alarm pintu juga mengalami inovasi, baik dari mekanisme kerja perangkat, cara instalasi, kemudahan dalam pemakaian, dan sebagainya. Produsen-produsen perangkat pengamanan rumah menaruh perhatian bukan hanya pada ketangguhan perangkat, namun juga persaingan harga. Alarm pintu bukan barang langka di pasaran, beragam merk dibanderol dengan harga kompetitif tergantung ekspektasi konsumen.

2.3 *Internet of Things (IoT)*



Gambar 2.1 Ilustrasi *Internet of Things (IoT)* [4]

Internet Of Things atau sering disebut IoT adalah sebuah gagasan dimana semua benda di dunia nyata dapat berkomunikasi satu dengan yang lain sebagai bagian dari satu kesatuan sistem terpadu menggunakan jaringan internet sebagai penghubung. misalnya CCTV yang terpasang di sepanjang jalan dihubungkan dengan koneksi internet dan disatukan di rung kontrol yang jaraknya mungkin puluhan kilometer. atau sebuah rumah cerdas yang dapat dimanage lewat smartphone dengan bantuan koneksi internet. pada dasarnya perangkat IoT terdiri dari sensor sebagai media pengumpul data,sambungan internet sebagai media komuniakasi dan server sebagai pengumpul informasi yang diterima sensor dan untuk analisa [5].

Internet of Things menjadi sebuah bidang penelitian tersendiri semenjak berkembangnya teknologi internet dan media komunikasi lain, semakin berkembang keperluan manusia tentang teknologi, maka semakin banyak penelitian yang akan hadir, *Internet of Things* salah satu hasil pemikiran para peneliti yang

mengoptimasi beberapa alat seperti media sensor, *radio frequency identification* (RFID), *wireless sensor network* serta *smart object* lain yang memungkinkan manusia mudah berinteraksi dengan semua peralatan yang terhubung dengan jaringan internet [6]. Konsep IoT bertujuan untuk membuat internet semakin berkembang dan meluas. Selanjutnya, dengan memungkinkan akses dan interaksi yang mudah dengan beragam perangkat seperti, peralatan rumah tangga, kamera cctv, sensor pemantauan, aktuator, display, kendaraan, dan sebagainya. IoT akan mendorong pengembangan sejumlah aplikasi yang memanfaatkan jumlah dan variasi data yang berpotensi besar yang dihasilkan oleh objek tersebut untuk memberikan layanan baru kepada warga negara, perusahaan, dan administrasi publik [7].

Ide awal *Internet of Things* pertama kali dimunculkan oleh Kevin Ashton pada tahun 1999 di salah satu presentasinya. Kini banyak perusahaan besar mulai mendalami *Internet of Things* sebut saja Intel, Microsoft, Oracle, dan banyak lainnya. Banyak yang memprediksi bahwa pengaruh *Internet of Things* adalah “*the next big thing*” di dunia teknologi informasi, hal ini karena IoT menawarkan banyak potensi yang bisa digali. Contoh sederhana manfaat dan implementasi dari *Internet of Things* misalnya adalah kulkas yang dapat memberitahukan kepada pemiliknya via SMS atau email tentang makanan dan minuman apa saja yang sudah habis dan harus distok lagi [5].

Dasar prinsip kerja perangkat IoT adalah, benda di dunia nyata diberikan identitas unik dan dapat dikali di sistem komputer dan dapat di representasikan dalam bentuk data di sebuah sistem komputer. Pada awal-awal implementasi gagasan IoT pengenalan yang digunakan agar benda dapat diidentifikasi dan dibaca oleh komputer adalah dengan menggunakan kode batang (*Barcode*), Kode QR (*QR Code*) dan Identifikasi Frekuensi Radio (RFID). dalam perkembangannya sebuah benda dapat diberi pengenalan berupa IP address dan menggunakan jaringan internet untuk bisa berkomunikasi dengan benda lain yang memiliki pengenalan IP address [6].

Beberapa contoh standar IoT seperti berikut ini [8]:

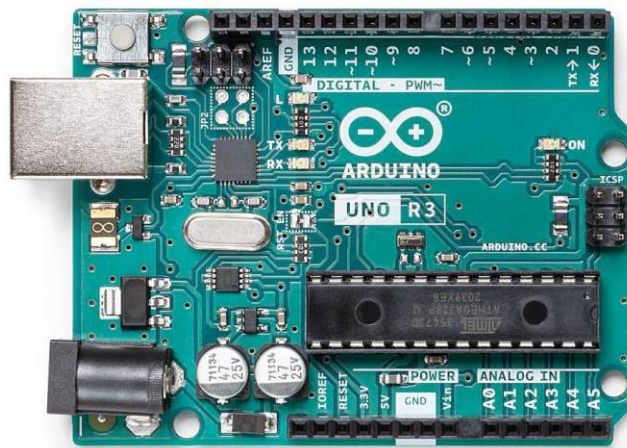
1. *IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN)*, yang merupakan standar terbuka yang ditetapkan oleh *Internet Engineering Task Force (IETF)*. Standar ini memungkinkan radio berdaya rendah untuk berkomunikasi ke internet, termasuk 804.15.4, *Bluetooth Low Energy*, dan *Z-Wave*.
2. *Zigbee*, yaitu jaringan nirkabel berdaya rendah dan berkecepatan data rendah yang kerap digunakan terutama di lingkungan hunian dan juga unit usaha. *Zigbee* didasarkan pada standar IEEE 802.15.4.
3. *Data Distribution Service (DDS)* atau Layanan Distribusi Data, yang dikembangkan oleh *Object Management Group* dan merupakan standar industri IoT untuk komunikasi antar mesin (*machine-to-machine/M2M*) yang *real-time*, terukur, dan berkinerja tinggi.

Sedangkan *frameworks* IoT, beberapa contohnya adalah sebagai berikut

1. *Amazon Web Services (AWS) IoT*, yaitu platform *cloud computing* untuk IoT yang dirilis oleh Amazon.
2. *Arm Mbed IoT*, yang merupakan platform *open-source* guna pengembangan aplikasi IoT berbasis mikrokontroler Arm.
3. Platform *Microsoft Azure IoT Suite*, yaitu serangkaian layanan yang memungkinkan *user* untuk berinteraksi dan menerima data dari perangkat IoT, serta melakukan berbagai operasi pada data.
4. *Calvin*, yaitu platform *open-source* untuk IoT yang dikembangkan oleh Ericsson. *Calvin* menyertakan kerangka pengembangan untuk *developer* aplikasi, serta lingkungan *runtime* untuk menangani aplikasi yang sedang berjalan.

2.4 Arduino Uno R3 ATmega328P

Arduino Uno adalah *board* mikrokontroler berbasis ATmega328. Memiliki 14 pin *input* dan *output* digital. Dimana 6 pin *input* tersebut digunakan sebagai *output* PWM dan 6 pin *input* analog, 16 MHz osilator kristal, koneksi USB, *jack* power, ICSP *header*, dan tombol reset. Untuk mendukung mikrokontroler agar dapat digunakan cukup hanya menghubungkan kabel USB atau listrik dengan AC yang ke adaptor DC atau baterai untuk menjalankannya [9].



Gambar 2.2 Papan mikrokontroler Arduino Uno R3 [10]

2.4.1 Fitur Arduino Uno

Beberapa fitur yang melekat pada Arduino Uno yang juga merupakan keunggulannya atas *development boards* lainnya, diantaranya adalah [11]:

1. Mudah digunakan dan diprogram.
2. Dukungan untuk input analog dan digital.
3. Output PWM yang memungkinkan kontrol yang halus pada aktuator.
4. Kemampuan untuk berkomunikasi dengan perangkat lain melalui koneksi serial.
5. Berbagai jenis sensor dan ekstensi yang kompatibel.

6. Dukungan untuk menyimpan data pada EEPROM (*Electrically Erasable Programmable Read-Only Memory*).
7. Perangkat keras dan perangkat lunak yang dioptimasi untuk pemula.

2.4.2 Spesifikasi Teknis

Spesifikasi teknis yang dimiliki oleh Arduino Uno, sebagaimana tabel di bawah ini [12]:

Tabel 2.1 Spesifikasi teknis *controller board* Arduino Uno R3

<i>Microcontroller</i>	ATMega328P
<i>Operating Voltage</i>	5V
<i>Input Voltage (recommended)</i>	7-12V
<i>Input Voltage (limit)</i>	6-20V
<i>Digital I/O Pins</i>	14
<i>PWM Digital I/O Pins</i>	6
<i>Analog Input Pins</i>	6
<i>DC Current per I/O Pin</i>	20 mA
<i>DC Current for 3.3V Pin</i>	50 mA
<i>Flash Memory</i>	32 KB (ATMega328P) yang terpakai oleh <i>bootloader</i> 0.5 KB
<i>SRAM</i>	2 KB (ATMega328P)
<i>EEPROM</i>	1 KB (ATMega328P)
<i>Clock Speed</i>	16 MHz
<i>LED_BUILTIN</i>	13
Panjang	68.6 mm
Lebar	53.4 mm

2.4.3 Connector Pinouts

Ketersediaan pin yang cukup memadai memungkinkan peminat dalam perancangan dan pengembangan perangkat elektronika bereksplorasi. Tekelompok ke dalam pin analog dan digital, *pinouts* yang ada pada papan Arduino Uno adalah sebagai berikut [13]:

A. Pin Analog

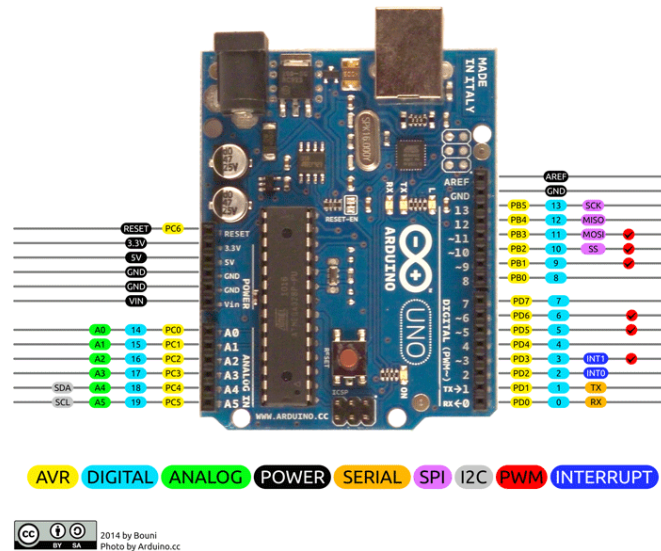
Tabel 2.2 *Pinout* analog papan kendali Arduino Uno R3

Pin	Function	Type	Description
1	NC	NC	Not connected
2	IOREF	IOREF	Reference for digital logic V – connected to 5V
3	Reset	Reset	Reset
4	+3V3	Power	+3V3 Power Rail
5	+5V	Power	5V Power Rail
6	GND	Power	Ground
7	GND	Power	Ground
8	VIN	Power	Voltage input
9	A0	Analog/GPIO	Analog input 0/GPIO
10	A1	Analog/GPIO	Analog input 1/GPIO
11	A2	Analog/GPIO	Analog input 2/GPIO
12	A3	Analog/GPIO	Analog input 3/GPIO
13	A4/SDA	Analog input/I2C	Analog input 4/I2C Data line
14	A5/SCL	Analog input/I2C	Analog input 5/I2C Clock line

B. Pin Digital

Tabel 2.3 *Pinout* digital papan kendali Arduino Uno R3

Pin	Function	Type	Description
1	D0	Digital/GPIO	Digital pin 0/GPIO
2	D1	Digital/GPIO	Digital pin 1/GPIO
3	D2	Digital/GPIO	Digital pin 2/GPIO
4	D3	Digital/GPIO	Digital pin 3/GPIO
5	D4	Digital/GPIO	Digital pin 4/GPIO
6	D5	Digital/GPIO	Digital pin 5/GPIO
7	D6	Digital/GPIO	Digital pin 6/GPIO
8	D7	Digital/GPIO	Digital pin 7/GPIO
9	D8	Digital/GPIO	Digital pin 8/GPIO
10	D9	Digital/GPIO	Digital pin 9/GPIO
11	SS	Digital	SPI Chip Select
12	MOSI	Digital	SPI1 Main Out Secondary In
13	MISO	Digital	SPI Main In Secondary Out
14	SCK	Digital	SPI serial clock output
15	GND	Power	Ground
16	AREF	Digital	Analog reference voltage
17	A4/SD4	Digital	Analog input 4/I2C Data line (duplicated)
18	A5/SD5	Digital	Analog input 5/I2C Clock line (duplicated)



Gambar 2.3 Pinout Arduino Uno R3 [14]

2.5 NodeMCU ESP8266 V3

ESP8266 merupakan sebuah mikrokontroler yang sering digunakan untuk perangkat *Internet of Things* atau yang biasa disebut IoT. Mikrokontroler buatan Espressif Systems ini mempunyai fitur yang cukup lengkap dan mudah digunakan. Salah satu fitur yang paling menonjol adalah modul Wi-Fi. Dengan modul Wi-Fi ini, kita bisa menghubungkan ESP8266 ke internet melalui *access point* sehingga bisa digunakan sebagai perangkat IoT. Selain dapat dijadikan *client* yang terhubung ke *access point*, ESP8266 ini juga bisa dijadikan sebagai *access point* yang bisa digunakan sebagai *web server* [15].



Gambar 2.4 Controller board NodeMCU ESP8266 V3 [16]

Modul Wifi ESP8266 NodeMCU merupakan turunan pengembangan dari modul *platform IoT (Internet of Things)* keluarga esp8266. Pengembangan Kit ini

mengintegrasikan GPIO, PWM (*Pulse Width Modulation*), I2C, dan ADC (*Analog to Digital Converter*) semua dalam satu *board*. NodeMCU ini sendiri yaitu *boardnya* yang berukuran sangat kecil yaitu panjang 4.83 cm, lebar 2.54 cm, dengan berat 7 gram serta dengan daya yang rendah untuk menyalakanya (Anggelausia, 2019). NodeMCU ESP8266 V3 didasarkan pada ESP-12E (Modul WiFi berbasis ESP8266). NodeMCU juga merupakan firmware dan kit pengembangan sumber terbuka yang membantu Anda membuat prototipe produk IOT Anda dalam beberapa baris skrip LUA, dan tentu saja Anda selalu dapat memprogramnya dengan Arduino IDE [17].

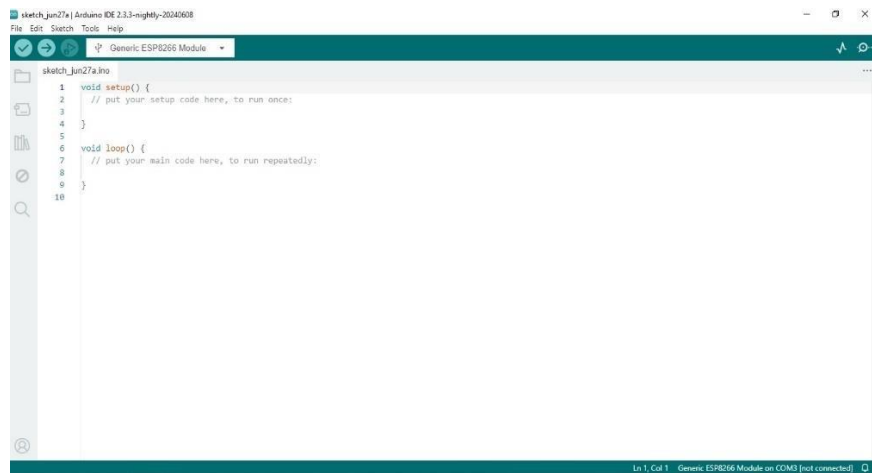
2.5.1 Spesifikasi

Spesifikasi teknis ESP8266 adalah sebagai berikut [18]:

1. Voltage:3.3V.
2. Wi-Fi Direct (P2P), soft-AP.
3. Current consumption: 10uA~170mA.
4. Flash memory attachable: 16MB max (512K normal).
5. Integrated TCP/IP protocol stack.
6. Processor: Tensilica L106 32-bit.
7. Processor speed: 80~160MHz.
8. RAM: 32K + 80K.
9. GPIOs: 17 (multiplexed with other functions).
10. Analog to Digital: 1 input with 1024 step resolution.
11. +19.5dBm output power in 802.11b mode
12. 802.11 support: b/g/n.
13. Maximum concurrent TCP connections: 5.

2.6 Arduino Integrated Development Environment (IDE)

Arduino IDE adalah piranti lunak yang digunakan membuat kode program yang akan ditanamkan pada papan mikrokontroler semisal Arduino Uno, ESP8266, ESP-32, dan yang semisalnya. Sebagai *software* berbasis sumber terbuka, Arduino IDE dapat diunduh tanpa membayar langsung melalui *website* resmi Arduino di URL <https://www.arduino.cc/en/software>.



Gambar 2.6 Antarmuka *software* Arduino IDE

Arduino IDE menggunakan bahasa pemrograman C++. Bahasa C++ adalah bahasa pemrograman yang *strongly-typed* dan fleksibel, serta banyak digunakan untuk pengembangan perangkat lunak dan sistem komputer. Bahasa C++ dikembangkan oleh ilmuwan komputer Denmark, Bjarne Stroustrup dan dirilis pada tahun 1985. Bahasa ini merupakan adaptasi dari bahasa pemrograman C yang populer dengan menambahkan beberapa fitur-fitur canggih seperti pemrograman berorientasi objek (OOP) untuk membangun program yang terstruktur dan efisien [21].

2.7 Fingerprint Scanner AS608



Gambar 2.7 Modul pemindai sidik jari AS608 [22]

Sensor sidik jari AS608 didasarkan pada chip AS608. Sensor ini dapat menangkap sidik jari dan mengirimkan data ke mikrokontroler melalui komunikasi serial. Sensor ini dapat menyimpan hingga 255 ID sidik jari sehingga cocok untuk proyek kecil (srituhobby.com, t.th). Pada artikel lain ditemukan bahwa AS608 sejatinya dapat menampung hingga 162 ID sidik jari.

2.7.1 Fitur

Beberapa fitur yang terdapat pada modul sensor sidik jari AS608 adalah sebagai berikut [23]:

1. Onboard flash memory.
2. Dapat menampung hingga 162 *fingerprints*.
3. Antarmuka *TTL serial* dengan kontroler dan sistem lainnya.
4. *Optical technology*.
5. Ukuran yang cukup ringkas.
6. Tangkapan *image* hingga 500 dpi.

2.7.2 Spesifikasi

Spesifikasi teknis yang dimiliki sensor sidik jari AS608 adalah [23]:

1. *Voltage supply range: 3.6V to 6V*

2. *Maximum operating current: 120mA*
3. *Peak current: 150mA*
4. *Max Prints imaging time: 1s*
5. *False accept rate (FAR): <0.001%*
6. *False reject rate (FRR): <1.0%*
7. *Interface: UART or TTL serial*
8. *Storage capacity: 162 fingerprints*
9. *Signature file: 256 bytes*
10. *Template file: 512 bytes.*
11. *Default baud rate: 57600.*
12. *Window area: 14mm x 18mm*
13. *Working temperature: -20 °C to 50 °C*
14. *Working humidity: 40% RH-85%RH*

2.8 LCD I2C 16x2



Gambar 2.8 LCD I2C 16x2 [24]

LCD adalah media tampilan yang paling mudah untuk diamati karena menghasilkan tampilan karakter yang baik dan cukup banyak. Pada LCD 16×2 dapat ditampilkan 32 karakter, 16 karakter pada baris atas dan 16 karakter pada baris bawah. LCD 16×2 pada umumnya menggunakan 16 pin sebagai kontrolnya, tentunya akan sangat boros apabila menggunakan 16 pin tersebut. Karena itu,

digunakan driver khusus sehingga LCD dapat dikontrol dengan jalur I2C. Melalui I2C maka LCD dapat dikontrol dengan menggunakan 2 pin saja yaitu SDA dan SCL [25].

Spesifikasi teknis yang dimiliki oleh modul LCD I2C adalah sebagai berikut [26]:

1. Tampilan 2 baris, masing-masing baris 16 karakter, 5x8 pixel.
2. *Display controller*: HD44780 (standar industri LCD).
3. Dilengkapi lampu latar warna biru/hijau/kuning.
4. Sudut pandang lebar dengan tingkat kontras yang dapat diatur.
5. Tegangan kerja: 5V DC.
6. Dimensi modul: 80 x 36 x 12 mm.
7. Dimensi layar tampilan: 64,5 x 16 mm

2.9 Reed Switch MC-38



Gambar 2.9 Sensor *reed* MC-38 [27]

Dikenal juga sebagai sensor magnet, *reed switch* merupakan salah satu jenis sensor yang terbilang sangat sederhana. *Reed switch* hanya terdiri dari dua buah plat yang saling berdekatan. *Reed switch* adalah sensor yang berfungsi juga sebagai saklar yang aktif atau terhubung apabila di area jangkauan nya terdapat medan magnet. Medan magnet yang cukup kuat jika melalui area sekitar *reed switch*, dua

buah plat yang salaing berdekatan tadi akan terhubung sehingga akan memberikan rangkaian tertutup bagi rangkaian yang dipasangkannya [28].

2.10 Modul Relay 5V DC

Relay merupakan saklar listrik atau saklar elektrik yang dapat membuka atau menutup rangkaian lain dalam sebuah kondisi tertentu. Pada fungsinya, *relay* merupakan sakelar yang dapat membuka atau menutup dengan tenaga listrik melalui *coil relay* yang berada dalam *relay* tersebut. *Relay* merupakan komponen elektronika berupa saklar elektronik yang digerakkan oleh arus listrik. Secara prinsip, relai merupakan tuas saklar dengan lilitan kawat pada batang besi (*solenoid*) di dekatnya [29].

Ketika *solenoid* dialiri arus listrik, tuas akan tertarik karena adanya gaya magnet yang terjadi pada solenoid sehingga kontak saklar akan menutup. Pada saat arus dihentikan, gaya magnet akan hilang, tuas akan kembali ke posisi semula dan kontak saklar kembali terbuka. Dalam pemakaiannya biasanya *relay* yang digerakkan dengan arus DC dilengkapi dengan sebuah dioda yang diparalel dengan lilitannya dan dipasang terbalik yaitu anoda pada tegangan (-) dan katoda pada tegangan (+). Ini bertujuan untuk mengantisipasi sentakan listrik yang terjadi pada saat relay berganti posisi dari *on* ke *off* agar tidak merusak komponen di sekitarnya [29].



Gambar 2.10 Modul *relay* 5V [30]

2.11 *Buzzer*

Buzzer adalah komponen elektronik yang digunakan untuk menghasilkan suara atau bip. Ini adalah perangkat output yang mengubah sinyal listrik menjadi suara. Buzzer sering digunakan dalam perangkat rumah tangga, jam alarm, komputer, dan banyak perangkat elektronik lainnya sebagai cara untuk memberikan indikasi audio kepada pengguna [31].

2.11.1 *Fitur*

1. Mekanisme Penghasil Suara: Buzzer umumnya bekerja dengan menggunakan elektromagnet, piezoelektrik, atau mekanisme mekanis. Buzzer piezoelektrik, misalnya, menggunakan disk piezoelektrik yang bergetar ketika tegangan listrik diterapkan, menghasilkan suara.
2. Frekuensi Suara: Buzzer menghasilkan suara dengan frekuensi tertentu, yang biasanya dinyatakan dalam Hertz (Hz). Frekuensi ini menentukan nada suara yang dihasilkan.
3. Volume: Intensitas suara yang dihasilkan buzzer biasanya dinyatakan dalam desibel (dB). Volume ini bisa bergantung pada tegangan yang diterapkan dan konstruksi buzzer.
4. Tipe Aktuasi: Beberapa buzzer diaktifkan secara langsung oleh tegangan (buzzer aktif), sedangkan yang lain memerlukan sirkuit pembangkit sinyal eksternal untuk menghasilkan suara (buzzer pasif) [31].

2.11.2 *Jenis-jenis Buzzer*

1. Buzzer Elektromagnetik: Menggunakan kumparan dan magnet untuk menghasilkan getaran yang menghasilkan suara.
2. Buzzer Piezoelektrik: Menggunakan material piezoelektrik yang bergetar ketika tegangan listrik diterapkan. Umumnya lebih kecil dan menghasilkan suara yang lebih tajam dibandingkan dengan buzzer elektromagnetik.
3. Buzzer Mekanis: Menggunakan komponen mekanis seperti diaphragm untuk menghasilkan suara. Tidak umum digunakan dalam aplikasi elektronik modern.

4. Buzzer digunakan untuk memberikan indikasi audio dalam berbagai situasi, seperti memberi peringatan, menandakan operasi perangkat, atau sebagai umpan balik dalam antarmuka pengguna. Dalam dunia elektronik hobi seperti Arduino, buzzer sering digunakan dalam proyek-proyek untuk memberikan sinyal audio sederhana seperti pemberitahuan atau alarm.

Dari segi penerapan, terdapat 2 jenis buzzer, yaitu aktif dan pasif.

1. Buzzer aktif, adalah buzzer yang sudah memiliki suaranya sendiri saat diberikan tegangan listrik.
2. Buzzer pasif, adalah jenis buzzer yang tidak memiliki suara sendiri sehingga perlu menentukan jenis suara yang akan dihasilkan.

Pada perangkat alarm, penulis menggunakan 2 (dua) macam *buzzer*. Yaitu *buzzer* aktif, menggunakan SFM-27 dan *buzzer* pasif dengan tegangan 5V.



Gambar 2.11 Buzzer aktif SFM-27 [32]



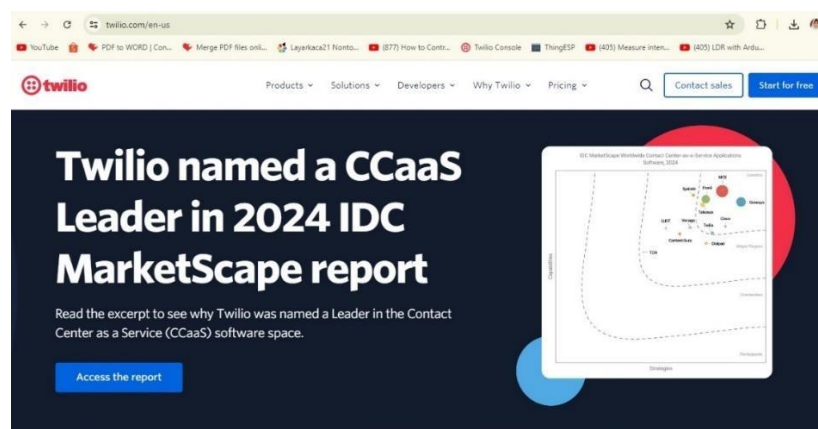
Gambar 2.12 Buzzer pasif 5V [33]

2.12 ThingESP

ThingESP adalah *library* yang digunakan oleh *client* untuk menghubungkan perangkat *Internet of Things* (IoT) ke platform *cloud*. ThingESP menyediakan *Application Programming Interface* (API), yang berfungsi sebagai penghubung antara sebuah aplikasi dan aplikasi lainnya, untuk memungkinkan integrasi fitur tanpa harus menambahkan data secara manual. Mudah-mudahan, peran API seperti seorang interpreter atau penerjemah yang membantu dua orang dari negara dan bahasa yang berbeda untuk saling berkomunikasi. ThingESP merupakan *client library* yang dirancang khusus untuk Arduino IDE yang menyambungkan perangkat IoT ke *ThingESP Cloud Platform*.

2.13 Twilio

Menyitir dari halaman resminya, Twilio adalah platform keterlibatan pelanggan yang digunakan oleh lebih dari 300.000 perusahaan global dan pengganggu digital ditambah lebih dari 10 juta pengembang di seluruh dunia untuk membangun pengalaman unik dan terpersonalisasi bagi pelanggan mereka. Twilio dikenal dalam mendemokratisasi saluran seperti suara, teks, chat, video, dan email melalui API, sehingga memberikan organisasi alat untuk membangun interaksi yang bermakna dengan pelanggan di saluran yang dipilih [34].



Gambar 2.13 Tangkapan layar laman beranda Twilio [35]

Twilio memungkinkan pengembang perangkat lunak membangun proses komunikasi bisnis secara semi-otomatis yang memungkinkan:

1. Melakukan/menerima panggilan telepon.
2. Mengirim/menerima pesan teks.
3. Meluncurkan kampanye email.
4. Tetap terhubung dengan pelanggan

Selain itu, Twilio juga mengotomatiskan verifikasi pengguna menggunakan API layanan webnya [36].

2.14 WhatsApp



Gambar 2.14 *Icon WhatsApp* [37]

WhatsApp adalah aplikasi berkirim pesan dan panggilan yang sederhana, aman, dan reliabel, serta dapat diunduh ke ponsel di seluruh dunia secara gratis . Jika dilihat dari fungsinya WhatsApp hampir sama dengan aplikasi SMS yang biasa Anda gunakan di ponsel lama. Tetapi WhatsApp tidak menggunakan pulsa, melainkan data internet [38]. Untuk menggunakan aplikasi ini, secara gratis pengguna *smartphone* berbasis Android OS dapat mengunduhnya di PlayStore.

2.14.1 Fitur-fitur Unggulan WhatsApp

WhatsApp mempunyai beberapa fitur [38]:

1. Mengirim pesan teks
2. Mengirim foto dari galeri ataupun dari kamera
3. Mengirim video
4. Mengirimkan berkas-berkas kantor atau yang lainnya
5. Menelpon melalui suara, termasuk mengirim pesan suara yang dapat didengarkan oleh penerima setiap saat.
6. Berbagi lokasi memanfaatkan GPS
7. Mengirimkan kartu kontak
8. WhatsApp juga mendukung beberapa emoji, namun untuk stiker, WhatsApp tergolong minimalis. Berbeda dengan LINE yang lebih getol mengembangkannya.
9. Di WhatsApp, pengguna juga dapat mengatur panel profilnya sendiri, terdiri dari nama, foto, status serta beberapa alat pengaturan privasi untuk melindungi profil dan juga alat bantuan untuk membackup pesan, mengubah nomor akun dan melakukan pembayaran.

BAB III

RANCANG BANGUN PERANGKAT

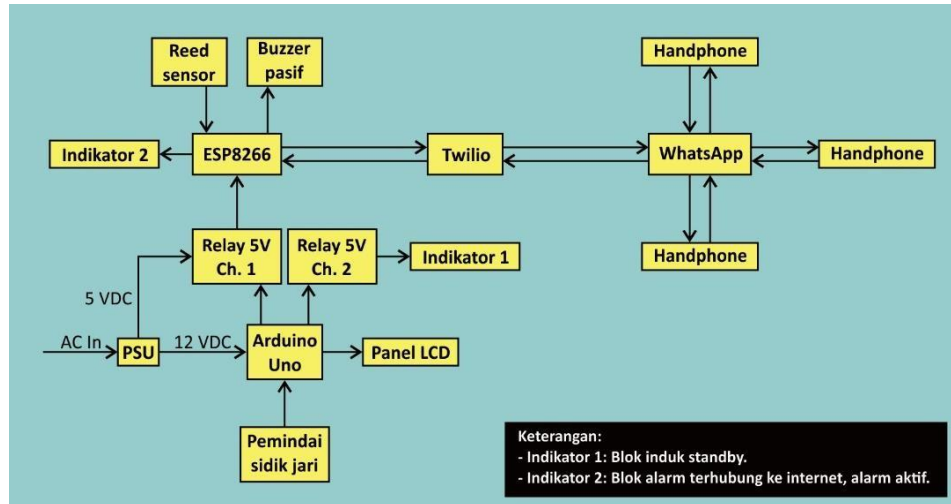
3.1 Tujuan Perancangan

Perangkat alarm ini dirancang sebagai sebuah perangkat sistem keamanan yang sederhana, mudah digunakan namun memiliki kemampuan pencegahan tindak kejahatan yang handal. Sidik jari sebagai “fitur” identifikasi nyaris mustahil untuk dipalsukan. Karakter unik sidik jari yang berbeda pada setiap manusia membuat akses kendali utama untuk perangkat alarm pada situasi darurat menjadi sangat terbatas dan tidak mudah ditembus.

Pemanfaatan aplikasi telepon seluler WhatsApp sebagai media kontrol dan pemantauan adalah salah satu kemudahan yang diaplikasikan pada perangkat alarm. Khusus di Indonesia, mengutip goodstat.id (Hasya, 2023) WhatsApp menduduki peringkat pertama platform komunikasi berbasis internet yang paling banyak digunakan, dengan jumlah sebanyak 92,1% pemakai internet yang berusia 16-64 tahun. Popularitas tersebut bukan tanpa alasan. *User friendly*, itulah salah satu andalan aplikasi perpesanan WhatsApp yang juga diterapkan pada prosedur kendali dan pemantauan perangkat alarm.

Pemrograman *controller board* menggunakan Arduino IDE yang bersifat sumber terbuka memungkinkan pengaturan serta pengembangan protokol instruksi pada perangkat alarm menjadi fleksibel. Pengguna dapat melakukan *setting* sesuai kebutuhan. Inovasi juga terbuka untuk diterapkan pada perangkat alarm, seperti; penambahan fitur kunci darurat, penambahan sensor deteksi gerak, dan sebagainya.

3.2 Diagram Blok



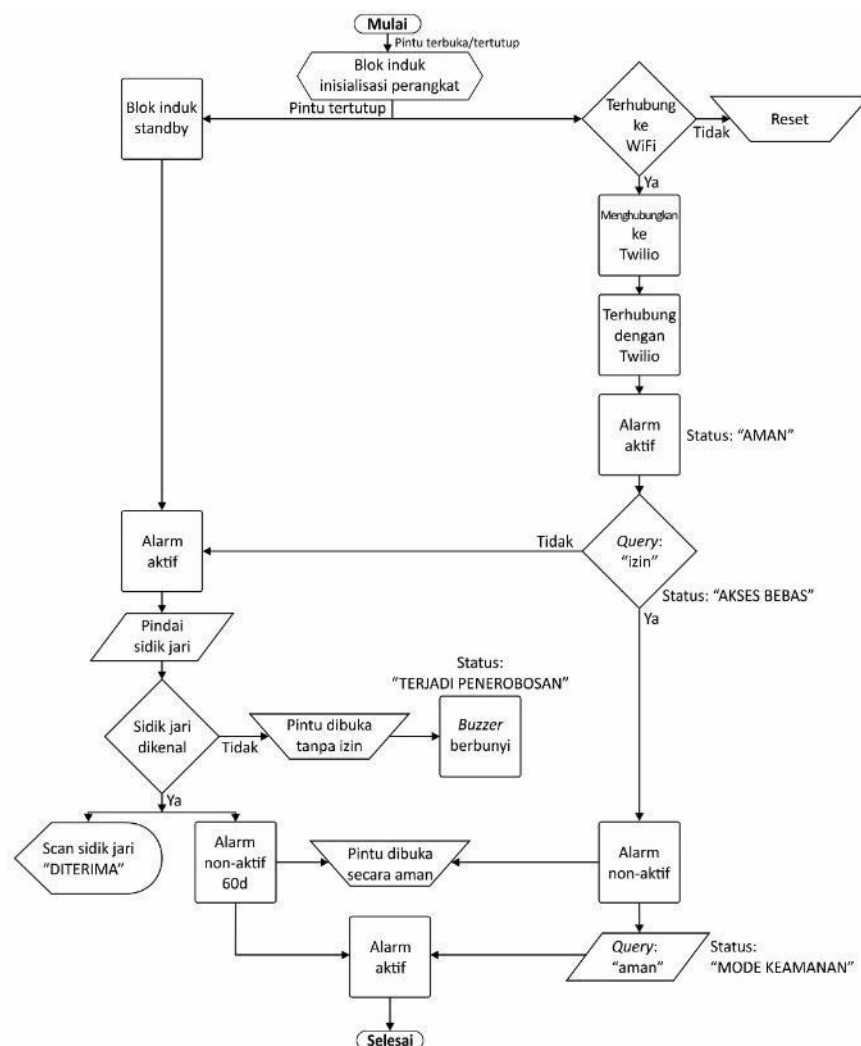
Gambar 3.1 Diagram blok perangkat alarm

3.3 Prinsip Kerja Perangkat

1. Perangkat alarm terdiri dari tiga bagian; blok **power supply**, **blok induk** (yang terdiri dari: papan kendali Arduino Uno R3, modul pemindai sidik jari, modul *relay* 5VDC 2 channel, penampil LCD, lampu indikator), dan **blok alarm** (yang terdiri dari: papan kendali NodeMCU ESP8266 Lolin, *reed sensor*, *buzzer* pasif, lampu indikator).
2. Blok induk ditenagai dengan tegangan 12VDC langsung dari PSU, blok alarm mendapatkan arus 5VDC melalui modul *relay* yang dikendalikan oleh blok induk.
3. Pada keadaan tertentu, dengan akses identifikasi sidik jari blok induk dapat memutus arus DC ke blok alarm selama 60 detik dan mengalirkan arus kembali secara otomatis.
4. Pada keadaan wajar, alarm dapat dikendalikan dan dipantau melalui *handphone* dengan aplikasi WhatsApp yang memanfaatkan layanan *cloud* Sandbox dari Twilio.

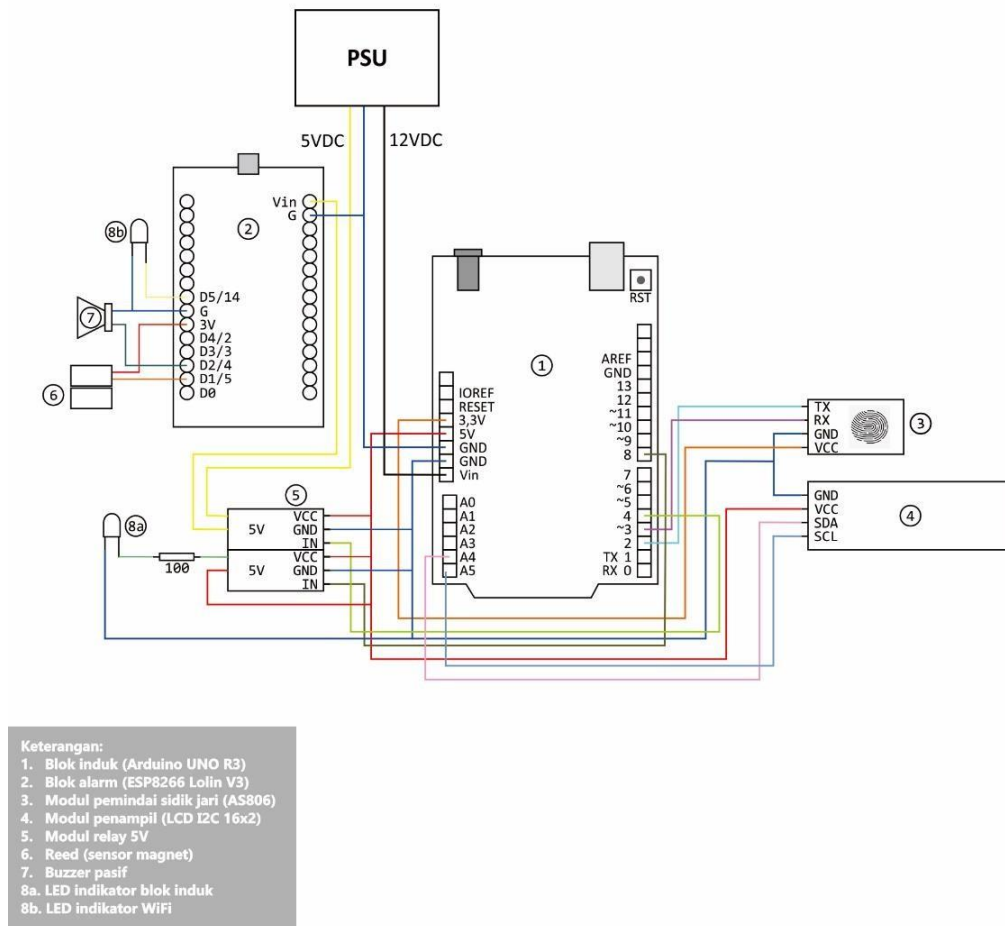
5. Akses kendali dan pemantauan melalui WhatsApp dapat dilakukan oleh/dari lebih dari 1 (satu) nomor WhatsApp yang diberikan kode *request* untuk turut memiliki hak akses atas perangkat.
6. Penonaktifan alarm pada situasi terjadi penerobosan hanya dapat dilakukan melalui akses identifikasi sidik jari atau memutus arus listrik perangkat secara keseluruhan.

3.4 Flowchart Kerja Perangkat



Gambar 3.2 Flowchart perangkat alarm

3.5 Skema Rangkaian



Gambar 3.3 Skema rangkaian

3.6 Perakitan Perangkat

Perangkat alarm secara keseluruhan terdiri dari tiga blok, yaitu blok *PSU*, blok induk, dan blok alarm. Blok induk dikendalikan oleh papan kontroler Arduino Uno, sementara blok alarm dikendalikan oleh ESP8266 Lolin. Keduanya memiliki fungsi dan tugas yang berbeda. Disebabkan spesifikasi tugas berbeda, pada papan kendali yang berbeda, perakitan perangkat dilakukan dalam 3 tahap; perakitan blok induk (beserta pengkodean instruksi), perakitan blok alarm (beserta pengkodean instruksi), penggabungan.

3.6.1 Perangkaian Perangkat Keras

A. Power Supply Unit (PSU)

Suplai daya untuk perangkat alarm ini diperoleh dari PSU yang biasa digunakan untuk *Desktop Computer*. Sejatinya PSU yang digunakan memiliki 3 (tiga) tegangan keluaran, yaitu; 3.3V, 5V, dan 12V, namun untuk kebutuhan perangkat alarm hanya keluaran 5V dan 12V yang terpakai. Tegangan 5V diarahkan ke blok alarm melalui modul *relay* yang dikendalikan blok induk, sedangkan tegangan 12V mengalir ke papan kendali Arduino Uno yang berada pada blok induk.

B. Blok Induk

Alat-alat yang beroperasi di blok induk adalah;

1. Papan kendali Arduino Uno Rev. 3 ATmega 328P, terpasang pada Arduino *Shield*.
2. Modul sensor sidik jari AS608
3. Panel LCD I2C 16x2
4. Modul *relay* 5 VDC 2/4 *channel*
5. *White* LED 3-5 V

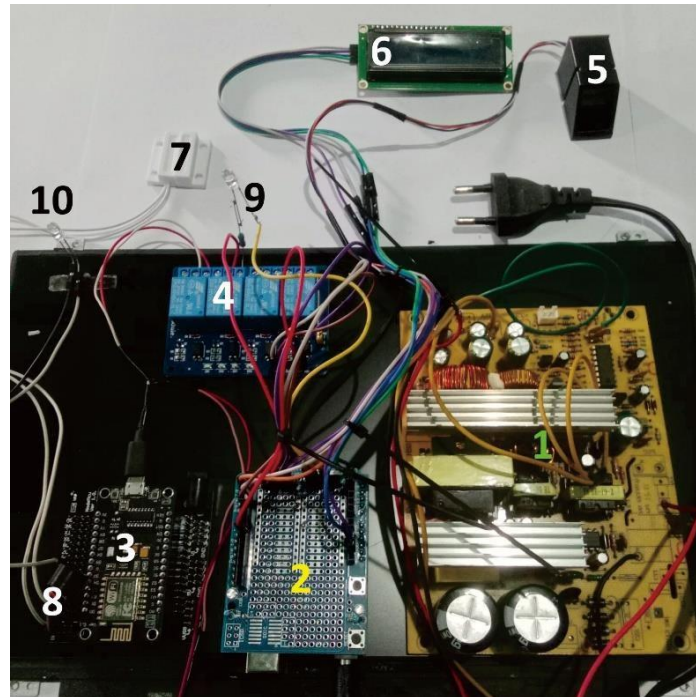
C. Blok Alarm

Spesifikasi teknis blok alarm adalah sebagai berikut

1. *Controller board* NodeMCU ESP8266 V3 Lolin, terpasang pada NodeMCU *Baseboard*.
2. *Reed switch*
3. *Buzzer* pasif
4. *White* LED 3-5 V

Pada perangkaian tahap awal, daya masih dialirkan melalui *port* DC Input untuk kedua papan kendali. Hal ini ditujukan agar jika terjadi ketidakstabilan asupan daya, potensi kerusakan terhadap papan kendali maupun komponen lainnya dapat dicegah secepatnya. Selain jalur *input* daya, berdasarkan skema yang telah

dipersiapkan sebelumnya, rangkaian awal keseluruhan perangkat alarm tampak pada gambar berikut



Gambar 3.4 Rangkaian awal perangkat alarm

Keterangan:

1. *Power Supply Unit.*
2. *Controller board Arduino Uno Rev.3.*
3. *Controller board NodeMCU ESP8266 Lolin.*
4. *Modul relay Modul relay 5 VDC 4 channel.*
5. *Modul pemindai sidik jari AS608.*
6. *Panel LCD I2C 16x2.*
7. *Reed sensor.*
8. *Buzzer pasif.*
9. *LED indikator blok induk.*
10. *LED indikator blok alarm.*

3.6.2 Coding Program Blok Induk (Arduino Uno)

Kode program blok induk bukan kompilasi asli dari penulis, melainkan modifikasi dari sumber <https://srituhobby.com/how-to-use-the-as608-fingerprint-sensor-with-arduino/>. Kode utuh dari sumber yang penulis gunakan terlampir pada laporan ini. Beberapa perubahan pada kompilasi kode, penulis tampilkan sebagai perbandingan dengan kode asli.

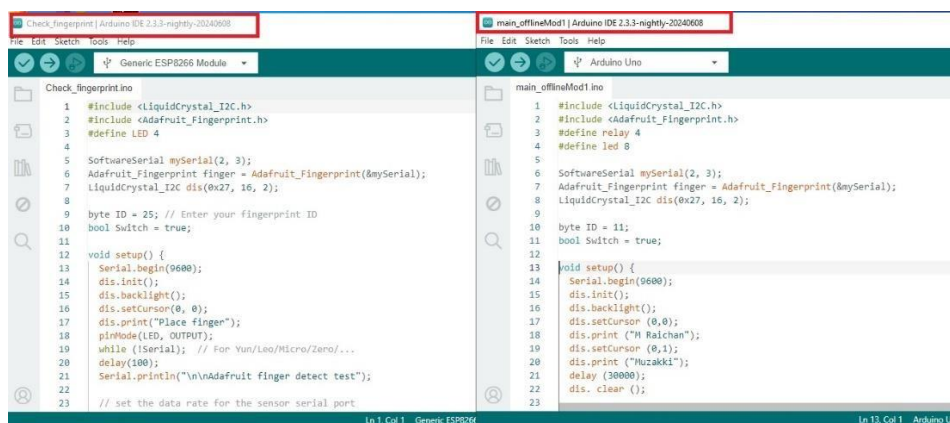
A. Mendaftarkan Sidik Jari

Pada tahap pendaftaran sidik jari, penulis sama sekali tidak melakukan perubahan atas kompilasi kode yang telah penulis unduh melalui situs *web* di atas. Perubahan tidak diperlukan karena pada dasarnya kode yang terunduh tersebut berbasis kode dari program *library* **Adafruit_Fingerprint.h**. Kemudahan dari penggunaan kode yang terunduh tersebut adalah tidak perlu untuk membuka *sketch* yang berbeda ketika hendak melakukan *enroll* sidik jari, sebagaimana lazimnya jika mengakses fitur *enroll* tersebut melalui menu *Example* pada Arduino Ide.

B. Instruksi Blok Induk

Beberapa perubahan pada kode dari sumber asli adalah sebagai berikut:

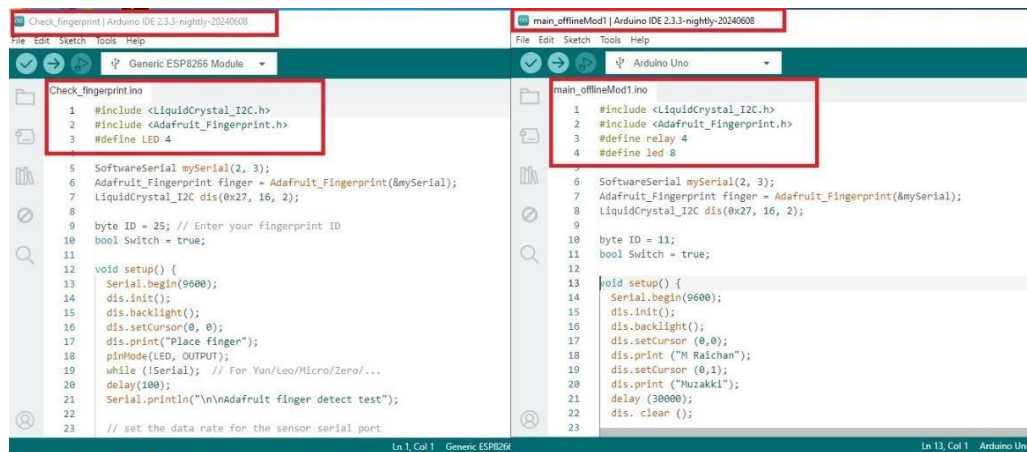
1. Nama *Sketch*



3.5 Nama *Sketch*

Nama *sketch* yang berisi kode dari sumber asli adalah **Check_fingerprint**, berdampingan dengan *sketch* berisi kode yang ditulis ulang, **main_offlineMod1**.

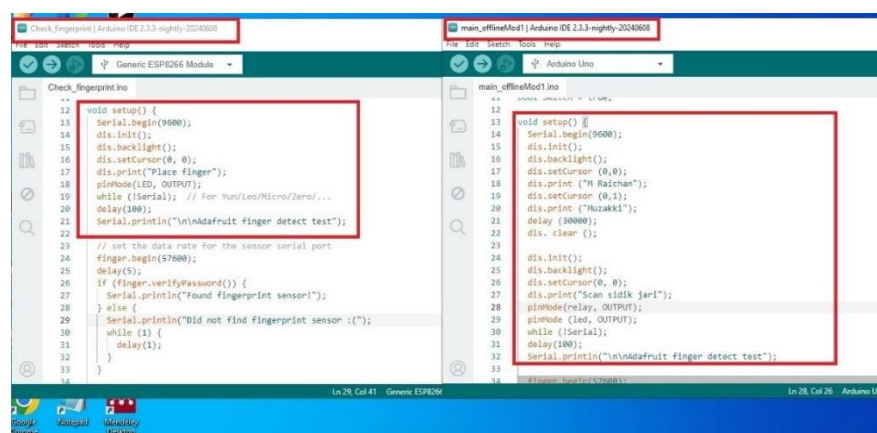
2. Perubahan bagian *header* dan deklarasi variabel



Gambar 3.6 *Header* dan deklarasi variabel

Perubahan pada deklarasi variabel disesuaikan dengan komponen yang tersambung ke papan kendali Arduino Uno, satu buah modul *relay* 4 *channel* dan satu buah LED putih yang akan menjadi indikator blok induk beroperasi.

3. Perubahan instruksi awal

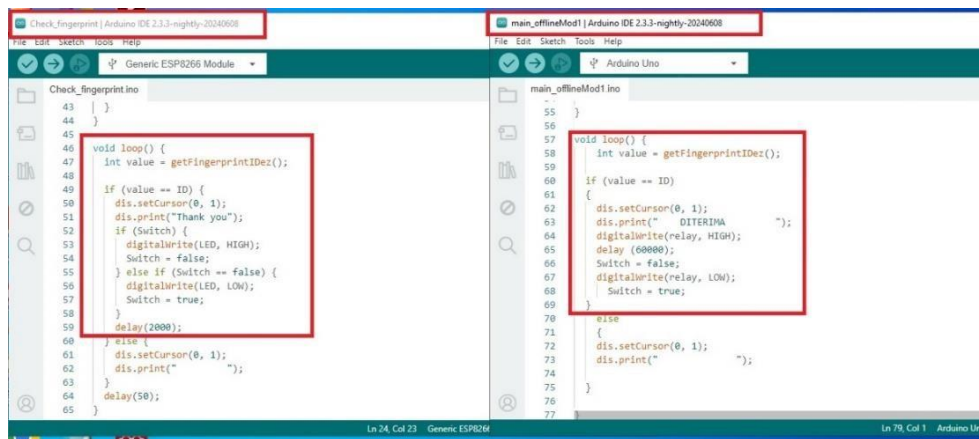


Gambar 3.7 *Void setup*

Inisiasi ini bertujuan memberi jeda kepada pengguna melakukan tindakan-tindakan yang diperlukan—seperti menutup dan mengunci pintu—sebelum fungsi alarm aktif. Hal terpenting yang ditambahkan pada bagian ini adalah fungsi *delay* selama 30 detik sebelum eksekusi program utama. Pada tahap ini, hanya panel LCD dan pemindai sidik jari yang memberi tanda bahwa perangkat mulai bekerja.

4. Perubahan program utama

Instruksi utama yang ditanamkan pada perangkat dengan menggunakan program Arduino IDE dituliskan pada bagian *void loop*.



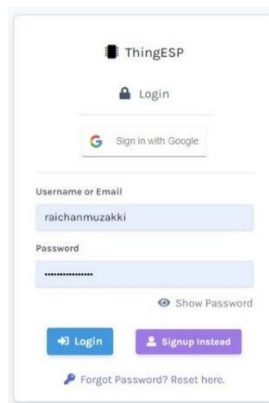
Gambar 3.8 Void loop

Pada bagian ini statemen *else if* dihapus, yang berarti hanya dilakukan satu kali pemindaian. Sidik jari yang teridentifikasi menjadi akses memutus aliran daya ke blok alarm melalui *relay* selama jeda 1 menit dengan penambahan fungsi *delay*. Sedangkan jika sidik jari tidak teridentifikasi, maka tidak ada prosedur kerja alarm yang mengalami perubahan. *Output HIGH* pada *relay* menunjukkan bahwa sambungan ke VIN blok alarm berada pada posisi NC.

3.6.3 Konfigurasi ThingESP, TwilioWhatsApp

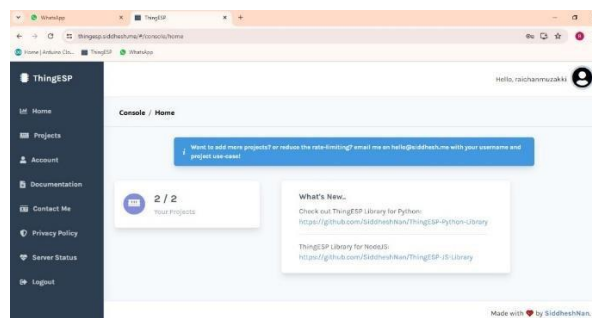
1. Mendaftar/masuk akun ThingESP

Pendaftaran atau masuk kembali ke akun ThingESP dengan mengunjungi website <https://thingesp.siddhesh.me/#/>. Jika belum memiliki akun, membuat akun baru dapat dilakukan secara mudah dengan menggunakan akun Google yang aktif.



Gambar 3.9 Kotak verifikasi *username* dan *password* ThingESP

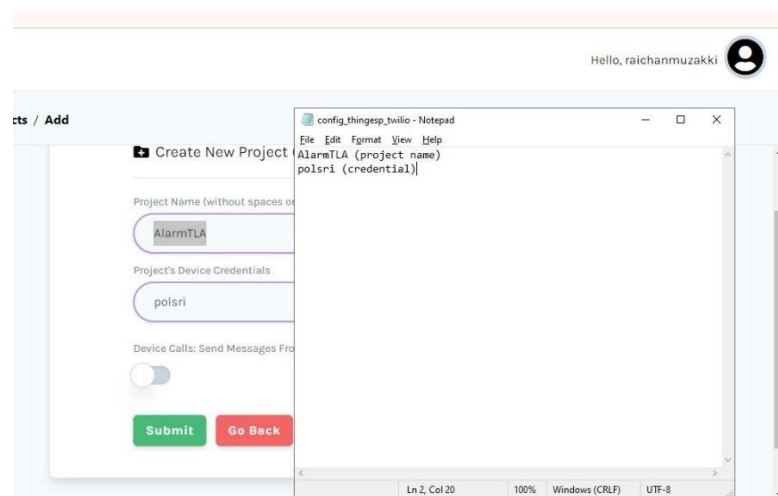
Setelah tahap verifikasi, selanjutnya laman beranda ThingESP tampil seperti gambar di bawah ini. Nama pengguna (*username*) tampil pada pojok kanan atas laman beranda. Penting untuk mengingat ataupun menyalin *username* ThingESP, karena akan digunakan ketika penulisan kode program perangkat, bersama dengan nama proyek dan kredensial perangkat yang akan didapat pada tahap pembuatan proyek.



Gambar 3.10 Beranda ThingESP

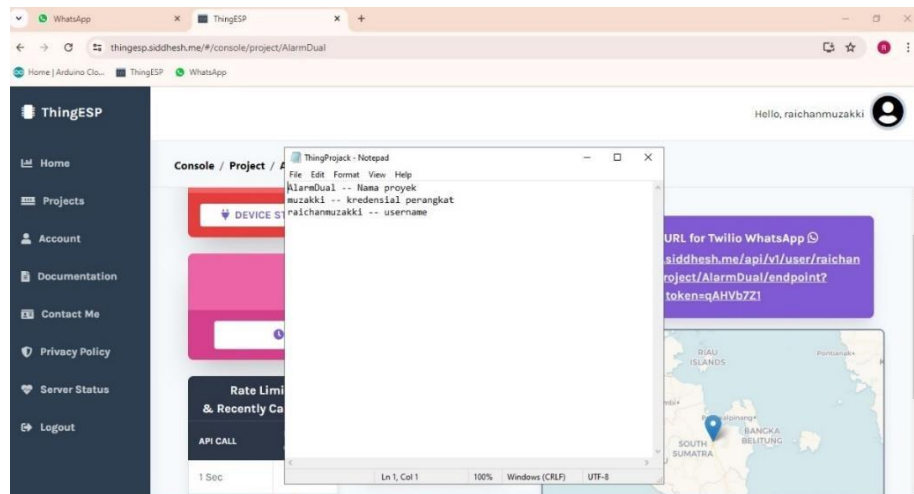
2. Membuat *project* ThingESP

Nama proyek ThingESP dan kredensial perangkat selanjutnya akan terpakai pada tahap penyusunan kode program blok alarm (*controller board* ESP8266). Untuk menghindari kekeliruan, nama proyek dan kredensial perangkat yang didaftarkan di ThingESP disalin terlebih dahulu kemudian disimpan. Penulis hanya menggunakan Notepad untuk menyalin nama proyek dan kredensial perangkat.



Gambar 3.11 Salinan nama proyek dan kredensial perangkat

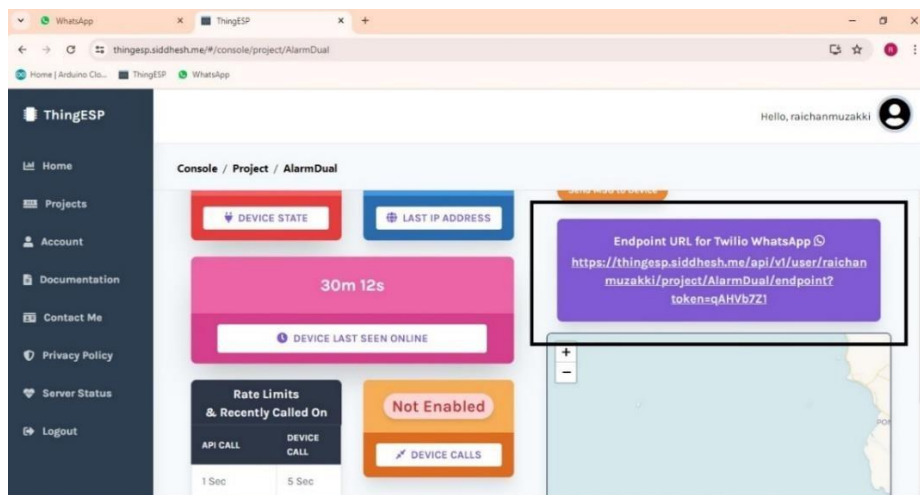
Pada gambar di atas penulis menggunakan nama proyek (perangkat IoT yang akan menggunakan layanan API dari ThingESP) **AlarmTLA** dan kredensial perangkat **polsri**. Nama *project* tersebut hanya contoh, karena sebelumnya penulis telah membuat proyek yang menjadi obyek Tugas Laporan Akhir dengan nama **AlarmDual** dan kredensial perangkat **muzakki**, sebagaimana tangkapan layar berikut:



Gambar 3.12 Nama proyek dan kredensial perangkat tugas Laporan Akhir

Selanjutnya, ThingESP akan memberikan *Endpoint URL* yang merupakan titik akhir perangkat (*API client*) akan dapat diakses atau dilayani perangkat lainnya (*API server*) dalam suatu jaringan internet. Cukup klik pada tautan, secara otomatis alamat *endpoint* akan tersalin dengan sendirinya ke *clipboard*.

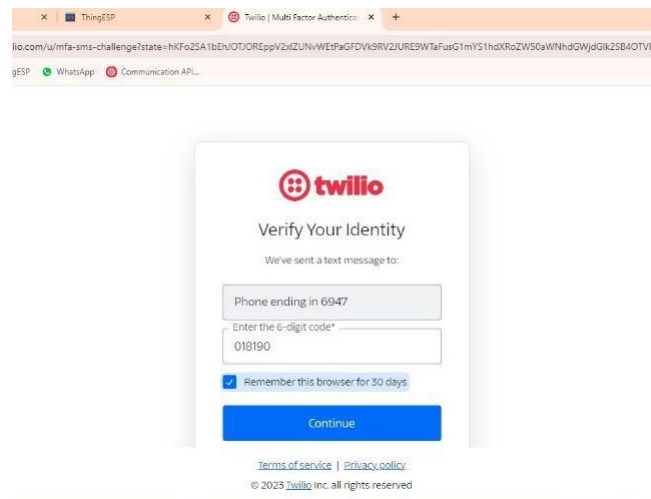
Hingga tahap ini, konfigurasi ThingESP selesai. Langkah selanjutnya adalah melakukan konfigurasi Twilio WhatsApp, sebaiknya *tab* peramban yang membuka situs *web* ThingESP biarkan tetap terbuka.



Gambar 3.13 *Endpoint URL* perangkat alarm *project* ThingESP

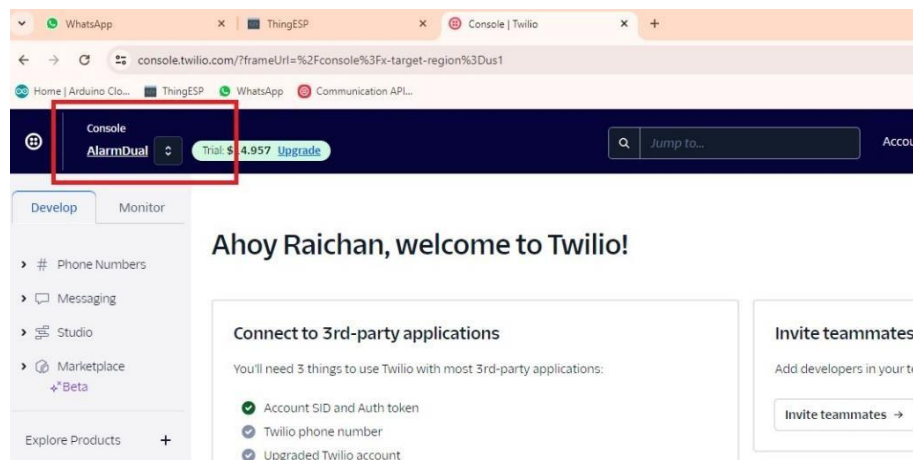
3. Masuk/mendaftar akun Twilio

Website Twilio dapat dikunjungi melalui URL <https://www.twilio.com/en-us>. Sama seperti ThingESP, jika belum memiliki akun Twilio, prosedur pendaftaran dapat dilakukan dengan menggunakan akun Google yang aktif. Selanjutnya Twilio mengirimkan kode otentifikasi melalui SMS ke nomor *handphone* pemilik akun. Nomor ini dapat saja berbeda dengan nomor WhatsApp yang akan dihubungkan ke layanan API Twilio.



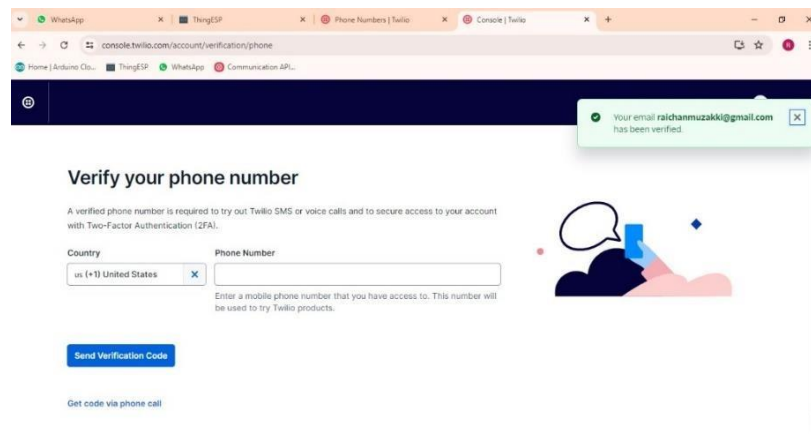
Gambar 3.14 Memasukkan kode otentifikasi ke kotak verifikasi

Setelah memasukkan kode otentifikasi yang telah diterima melalui SMS, layar konsol Twilio akan segera terbuka. Masih di laman konsol, langkah selanjutnya adalah membuat akun baru untuk proyek yang sebelumnya telah dikonfigurasi di ThingESP.

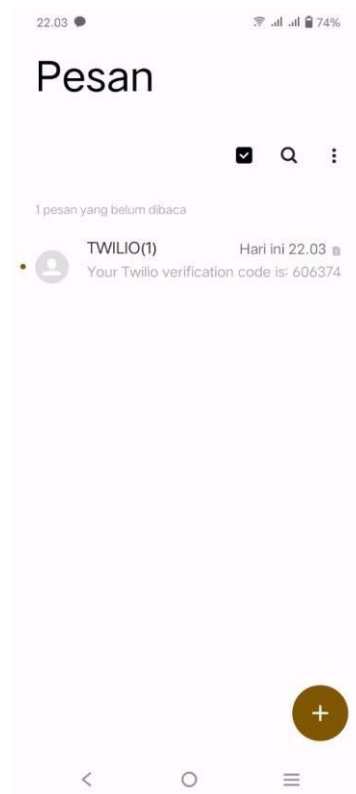


Gambar 3.15 Akun baru konsol Twilio

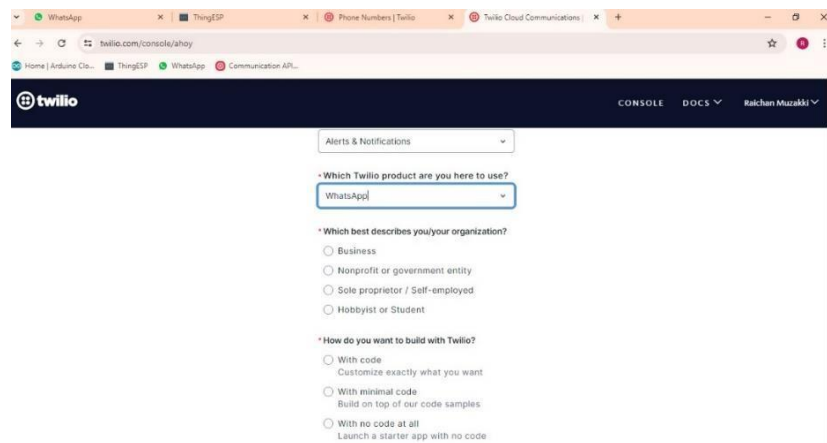
Twilio akan kembali meminta verifikasi. Setelah memasukkan kode negara dan nomor telepon yang sesuai, Twilio mengirimkan kode verifikasi melalui SMS ke nomor bersangkutan.



Gambar 3.16 Verifikasi nomor telepon yang juga nomor WhatsApp



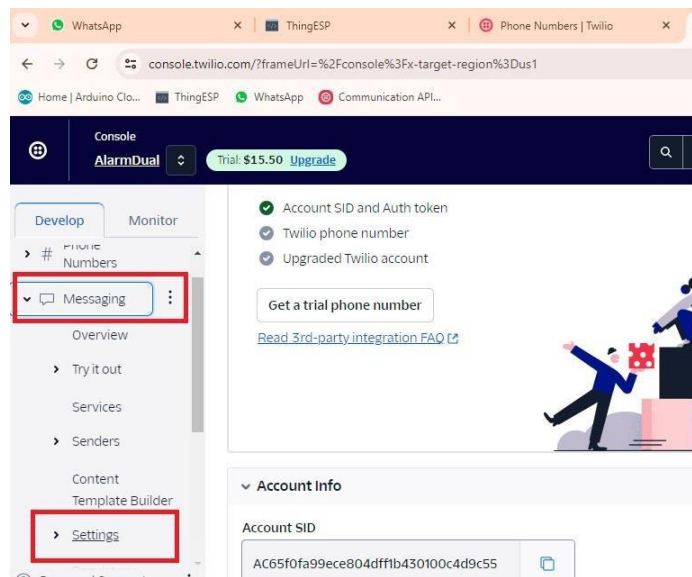
Gambar 3.17 Kode verifikasi nomor telepon



Gambar 3.18 Memilih aplikasi perpesanan yang akan digunakan

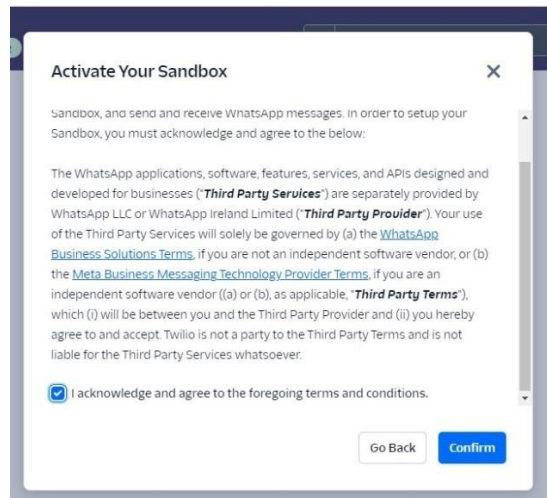
3. Konfigurasi Twilio Sandbox

Karena aplikasi pihak ketiga yang akan digunakan untuk mengirimkan pesan ke perangkat API adalah WhatsApp, maka layanan Twilio yang dipakai adalah Sandbox.

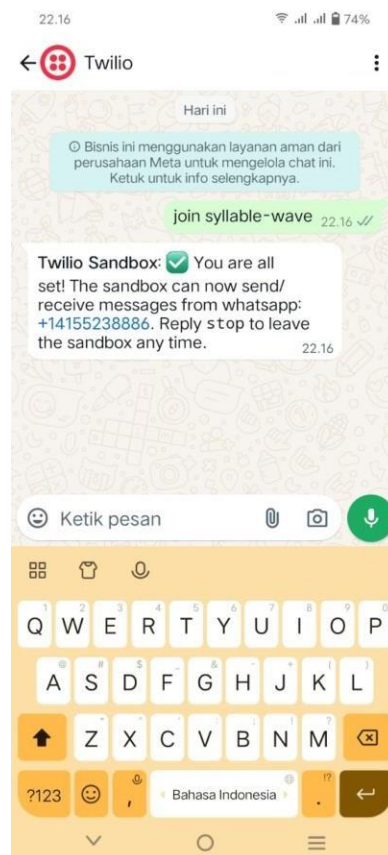


Gambar 3.19 Pengaturan model perpesanan

Jika belum memiliki atau belum pernah menggunakan layanan Twilio Sandbox untuk WhatsApp, memulai layanan dapat dilakukan dengan klik **Try it out**, kemudian akan muncul menu mengaktifkan layanan Sandbox untuk WhatsApp. Selanjutnya setelah memberikan konfirmasi mengaktifkan Sandbox, Twilio memberikan kode permintaan penggunaan layanan untuk dikirimkan ke nomor server Twilio WhatsApp melalui nomor WhatsApp yang akan digunakan. Pada proyek yang penulis buat, kode *request* adalah **join syllable-wave**.



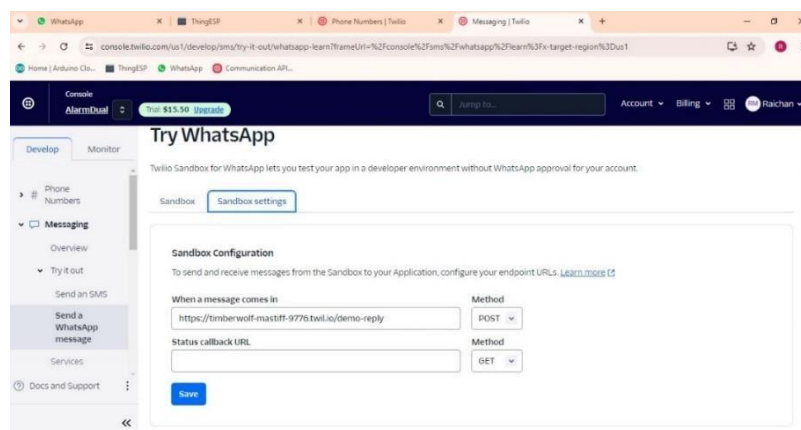
Gambar 3.20 Mengaktifkan Twilio Sandbox



Gambar 3.21 Kode *request* dan notifikasi nomor WA telah dapat menggunakan layanan Sandbox

Kode *request* yang diberikan dapat dibagikan ke nomor WhatsApp lainnya. Setiap nomor WhatsApp yang mengirimkan kode tersebut ke *server* Twilio akan dapat memiliki hak akses atas perangkat API yang terhubung ke Twilio dengan kode *request* yang sama. Hal ini merupakan salah satu keunikan serta kemudahan yang ditawarkan oleh Twilio. Dengan demikian, misal perangkat alarm yang terpasang di sebuah rumah dengan beberapa anggota keluarga pengguna WhatsApp, setiap anggota keluarga akan memiliki hak akses atas perangkat tersebut.

Setelah memperoleh konfirmasi menggunakan layanan Sanbox, selanjutnya masuk ke *Sandbox settings* untuk memasukkan *Endpoint URL* (yang didapat dari ThingESP) perangkat.



Gambar 3.22 Konfigurasi Sandbox

Terakhir, klik **Save**, konfigurasi Twilio Whatsapp selesai.

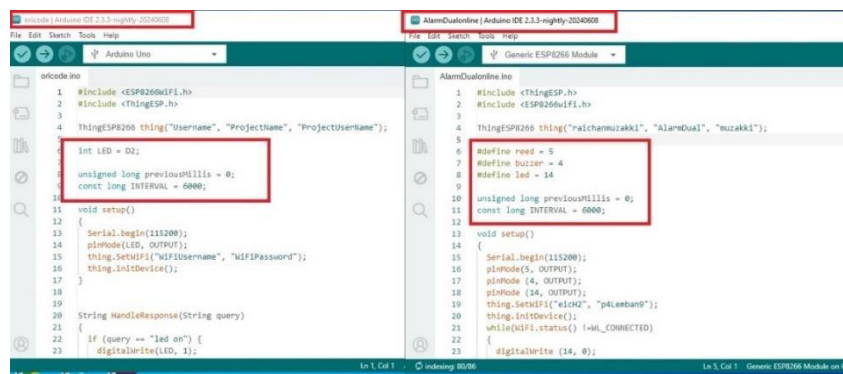
3.6.4. Coding Program Blok Alarm (ESP8266)

Seperti halnya *coding* blok induk, kode program untuk blok alarm penulis ambil dari sumber <https://github.com/Tech-Trends-Shameer/Esp-8266-Projects/tree/main/Control-LED-Using-WhatsApp> milik Shameer Mohamed. Kode asli yang berasal sumber di atas dari penulis cantumkan pada bagian lampiran Laporan Akhir ini.

Beberapa modifikasi signifikan, mengingat kode asli bukan kode program untuk perangkat alarm. Sebagaimana pada bagian *coding* blok induk, kode yang penulis kompilasi ulang ditampilkan (sebagai tangkapan layare) berdampingan dengan kode asli. *Sketch* kode asli penulis simpan dengan nama **oricode**, sementara kode program yang penulis susun ulang tersimpan sebagai *sketch* dengan nama **AlarmDualonline**.

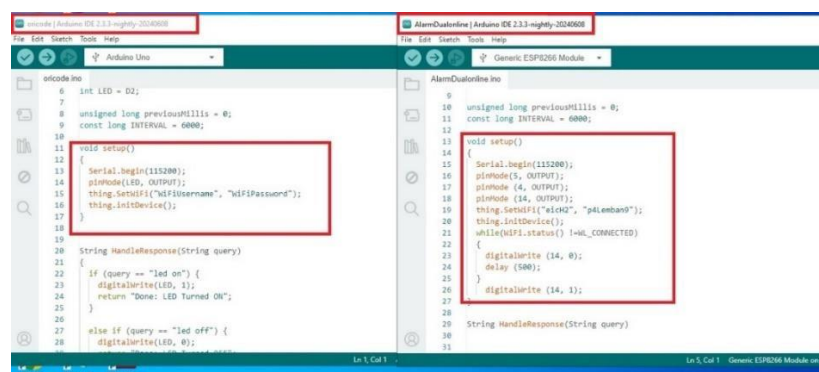
1. Perubahan deklarasi variabel

Bagian ini disesuaikan dengan komponen input/output perangkat alarm. Penambahan paling mencolok adalah kehadiran *reed sensor* (yang terpasang pada pin 5/D1) dan *buzzer* (yang terpasang pada pin 4/D2)



Gambar 3.23 Perubahan deklarasi variabel

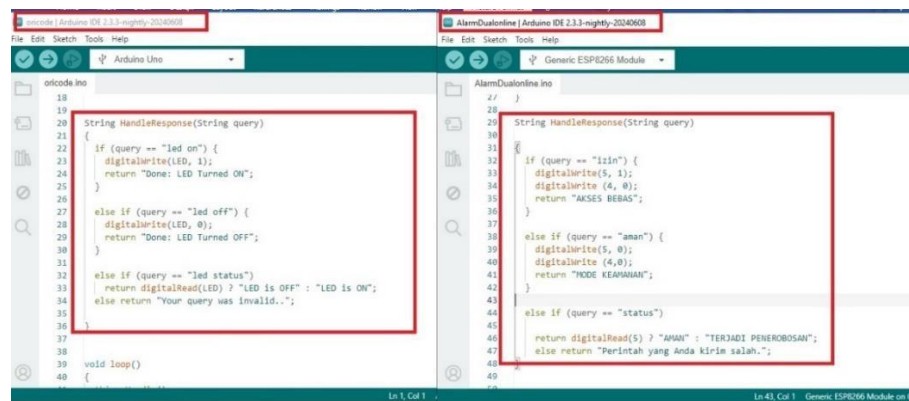
2. Perubahan void setup



Gambar 3.24 Perubahan void setup

Pada konfigurasi memulai program, penulis menambahkan *statement* logika **while** yang akan memberikan perintah ke LED (pin 14) bertindak sebagai indikator dengan menyala jika ESP8266 telah terhubung ke koneksi WiFi. Logika **while** ini digunakan sebagai instruksi berulang setiap kali ESP8266 terkoneksi ke WiFi, baik ketika pertama kali perangkat dihidupkan atau *reboot* setelah terjadi penonaktifan sementara melalui pemutusan arus daya oleh blok induk.

3. Perubahan *query*

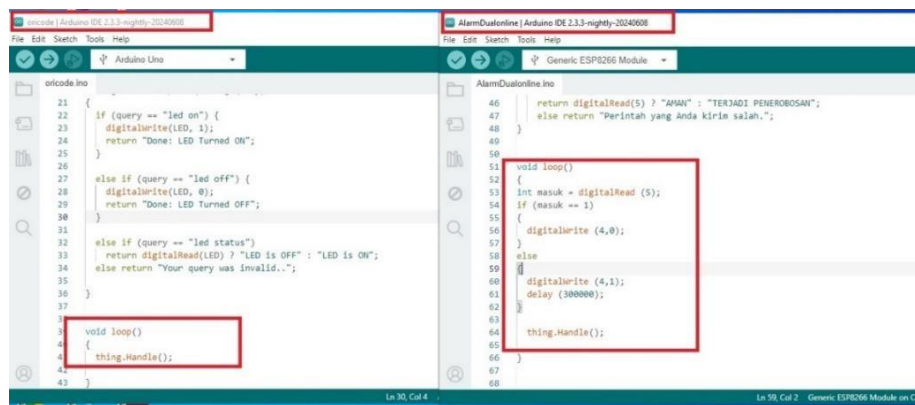


Gambar 3.25 Perubahan *query* yang akan dikirim dan diterima melalui WhatsApp

4. Perubahan/penambahan instruksi utama

Bagian *void loop* dapat dikatakan sebagai perubahan paling signifikan, yang merubah kode awal sebagai instruksi menyalakan atau mematikan lampu, sepenuhnya menjadi kode program untuk perangkat alarm dengan modul *reed* sebagai sensor. Tidak banyak yang penulis masukkan pada bagian *void loop*, hanya memasukkan nilai integer 1 ketika *reed*/pintu tertutup (yang berarti nilai integer = 0 dalam keadaan terbuka), dan pernyataan logika; jika *reed* tertutup (1) maka *buzzer* tidak berbunyi (0), namun selain dari logika tersebut maka yang terjadi adalah kebalikannya. Dengan memasukkan nilai *delay* sebesar 300000, *buzzer* akan tetap berbunyi selama 3 (tiga) menit—dan dapat disesuaikan ulang oleh pengguna—setelah *reed* terbuka. ‘Menahan’ *delay* pada jeda 3 (tiga) menit setelah *reed* terbuka,

berarti kendatipun *reed*/pintu ditutup kembali—dalam keadaan setelah dibuka tanpa menonaktifkan alarm terlebih dahulu—tidak akan menghentikan bunyi yang berasal dari *buzzer*. Dalam keadaan seperti ini, satu-satunya cara—prosedural—menonaktifkan alarm adalah melalui pemindaian sidik jari yang telah terdaftar pada blok induk.



Gambar 3.26 Penambahan *void loop*

Kode program blok alarm selesai dibuat, setelah mengunggah kode ke papan kendali NodeMCU ESP8266 Lolin, penulis melakukan beberapa percobaan dengan mengirimkan beberapa *query*. Hasilnya tampak sebagaimana gambar-gambar berikut:



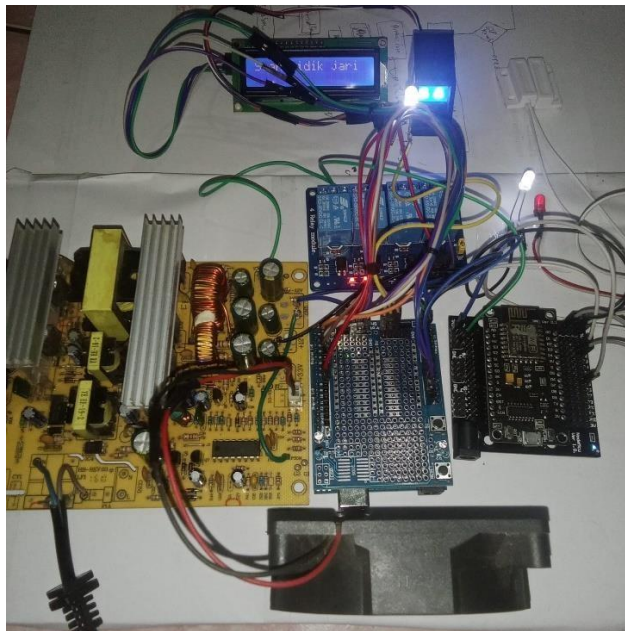
Gambar 3.27 Uji coba dengan *query* “status”

3.6.5 Rangkaian Keseluruhan Perangkat

Rangkaian *final* perangkat (keras dan lunak) alarm mengalami beberapa perubahan dari rangkaian awal pada piranti keras. Dua hal yang paling mencolok adalah;

1. Jalur asupan daya tidak lagi melalui *port* DC Input, PSU dapat mengalir daya dengan sangat stabil. Kedua papan kendali diasupi daya melalui *pin* VIN. Penggunaan kabel dapat dikurangi, *port* di bagian belakang kedua papan kendali tetap terbuka agar penyesuaian ulang dapat dilakukan tanpa mengeluarkan papan kendali dari *case* perangkat.
2. Penambahan *exhaust fan* guna mengurangi/membuang temperatur tinggi yang berasal dari regulator PSU dan papan kendali Arduino Uno.

Setelah ditanamkan instruksi pada setiap papan kendali, perangkat alarm siap untuk ditempatkan pada *casing* perangkat, sebagaimana tampak pada gambar berikut (pada gambar, tanpa mempengaruhi kinerja perangkat alarm untuk sementara *buzzer* pasif diganti dengan LED berwarna merah):



Gambar 3.28 Rangkaian keseluruhan perangkat alarm

BAB IV

PEMBAHASAN

4.1 *Device Starts*

Sebelum pengujian perangkat, telah dilakukan input sidik jari pengguna—dalam hal ini, penulis—yang memiliki *privilege* ke kendali blok induk. Prosedur menempatkan sidik jari di *database* blok induk dengan kode program sebagaimana dicantumkan dalam BAB III, selanjutnya mengikuti langkah-langkah arahan melalui jendela Serial Monitor pada program Arduino Uno.

Tabel 4.1 Kondisi perangkat setelah dinyalakan

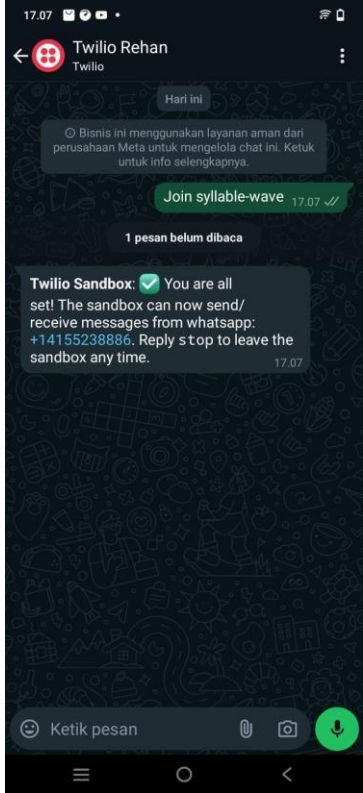
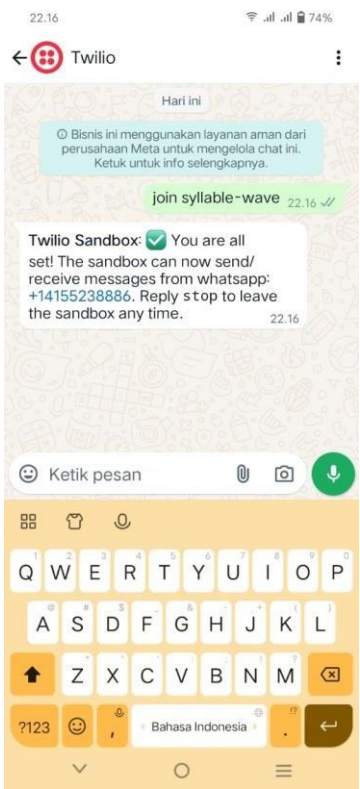
Blok	Selang waktu	Keterangan
Induk	30 detik	<i>Standby</i>
Alarm	30~32 detik	Blok alarm terhubung ke <i>router/AP/modem/Hotspot</i>

Selang waktu 30 detik sebelum blok induk berada pada keadaan *standby* ditujukan untuk memberi kesempatan kepada pengguna melakukan hal-hal seperti menutup dan mengunci pintu, dan lain sebagainya, sebelum mode alarm benar-benar aktif. Blok induk masuk ke fase *standby* ditandai dengan lampu indikator (MAIN) pada panel pemindai sidik jari menyala.

Blok alarm terhubung ke jaringan internet dalam rentang waktu relatif. Faktor-faktor yang mungkin berpengaruh, seperti jarak dan perintang antara perangkat alarm dan AP ataupun *router*, kualitas perangkat, dan sebagainya. Kondisi terhubung ditandai dengan lampu indikator (CONNECTED) pada panel pemindai sidik jari menyala. Ketika blok alarm terhubung dengan perangkat internet (seperti modem atau *router*), fungsi alarm belum benar-benar aktif jika kualitas sambungan internet buruk atau tidak ada sama sekali. Untuk memastikan hal ini, dapat dilakukan pemeriksaan melalui WhatsApp dengan mengirimkan pesan “status” ke Twilio.

4.2 Menambah Pengguna (Akses Kontrol dan Monitor via WhatsApp)

Tabel 4.2 Request layanan Twilio Sandbox

<i>Handphone 2</i>	<i>Handphone 3</i>
 The screenshot shows a WhatsApp chat interface on a dark theme. At the top, the contact is 'Twilio Rehan' with a Twilio logo. A system message in Indonesian states: 'Bisnis ini menggunakan layanan aman dari perusahaan Meta untuk mengelola chat ini. Ketuk untuk info selengkapnya.' Below this is a green message from 'Join syllable-wave' at 17.07. A status bar indicates '1 pesan belum dibaca'. The main message from 'Twilio Sandbox' at 17.07 says: 'You are all set! The sandbox can now send/receive messages from whatsapp: +14155238886. Reply stop to leave the sandbox any time.' The bottom shows a text input field with 'Ketik pesan' and a keyboard.	 The screenshot shows a WhatsApp chat interface on a light theme. At the top, the contact is 'Twilio' with a Twilio logo. A system message in Indonesian is the same as in Handphone 2. Below it is a green message from 'Join syllable-wave' at 22.16. The main message from 'Twilio Sandbox' at 22.16 says: 'You are all set! The sandbox can now send/receive messages from whatsapp: +14155238886. Reply stop to leave the sandbox any time.' The bottom shows a text input field with 'Ketik pesan' and a keyboard.

4.3 Kontrol dan Monitor melalui WhatsApp

Pada situasi sebenarnya, teks yang dikirimkan tanpa menggunakan tanda petik (“), begitupun dengan pesan balasan

Tabel 4.3 Kendali dan pemantauan melalui pesan WhatsApp

Pesan dikirim	Balasan	Status alarm	Keterangan
“Aman”	“MODE KEAMANAN”	Aktif	Alarm: <i>enabled</i>
“Izin”	“AKSES BEBAS”	Aktif	Alarm: <i>disabled</i>
“Status”	“AMAN”	Aktif	<i>Enabled/disabled</i>
	“Your device is offline..”	Tidak aktif/aktif	Kemungkinan: 1. Perangkat alarm tidak hidup. 2. Perangkat alarm hidup, namun blok alarm mengalami gangguan, seperti koneksi internet yang buruk atau tidak ada sama sekali.
“ ”	“Perintah yang Anda kirim salah.”	Aktif/tidak aktif	-

4.4 Respon Sensor

Tabel 4.4 Respon sensor

Sensor	Respon (<i>input</i>)	Output
<i>Reed</i>	Kedua bilah <i>reed sensor</i> terpisah ± 3 mm	<i>Buzzer</i> aktif dan pasif berbunyi
Pemindai sidik jari (sidik jari 1)	$\pm 0,5$ detik ~	Blok induk memutus aliran daya ke blok alarm. Alarm nonaktif.
Pemindai sidik jari (sidik jari 2)	± 1 detik ~	Blok induk memutus aliran daya ke blok alarm. Alarm nonaktif.
Pemindai sidik jari (sidik jari 3)	± 1 detik ~	Blok induk memutus aliran daya ke blok alarm. Alarm nonaktif.

Waktu mutlak pemindaian sidik jari menjadi tidak dapat dipastikan. Kondisi ini mungkin muncul dipengaruhi beberapa faktor, seperti jari yang digunakan untuk identifikasi tidak berada dalam keadaan bersih, benda-benda (seperti debu, minyak, dan sebagainya) yang berada pada *surface* pemindai sehingga *scanner* menjadi kurang sensitif, dan sebagainya. Faktor-faktor non teknis menjadi penyebab waktu respon sensor pemindai untuk sidik jari dan waktu pemindaian yang berbeda menjadi berbeda pula. Jika identitas sidik jari dikenali (telah tersimpan di pangkalan data), modul LCD I2C menampilkan teks “DITERIMA” dan lampu indikator (CONNECTED) pada panel sidik jari padam.

4.5 Kontrol dan Monitor via WhatsApp saat Terjadi Penerobosan

Tabel 4.5 Pengiriman perintah dan pemantauan saat terjadi penerobosan

Kondisi perangkat	Pesan dikirim	Balasan
Bilah sensor magnet terpisah, <i>buzzers</i> berbunyi.	“status”/”izin”/”aman”/”....”	-

4.6 Analisis

Sebagai perangkat berbasis *Internet of Things*, alarm hanya aktif saat terhubung ke internet, tidak tersedia fitur alarm moda luring. Sensor *reed* sebagai modul *input* pemicu tanda peringatan jika terjadi bahaya ditancapkan hanya di papan kendali NodeMCU ESP8266. Kendati demikian, intervensi manual tetap dibutuhkan pada situasi “darurat”, yaitu akses melalui pemindaian sidik jari yang mengharuskan pengguna—yang memiliki otoritas—berhadapan langsung dengan perangkat alarm.

Instruksi yang ditanamkan ke kedua *chip* papan kendali berjalan dengan sangat baik, sesuai dengan yang diharapkan. Khusus untuk blok alarm, kode asli (sebagaimana disebutkan pada BAB III) untuk menyalakan LED dengan perintah melalui WhatsApp dapat dirubah menjadi kode program perangkat alarm berbasis IoT cukup dengan menambahkan beberapa baris statemen logika pada bagian *void loop*.

Pada blok induk, papan kendali Arduino Uno sebenarnya hanya “menguasai” modul *relay* yang difungsikan sebagai *automatic switch*. Penonaktifan alarm artinya adalah protokol pemutusan tegangan masuk ke blok alarm melalui *relay* oleh blok induk. *Reactivation* blok alarm tidak memiliki prosedur lain selain melalui waktu selang yang dibatasi oleh kendali Arduino Uno. Hal ini sebenarnya merupakan salah satu kekurangan dari perangkat alarm, namun sangat mungkin untuk diabaikan mengingat akses kendali dengan sidik jari ditujukan hanya untuk situasi-situasi yang tidak diharapkan.

Dua hal yang penulis akui masih sangat perlu untuk diperbaiki dari perangkat alarm;

1. ketiadaan notifikasi melalui WhatsApp jika terjadi penerobosan, dan
2. ketika terjadi penerobosan, tidak dapat dilakukan pengendalian ataupun pemantauan melalui aplikasi WhatsApp.

Untuk kekurangan pertama, hal tersebut disebabkan oleh model layanan *cloud*—yaitu Twilio Sandbox—yang hanya memberikan *feedback* ke WhatsApp pengguna

berdasarkan pesan yang dikirimkan. Kekurangan ini dapat diatasi dengan memanfaatkan WhatsApp *bot* seperti api.callmebot.com salah satunya. Hanya saja, tidak memungkinkan untuk menggunakan dua layanan API sekaligus untuk satu perangkat, penulis dihadapkan pilihan antara notifikasi atau kontrol dan pemantauan. Jikapun menggunakan *platform* IoT semisal *Blynk IoT* yang memungkinkan untuk mnampung keduanya, kemudahan penggunaan WhatsApp menjadi opsi terbaik jika perangkat ditujukan agar mudah digunakan oleh banyak pengguna, misal oleh seluruh anggota dalam satu keluarga yang masing-masing memiliki *handphone*, dan secara lazim telah akrab dengan WA sebagai media perpesanan.

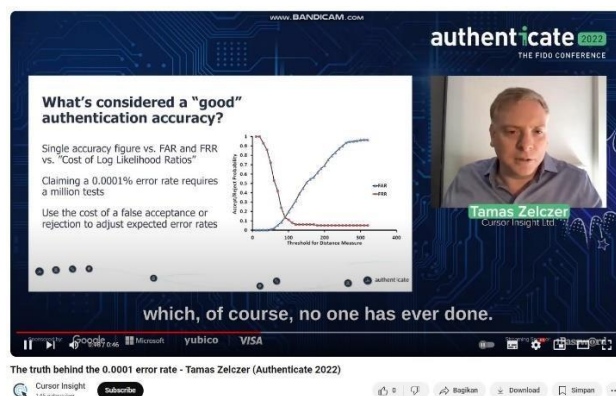
Masalah ke-dua—yang penulis ketahui setelah perangkat rampung dan diuji coba—adalah disebabkan oleh kekurangan instruksi pada kode program blok alarm yang menyebabkan variabel waktu **delay ()** yang seharusnya hanya untuk *buzzer* juga berdampak pada permintaan status ke Sandbox. Satu-satunya cara mengetahui sedang terjadi penerobosan adalah dengan ketiadaan balasan dari Twilio selama waktu yang sama dengan berbunyinya *buzzer* di mana alarm ditempatkan ketika pengguna mengirimkan permintaan status.

Layanan Sandbox dari Twilio yang digunakan pada Tugas Laporan Akhir ini adalah layanan gratis. Karena merupakan layanan gratis, pemakaian layanan hanya berlaku selama 72 jam setelah mendapat *approve* dari Twilio untuk bergabung menggunakan perangkat. Namun, pengguna tidak perlu khawatir akan kehilangan manfaat dari layanan tersebut. Setelah masa pakai (72 jam) habis, pengguna cukup mengirimkan *request* dengan kode yang sama—sebagaimana permintaan sebelumnya—agar kembali memiliki hak akses atas perangkat selama 72 jam ke depan.

4.7 Analisis FRR dan FAR

Sebagai *disclaimer*, penulis tidak memiliki keterangan tentang prosedur kalibrasi model autentikasi pihak produsen modul sensor AS608 dalam menetapkan nilai FRR dan FAR beserta *threshold* yang ditetapkan dalam penentuan nilai. Mengacu kepada spesifikasi teknis dan *datasheet* yang didapat penulis, nilai *False Rejection Rate* (FRR) dan *False Acceptance Rate* (FRR) merupakan deskripsi yang paling berkaitan dengan tingkat akurasi dan kepresisian sistem autentikasi biometrik modul sensor sidik jari AS608. Pada *datasheet* dicantumkan nilai $<1,0\%$ untuk FRR, sedangkan FAR bernilai $<0,001\%$. Berdasarkan deskripsi tersebut, modul memiliki akurasi tinggi dan mumpuni untuk meng-*handle* proses autentikasi keamanan pada level yang cukup krusial.

FRR adalah kemungkinan pengguna yang sah ditolak oleh sistem, sedangkan FAR adalah kemungkinan identitas palsu (*impostor*) akan diterima oleh sistem. Jika mengacu kepada *datasheet* modul AS608, dapat diterangkan bahwa sistem akan menolak 1 dari 100 upaya autentikasi yang sah dan menerima hanya 1 dari 10.000 percobaan autentikasi *impostor*. Nilai tersebut pantas diragukan, mengingat untuk sampai ke nilai tersebut akan dibutuhkan *database* sangat besar, jutaan kali uji coba pemindaian yang bahkan melibatkan ratusan ribu identitas yang sama sekali belum dikenal oleh sistem. Uraian sederhana pendukung keraguan tersebut dapat disimak, salah satunya pada tautan <https://www.youtube.com/watch?v=xFbJLSjDs4Q>



Gambar 4.1 Tangkapan layar video *The truth behind the 0.0001 error rate*

Nilai dari data lapangan yang penulis dapatkan sangat berbeda dari nilai FRR dan FAR pada *datasheet* AS608. Dengan perhitungan [39] sebagai berikut; *user* yang terdaftar sebanyak 3 orang (User #1 hingga User #3), input identitas sidik jari dilakukan dengan 2 kali pemindaian untuk setiap pengguna. Total *image* pada pangkalan data sistem berarti sebanyak 6 *image*. Satu pengguna, katakanlah User #1 memiliki 1 templat dan 1 *image* sebagai verifikasi. Dengan demikian, setiap pengguna memiliki nilai asli (*genuine scores*) 1 dan secara total ada 3 nilai asli, secara keseluruhan nilai palsu (*impostor scores*) 4.

Sistem autentikasi biometrik memberikan skor antara 0 hingga 1 pada proses autentikasi, yang berarti skor 0 pasti tertolak dan skor 1 pasti diterima. Dengan menetapkan ambang (misal 0,7), upaya autentikasi yang berada pada nilai 0,7 hingga 1 dapat diterima, sedangkan sisanya tertolak. Pada perangkat yang dirancang dan dibangun penulis, asumsi *threshold* 0,7 dengan 3 *users* yang sidik jarinya telah terekam pada pangkalan data sistem. Jika 1 dari skor asli berada di bawah *threshold*, maka FRR sistem berarti $1/3 = 33,3\%$. Untuk FAR, dengan 6 kali pemindaian dari subjek yang tidak dikenal, dimiliki 24 *impostor scores*. Dengan asumsi bahwa 3 pengguna yang sidik jarinya telah terekam berada di atas ambang batas, maka didapatkan FAR adalah $3/24=12,5\%$.

Sebagai simpulan analisis dari hasil di atas, *gap* nilai yang cukup jauh tersebut disebabkan (paling tidak) oleh 2 faktor:

1. Sangat sedikit objek identifikasi dan verifikasi pada *database*, dan
2. Aplikasi Adafruit Fingerprint Sensor pada *library* program Arduino IDE yang hanya “meminta” 2 kali pemindaian pada proses input sidik jari pengguna, sehingga hanya didapatkan 1 *genuine score* untuk setiap pengguna.

Perhitungan yang menghasilkan nilai FRR dan FAR yang lebih kecil dengan menggunakan pangkalan data lebih besar pada saat kalibrasi autentikasi sistem sangat berpengaruh.

BAB V

PENUTUP

5.1 Kesimpulan

1. Selain sebagai pemberi tanda jika terjadi penerobosan terhadap pintu atau pembatas keamanan lainnya, perangkat alarm juga ditujukan untuk mengganggu konsentrasi atau mengalihkan perhatian pelaku penerobosan. Sebagai pemberi tanda peringatan, alarm mengeluarkan bunyi khusus yang dihasilkan oleh *buzzer* (aktif dan pasif) ketika kedua bilah sensor *reed* terpisah atau pintu dibuka tanpa melalui prosedur sebagaimana mestinya.
2. Ketidakmampuan perangkat mengirim notifikasi *online* jika terjadi penerobosan dapat diperbaiki dengan melakukan penambahan atau penggantian instruksi melalui pengkodean (ulang) untuk blok alarm (NodeMCU ESP8266). Meskipun hal ini mungkin saja akan mengakibatkan beberapa fitur yang telah tertanam sebelumnya menjadi hilang atau tidak berfungsi dengan baik.
3. Penonaktifan alarm saat terjadi penerobosan melalui pemindaian sidik jari, di satu sisi merupakan kekurangan, karena sejatinya moda alarm dikendalikan dan dipantau secara *online* melalui aplikasi WhatsApp. Namun merupakan keuntungan tersendiri yang sebelumnya telah direncanakan agar jika terjadi situasi yang tidak diharapkan, hanya pengguna dengan *privilege* saja yang memiliki akses kendali tersebut. Hal tersebut merupakan keuntungan, karena identitas sidik jari merupakan sesuatu yang sangat unik yang nyaris mustahil untuk dipalsukan, terlupa, atau dibajak.

5.2 Saran

1. Perbaikan/pengembangan lebih lanjut masih sangat terbuka. Dari sisi *hardware*, mungkin untuk ditambahkan fitur lain, seperti strobo, rekaman suara dengan menggunakan modul DF Player, dan lain sebagainya. Begitupun *housing* komponen-komponen perangkat, dapat dipertimbangkan untuk menggunakan bahan-bahan yang lebih tahan terhadap benturan. *Mounting* panel sidik jari, *placing* blok induk dan alarm sangat dianjurkan untuk berada pada posisi aman, terutama dengan pertimbangan jika terjadi penerobosan, pelaku tidak dapat merusak perangkat alarm.
2. Dari sisi piranti lunak, sangat diperlukan pengembangan instruksi notifikasi daring jika terjadi penerobosan.

DAFTAR PUSTAKA

- [1] Septryanti, A., & Permana, E. S. (2020). Pengaman Pintu Rumah Berbasis Sensor Sidik Jari Dan Magnetic Sensor. *CESS (Journal of Computer Engineering, System and Science)*, 5(2), 305.
<https://doi.org/10.24114/cess.v5i2.18061>.
- [2] kbbi. (t.th). *alarm*. Kamus Besar Bahasa Indonesia (KBBI). Diakses pada 1 Mei 2024, sumber dari: <https://kbbi.web.id/alarm>.
- [3] Blog. (2024). *Pentingnya Sensor Pintu dan Jendela dalam Keamanan Rumah*. SECOM. Diakses pada 1 Mei 2024, sumber dari: <https://secom.co.id/id/pentingnya-sensor-pintu-dan-jendela-dalam-keamanan-rumah/>.
- [4] https://diskominfo.badungkab.go.id/storage/olds/diskominfo/Pentingnya-Internet-of-Things-IOT_548530.jpg
- [5] Efendi, Y. (2018). Internet Of Things (Iot) Sistem Pengendalian Lampu Menggunakan Raspberry Pi Berbasis Mobile. *Jurnal Ilmiah Ilmu Komputer*, 4(2), 21–27. <https://doi.org/10.35329/jiik.v4i2.41>
- [6] Adhelia-09011181621009. (2009). whitepaper Internet of Things. *Cyber Resilience of Systems and Networks*, 2019(July 2016), 1–150.
http://link.springer.com/10.1007/978-3-319-77492-3_16.
- [7] Wilianto, & Kurniawan, A. (2018). Sejarah , Cara Kerja Dan Manfaat Internet of Things. *Matrix*, 8(2), 36–41.
- [8] Gillis, Alexander S. 2023. *Internet of Things (IoT)*. Diakses pada 10 Februari 2024 dari: <https://www.techtarget.com/iotagenda/definition/Internet-of-Things-IoT>
- [9] Prastyo, E. A. (2018). *Arduino UNO ATmega328P*. ARDUINO INDONESIA. <https://www.arduinoindonesia.id/2022/08/pengertian-dan-penjelasan-arduino-uno.html>

- [10] <https://m.media-amazon.com/images/I/71z22cRPeeL.jpg>
- [11] Oreki. (2023). *Spesifikasi Arduino R3: Pengertian Dasar, Spesifikasi Hardware, Wiring Mapping Pin Out, Perangkat Lunak, Program, dan Fitur*. TUTORIAL OTOMASI INDONESIA. Diakses pada 5 Mei 2024, sumber dari: <https://www.tutorial-otomasi.com/2023/09/spesifikasi-arduino-uno-r3-pengertian.html>
- [12] Prastyo, Elga Aris. (2024). *Ulasan Lengkap: Arduino Uno – Kelebihan dan Kekurangannya*. ARDUINO INDONESIA. Diakses pada 5 Mei 2024, sumber dari: <https://www.arduinoindonesia.id/2024/06/ulasan-lengkap-arduino-uno-kelebihan-dan-kekurangannya.html>
- [13] Manual, P. R. (1972). Target Areas. *The Lancet*, 300(7770), 222. [https://doi.org/10.1016/S0140-6736\(72\)91649-2](https://doi.org/10.1016/S0140-6736(72)91649-2)
- [14] <https://components101.com/microcontrollers/arduino-uno>
- [15] Adityaeka26. (2019). *Cara Install Arduino IDE untuk ESP8266*. IoT STUDIO. Diakses pada 13 Mei 2024, sumber dari: <https://iotstudio.labs.telkomuniversity.ac.id/cara-install-arduino-ide-untuk-esp8266/>
- [16] <https://iotstudio.labs.telkomuniversity.ac.id/cara-install-arduino-ide-untuk-esp8266/>
- [17] Dahoud, A. Al, & Fezari, M. (2018). NodeMCU V3 For Fast IoT Application Development. *Notes*, October, 5.
- [18] Started Getting Guide User. (2018). *Handson Technology User Manual V1.3 ESP8266 NodeMCU WiFi Development Board Getting Started User Guide*. 1–24. www.handsontec.com
- [19] <https://www.theengineeringprojects.com/2018/08/esp8266-pinout-datasheet-features-applications.html>
- [20] Alone, S., Capstone, O., Model, C., & C, C. (2006). *Technical Reference*

ESP8266. February, 1–22.

- [21] Sihotang, Amril Hakim. (2024). *Memahami Esensi Bahasa Pemrograman C++*. DICODING. Diakses pada 10 Mei 2024, sumber dari:
<https://www.dicoding.com/blog/memahami-esensi-bahasa-pemrograman-c/>
- [22]
<https://rajguruelectronics.com/ProductView?tokDatRef=MjU0&tokenId=NTY=&product=AS608%20Optical%20fingerprint%20reader%20module%20sensor>
- [23] indiamart.com. (n.d.). AS608 Optical Fingerprint Sensor Fingerprint Module. *Indiamart.Com*, 12. <https://www.indiamart.com/proddetail/as608-optical-fingerprint-sensor-fingerprint-module-2852135460062.html>
- [24] <https://www.sinauprogramming.com/2020/10/menampilkan-text-pada-lcd-16x2-arduino.html>
- [25] Roghib, Muh. (2018). *Program LCD I2C*. MENARA ILMU MIKROKONTROLLER UNIVERSITAS GADJAH MADA. Diakses pada 14 Mei 2024, sumber dari:
<https://mikrokontroler.mipa.ugm.ac.id/2018/10/02/program-lcd-i2c/>
- [26] CNC Store Bandung. (t.th). *LCD Character 16x2/1602/16x02/1602A*. CNC STORE BANDUNG. Diakses pada 26 Mei 2024, sumber dari:
<https://cncstorebandunggo.blogspot.com/2019/08/lcd-character-16x2160216x021602a.html>
- [27] <https://shopee.co.id/MC-38-Sensor-Pintu-Jendela-Door-Window-Magnetic-Reed-Switch-MC38-i.390846440.24204938291>
- [28] Zainal, Nur Roin, Widyanto Hadi, Samsul Bachri M. (2018). *Pengaruh Posisi Sudut Optimum Reed Switch pada Motor Brushless DC Axial Flux*. JAEI. Hal. 10.
- [29] Almira, Nafisa Tania. (2023). *Rancang Bangun Perangkat Lunak Robot*

Kontrol Penyemprot Pestisida Berbasis Internet of Things dan Joystick di Lahan Perkebunan. Palembang. Politeknik Negeri Sriwijaya.

- [30] <https://digiwarestore.com/id/relay/relay-module-4-channel-5v-high-and-low-trigger.html>
- [31] IoT Virtualab. (t.th). *Buzzer*. VIRTUALAB IOT. Diakses pada 1 Mei 2024, sumber dari: <https://te.eng.uho.ac.id/virtualab/manager/buzzer.html>
- [32] https://www.tokopedia.com/wiksatech/sfm-27-buzzer-speaker-active-continuous-alarm-dc-3-24v?utm_source=google&utm_medium=organic&utm_campaign=pdp-seo
- [33] Sumber: <https://shopee.co.id/Passive-Buzzer-Speaker-Buzer-Pasive-5V-for-Arduino-Pasif-i.906610.1739344332>
- [34] Twilio. (2024). *What Exactly is Twilio*. TWILIO. Diakses pada 26 Mei 2024, sumber dari: <https://www.twilio.com/en-us/resource-center/what-is-twilio-an-introduction-to-the-leading-customer-engagement-platform>
- [35] Sumber: <https://www.twilio.com/en-us>
- [36] Drozdov, Alex. (2024). *How Twilio Works: An Overview*. YELLOW. Diakses pada 20 Mei 2024, sumber dari: <https://yellow.systems/blog/what-is-twilio-how-does-it-work>
- [37] <https://hybrid.co.id/post/apa-itu-whatsapp/>
- [38] Winarso, Bambang. (2015). *Apa Itu WhatsApp, Sejarah dan Fitur-fitur Unggulannya*. HYBRID. Diakses pada 16 Mei 2024, sumber dari: <https://hybrid.co.id/post/apa-itu-whatsapp/>
- [39] Lihat <https://stackoverflow.com/questions/30503766/how-can-i-calculate-the-failed-acceptance-rate-and-false-recognition-rate>. Diakses pada 6 September 2024.

Lampiran 1

Kode program input sidik jari

```
#include <LiquidCrystal_I2C.h>
#include <Adafruit_Fingerprint.h>

LiquidCrystal_I2C dis(0x27, 16, 2);
SoftwareSerial mySerial(2, 3);
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);

uint8_t id;

void setup() {
  Serial.begin(9600);
  dis.init();
  dis.backlight();
  dis.setCursor(0, 0);
  dis.print("Place finger");
  while (!Serial);
  delay(100);
  Serial.println("\n\nAdafruit Fingerprint sensor enrollment");
  finger.begin(57600);

  if (finger.verifyPassword()) {
    Serial.println("Found fingerprint sensor!");
  } else {
    Serial.println("Did not find fingerprint sensor :(");
    while (1) {
      delay(1);
    }
  }
  finger.getParameters();
}

uint8_t readnumber(void) {
```

```

uint8_t num = 0;
while (num == 0) {
    while (! Serial.available());
    num = Serial.parseInt();
}
return num;
}

void loop() {
    Serial.println("Ready to enroll a fingerprint!");
    Serial.println("Please type in the ID # (from 1 to 127) you want to save this finger as...");
    id = readnumber();
    if (id == 0) {
        return;
    }
    Serial.print("Enrolling ID #");
    Serial.println(id);

    while (! getFingerprintEnroll() );
}

uint8_t getFingerprintEnroll() {
    int p = -1;
    Serial.print("Waiting for valid finger to enroll as #"); Serial.println(id);
    while (p != FINGERPRINT_OK) {
        p = finger.getImage();
        switch (p) {
            case FINGERPRINT_OK:
                Serial.println("Image taken");
                break;
            case FINGERPRINT_NOFINGER:
                //Serial.println(".");
                break;
            case FINGERPRINT_PACKETRECEIVEERR:
                Serial.println("Communication error");

```

```

        break;
    case FINGERPRINT_IMAGEFAIL:
        Serial.println("Imaging error");
        break;
    default:
        Serial.println("Unknown error");
        break;
    }
}

p = finger.image2Tz(1);
switch (p) {
    case FINGERPRINT_OK:
        Serial.println("Image converted");
        break;
    case FINGERPRINT_IMAGEMESS:
        Serial.println("Image too messy");
        return p;
    case FINGERPRINT_PACKETRECEIVEERR:
        Serial.println("Communication error");
        return p;
    case FINGERPRINT_FEATUREFAIL:
        Serial.println("Could not find fingerprint features");
        return p;
    case FINGERPRINT_INVALIDIMAGE:
        Serial.println("Could not find fingerprint features");
        return p;
    default:
        Serial.println("Unknown error");
        return p;
}

Serial.println("Remove finger");
delay(2000);

p = 0;
while (p != FINGERPRINT_NOFINGER) {

```

```

    p = finger.getImage();
}
Serial.print("ID "); Serial.println(id);
p = -1;
Serial.println("Place same finger again");
while (p != FINGERPRINT_OK) {
    p = finger.getImage();
    switch (p) {
        case FINGERPRINT_OK:
            Serial.println("Image taken");
            break;
        case FINGERPRINT_NOFINGER:
            //Serial.print(".");
            break;
        case FINGERPRINT_PACKETRECEIVEERR:
            Serial.println("Communication error");
            break;
        case FINGERPRINT_IMAGEFAIL:
            Serial.println("Imaging error");
            break;
        default:
            Serial.println("Unknown error");
            break;
    }
}
p = finger.image2Tz(2);
switch (p) {
    case FINGERPRINT_OK:
        Serial.println("Image converted");
        break;
    case FINGERPRINT_IMAGEMESS:
        Serial.println("Image too messy");
        return p;
    case FINGERPRINT_PACKETRECEIVEERR:

```

```

        Serial.println("Communication error");
        return p;
    case FINGERPRINT_FEATUREFAIL:
        Serial.println("Could not find fingerprint features");
        return p;
    case FINGERPRINT_INVALIDIMAGE:
        Serial.println("Could not find fingerprint features");
        return p;
    default:
        Serial.println("Unknown error");
        return p;
}

Serial.print("Creating model for #"); Serial.println(id);
p = finger.createModel();
if (p == FINGERPRINT_OK) {
    Serial.println("Prints matched!");
} else if (p == FINGERPRINT_PACKETRECEIVEERR) {
    Serial.println("Communication error");
    return p;
} else if (p == FINGERPRINT_ENROLLMISMATCH) {
    Serial.println("Fingerprints did not match");
    return p;
} else {
    Serial.println("Unknown error");
    return p;
}

Serial.print("ID "); Serial.println(id);
dis.setCursor(0, 1);
dis.print("ID: ");
dis.print(id);
p = finger.storeModel(id);
if (p == FINGERPRINT_OK) {
    Serial.println("Stored!");
    dis.print(" Stored!");
}

```

```
    }  
    else if (p == FINGERPRINT_PACKETRECEIVEERR) {  
        Serial.println("Communication error");  
        return p;  
    }  
    else if (p == FINGERPRINT_BADLOCATION) {  
        Serial.println("Could not store in that location");  
        return p;  
    }  
    else if (p == FINGERPRINT_FLASHERR) {  
        Serial.println("Error writing to flash");  
        return p;  
    }  
    else {  
        Serial.println("Unknown error");  
        return p;  
    }  
    return true;  
}
```

Lampiran 2

Kode Program Blok Induk

```
#include <LiquidCrystal_I2C.h>
#include <Adafruit_Fingerprint.h>
#define relay 4
#define led 8

SoftwareSerial mySerial(2, 3);
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);
LiquidCrystal_I2C dis(0x27, 16, 2);

byte ID = 1;
bool Switch = true;

void setup() {
  Serial.begin(9600);
  dis.init();
  dis.backlight();
  dis.setCursor (0,0);
  dis.print ("M Raichan");
  dis.setCursor (0,1);
  dis.print ("Muzakki");
  delay (30000);
  dis. clear ();

  dis.init();
  dis.backlight();
  dis.setCursor(0, 0);
  dis.print("Scan sidik jari");
  pinMode(relay, OUTPUT);
  pinMode (led, OUTPUT);
```

```

while (!Serial);
delay(100);
Serial.println("\n\nAdafruit finger detect test");

finger.begin(57600);
delay(5);
if (finger.verifyPassword()) {
  Serial.println("Found fingerprint sensor!");
} else {
  Serial.println("Did not find fingerprint sensor :(");
  while (1) {
    delay(1);
  }
}
finger.getParameters();
finger.getTemplateCount();
if (finger.templateCount == 0) {
  Serial.print("Sensor doesn't contain any fingerprint data. Please run the 'enroll' example.");
}
else {
  Serial.println("Waiting for valid finger...");
  Serial.print("Sensor contains "); Serial.print(finger.templateCount); Serial.println("
templates");
}
}

void loop() {
  int value = getFingerprintIDez();

  if (value == ID)
  {
    dis.setCursor(0, 1);

```

```

    dis.print("  DITERIMA  ");
    digitalWrite(relay, HIGH);
    delay (60000);
    Switch = false;
    digitalWrite(relay, LOW);
    Switch = true;
}
else
{
    dis.setCursor(0, 1);
    dis.print("  ");
}
}

int getFingerprintIDez() {
    uint8_t p = finger.getImage();
    if (p != FINGERPRINT_OK) return -1;
    p = finger.image2Tz();
    if (p != FINGERPRINT_OK) return -1;
    p = finger.fingerFastSearch();
    if (p != FINGERPRINT_OK) return -1;
    return finger.fingerID;
}

```

Lampiran 3

Kode Program Blok Alarm

```
#include <ThingESP.h>
#include <ESP8266wifi.h>
ThingESP8266 thing("raichanmuzakki", "AlarmDual", "muzakki");

#define reed = 5
#define buzzer = 4
#define buzzera = 2
#define led = 14

unsigned long previousMillis = 0;
const long INTERVAL = 6000;

void setup ()
{
  Serial.begin(115200);
  pinMode(5, OUTPUT);
  pinMode (4, OUTPUT);
  pinMode (2, OUTPUT);
  pinMode (14, OUTPUT);
  tone (4, 1000, 300);
  thing.SetWiFi("TLA", "1234567890");
  thing.initDevice();
  while(WiFi.status() !=WL_CONNECTED)
  {
    digitalWrite (14, 0);
    delay (500);
  }
  digitalWrite (14, 1);
}
```

```

String HandleResponse(String query)
{
    if (query == "izin") {
        digitalWrite(5, 1);
        digitalWrite (4, 0);
        return "AKSES BEBAS";
    }
    else if (query == "aman") {
        digitalWrite(5, 0);
        digitalWrite (4,0);
        return "MODE KEAMANAN";
    }
    else if (query == "status")
        return digitalRead(5) ? "AMAN" : "TERJADI PENEROBOSAN";
        else return "Perintah yang Anda kirim salah.";
    }
void loop()
{
    int masuk = digitalRead (5);
    if (masuk == 1)
    {
        noTone (4);
        digitalWrite (2, LOW);
    }
    else
    {
        tone (4, 720);
        digitalWrite (2, HIGH);
        delay (180000);
    }
    thing.Handle();
}

```


Lampiran 4

Kode asli kendali LED yang dimodifikasi sebagai program blok alarm

```
#include <ESP8266WiFi.h>
```

```
#include <ThingESP.h>
```

```
ThingESP8266 thing("Username", "ProjectName", "ProjectUserName");
```

```
int LED = D2;
```

```
unsigned long previousMillis = 0;
```

```
const long INTERVAL = 6000;
```

```
void setup()
```

```
{
```

```
  Serial.begin(115200);
```

```
  pinMode(LED, OUTPUT);
```

```
  thing.SetWiFi("WiFiUsername", "WiFiPassword");
```

```
  thing.initDevice();
```

```
}
```

```
String HandleResponse(String query)
```

```
{
```

```
  if (query == "led on") {
```

```
    digitalWrite(LED, 1);
```

```
    return "Done: LED Turned ON";
```

```
  }
```

```
  else if (query == "led off") {
```

```
    digitalWrite(LED, 0);
```

```
    return "Done: LED Turned OFF";  
}
```

```
else if (query == "led status")  
    return digitalRead(LED) ? "LED is OFF" : "LED is ON";  
else return "Your query was invalid..";
```

```
}
```

```
void loop()  
{  
    thing.Handle();  
  
}
```

	KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI	
	POLITEKNIK NEGERI SRIWIJAYA	
	JURUSAN TEKNIK ELEKTRO	
	Jalan Srijaya Negara, Palembang 30139	
	Telp. 0711-353414 Fax. 0711-355918	
Website : www.polisriwijaya.ac.id E-mail : info@polsri.ac.id		
KESEPAKATAN BIMBINGAN LAPORAN AKHIR (LA)		

Kami yang bertanda tangan di bawah ini,

Pihak Pertama

Nama : M Raichan Muzakki
NIM : 062130331093
Jurusan : Teknik Elektro
Program Studi : D3 Teknik Telekomunikasi

Pihak Kedua

Nama : Hj. Lindawati, ST, M.T.I
NIP : 187105282006042001
Jurusan : Teknik Elektro
Program Studi : Teknik Telekomunikasi

Pada hari ini Rabu tanggal 08 Mei 2024 telah sepakat untuk melakukan konsultasi bimbingan Laporan Akhir.

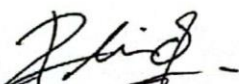
Konsultasi bimbingan sekurang-kurangnya 1 (satu) kali dalam satu minggu. Pelaksanaan bimbingan pada setiap hari Rabu pukul 14.00 tempat di Politeknik Negeri Sriwijaya.

Demikianlah kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Laporan Akhir.

Palembang, 8 Mei 2024

Pihak Pertama,

Pihak Kedua,


M Raichan Muzakki
NIM. 062130331093


Hj. Lindawati, ST, M.T.I
NIP.187105282006042001

Mengetahui,
Ketua Jurusan


Ir. Iskandar Lutfi, M.T.
NIP. 196501291991031002

	KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET DAN TEKNOLOGI	 
	POLITEKNIK NEGERI SRIWIJAYA	
	Jalan Srijaya Negara, Palembang 30139 Telp. 0711-353414 Fax. 0711-355918	
	Website : www.polisriwijaya.ac.id E-mail : info@polsri.ac.id	
KESEPAKATAN BIMBINGAN LAPORAN AKHIR (LA)		

Kami yang bertanda tangan di bawah ini,

Pihak Pertama

Nama : M Raichan Muzakki
NIM : 062130331093
Jurusan : Teknik Elektro
Program Studi : DIII Teknik Telekomunikasi

Pihak Kedua

Nama : Asriyadi, S.T., M.T.
NIP : 198404272015041000
Jurusan : Teknik Elektro
Program Studi : Teknik Telekomunikasi

Pada hari ini tanggal telah sepakat untuk melakukan konsultasi bimbingan Laporan Akhir. Konsultasi bimbingan sekurang-kurangnya 1 (satu) kali dalam satu minggu. Pelaksanaan bimbingan pada setiap hari pukul, tempat di Politeknik Negeri Sriwijaya. Demikianlah kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Laporan Akhir.

Palembang, Februari 2024

Pihak Pertama,



M Raichan Muzakki
NIM. 062130331093

Pihak Kedua,



Asriyadi, S.T., M.T.
NIP. 198404272015041000

Mengetahui,
Ketua Jurusan



Ir. Iskandar Lutfi, M.T.
NIP. 196501291991031002



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI SRIWIJAYA
JURUSAN TEKNIK ELEKTRO
 Jalan Sriwijaya Negara, Palembang 30139
 Telp. 0711-3534114 Fax. 0711-355918
 Website : www.polsriwajaya.ac.id E-mail : info@polsri.ac.id


LEMBAR BIMBINGAN LAPORAN AKHIR

Nama : M Raichan Muzakki

NIM : 062130331093

Jurusan/Program : Teknik Elektro / DIII Teknik Telekomunikasi

Studi

Judul Laporan Akhir : Rancang Bangun Alarm Dengan Akses Sidik Jari dan Aplikasi Smartphone Berbasis Internet of things (IoT) menggunakan Arduino Uno DAN Nodemcu ESP8266

Pembimbing I : Asriyadi, S.T., MT

No.	Tanggal	Uraian Bimbingan	Tanda Tangan Pembimbing
1.			
2.			
3.			
4.			
5.			

No.	Tanggal	Uraian Bimbingan	Tanda Tangan Pembimbing
8	1-7-2024	Acc BAB 3	
9	3-7-2024	Revisi Bab 4 & 5	
10	8-7-2024	Revisi BAB 4 & 5	
11	10-7-2024	Acc Bab 4 & 5	
12	11-7-2024		

Palembang, 12 Juli 2024

Koordinator Program Studi,

Ciksadan, S.T., M.Kom.
NIP 196809071993031003**Catatan:**

*) melingkari angka yang sesuai.

Ketua Jurusan/Ketua Program Studi harus memeriksa jumlah pelaksanaan bimbingan sesuai yang dipersyaratkan dalam Pedoman Laporan Akhir sebelum menandatangani lembar bimbingan ini.

Lembar pembimbingan LA ini harus dilampirkan dalam Laporan Akhir.



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN

POLITEKNIK NEGERI SRIWIJAYA

Jalan Sriwijaya Negara Palembang 30139

Telp. 0711-353414 fax 0711-355918

Website : www.polisriwijaya.ac.id E-mail : info@polisri.ac.id

REKOMENDASI UJIAN LAPORAN AKHIR (LA)

Pembimbing Laporan Akhir membenkan rekomendasi kepada,

Nama	M Raichan Muzakki
NIM	062130331093
Jurusan/Program Studi	Teknik Elektro/D3 Teknik Telekomunikasi
Judul Laporan Akhir	RANCANG BANGUN ALARM DENGAN AKSES SIDIK JARI DAN APLIKASI SMARTPHONEBERBASIS INTERNET OF THINGS (IoT)MENGUNAKAN ARDUINO UNO DAN NODEMCU ESP8266

Mahasiswa tersebut telah memenuhi persyaratan dan dapat mengikuti Ujian Laporan Akhir (LA) pada Tahun Akademik

Palembang, 12 Juli 2024

Pembimbing I,

(Hj. Lindawati, ST, M.T.I)
NIP 187105282006042001

Pembimbing II,

(Asriyadi, S.T., M.T)
NIP 198404272015041000





KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET DAN TEKNOLOGI
POLITEKNIK NEGERI SRIWIJAYA
 Jalan Srijaya Negara, Palembang 30139
 Telp. 0711-353414 Fax. 0711-355918
 Website : www.polisriwijaya.ac.id E-mail : info@polsri.ac.id
PELAKSANAAN REVISI LAPORAN AKHIR



Mahasiswa berikut,

Nama : M Raichan Muzakki
 NIM : 062130331093
 Jurusan/Program Studi : Teknik Elektro / DIII Teknik Telekomunikasi
 Judul Laporan Akhir : RANCANG BANGUN ALARM DENGAN AKSES SIDIK JARI DAN
 APLIKASI WHATSAPP BERBASIS INTERNET OF THINGS
 (IoT)MENGUNAKAN ARDUINO UNO DAN NODEMCU ESP8266

Telah melaksanakan revisi terhadap Laporan Akhir yang diujikan pada hari tanggal
 bulan tahun Pelaksanaan revisi terhadap Laporan Akhir tersebut telah disetujui
 oleh Dosen Penguji yang memberikan revisi:

No.	Komentar	Nama Dosen Penguji*)	Tanggal	Tanda Tangan
1	Sth di revisi	Ir. Abdul Rakhman, M.T.	19/8-24	
2	Sudah direvisi	Hj. Emilia Hesti, S.T., M.Kom	26/8-24	
3	Sudah di Acc	Sopian Soim, S.T., M.T.	12/10-2024	
4	Acc	Hj. Lindawati, S.T., M.T.I	17/9-24	

Palembang, Juli 2024

Ketua Penguji,

Ir. Abdul Rakhman, M.T.

NIP. 196006241990031002

Catatan:

- *) Dosen penguji yang memberikan revisi saat ujian laporan akhir.
 **) Dosen penguji yang ditugaskan sebagai Ketua Penguji saat ujian LA.
 Lembaran pelaksanaan revisi ini harus dilampirkan dalam Laporan Akhir.



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
POLITEKNIK NEGERI SRIWIJAYA
 Jalan Srijaya Negara, Palombang 30139
 Telp. 0711-353414 Fax. 0711-355918
 Website : www.polsriwijaya.ac.id E-mail : info@polsri.ac.id
LEMBAR BIMBINGAN LAPORAN AKHIR



Lombar . 1

Nama : M Raichan Muzakki
 NIM : 062130331093
 Jurusan/Program Studi : Teknik Elektro/D3 Teknik Telekomunikasi
 Judul Laporan Akhir : RANCANG BANGUN ALARM DENGAN AKSES SIDIK JARI DAN APLIKASI
 SMARTPHONEBERBASIS INTERNET OF THINGS (IoT)MENGUNAKAN
 ARDUINO UNO DAN NODEMCU ESP8266

Pembimbing I / II *) : Hj. Lindawati, ST, M.T.I/Asriyadi, S.T., M.T

No.	Tanggal	Uraian Bimbingan	Tanda Tangan Pembimbing
1.	30-5-2024	Revisi Bab 1	
2.	31-5-2024	Revisi Bab 1	
3.	12-6-2024	Acc, Bab 1, Revisi Bab 2	
4.	14-6-2024	Revisi Bab 2 & 3	
5.	16-6-2024	Revisi Bab 3, Acc Bab 2	
6.	25-6-2024	Revisi Bab 3	
7.	26-6-2024	Revisi Bab 3	

Lembar : 2

No.	Tanggal	Uraian Bimbingan	Tanda Tangan Pembimbing
8	1-7-2024	Acc Bab 3	
9	3-7-2024	Revisi Bab 4 & 5	
10	8-7-2024	Revisi Bab 4 & 5	
11	10-7-2024	Acc Bab 4 & 5	
12	11-7-2024	Rekomendasi ujian LA.	

Palembang, 8 Juli 2024

Ketua Jurusan/KPS,

Ciksadan, S.T., M.Kom
NIP.196809071993031003

Catatan:

*) melingkari angka yang sesuai.

Ketua Jurusan/Ketua Program Studi harus memeriksa jumlah pelaksanaan bimbingan sesuai yang dipersyaratkan dalam Pedoman Laporan Akhir sebelum menandatangani lembar bimbingan ini.

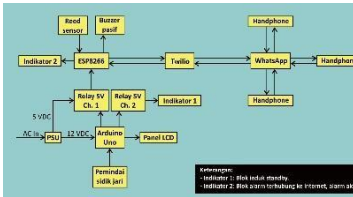
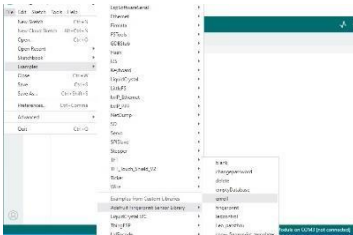
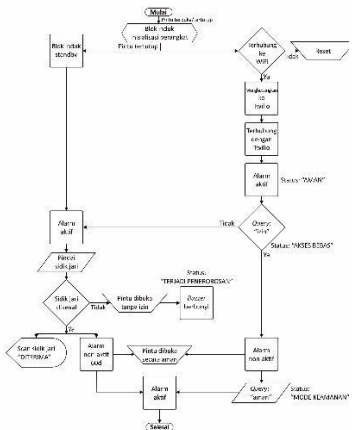
Lembar pembimbingan LA ini harus dilampirkan dalam Laporan Akhir.

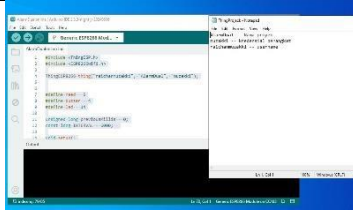
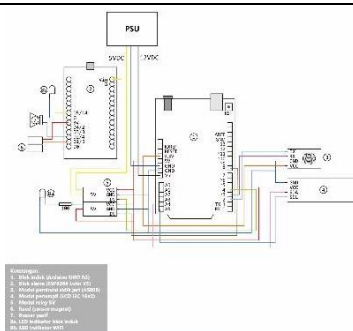
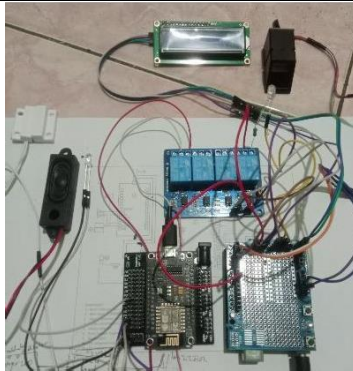
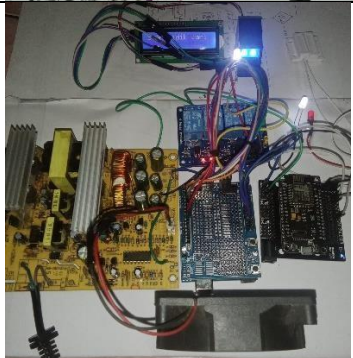
Lampiran 4

**LOGBOOK PEMBUATAN PERANGKAT UJI COBA RANCANG
BANGUN ALARM DENGAN AKSES SIDIK JARI DAN APLIKASI
WHATSAPP BERBASIS *INTERNET OF THINGS* MENGGUNAKAN
ARDUINO UNO R3 DAN NODEMCU ESP8266 V3**

Dosen Pembimbing I : Hj. Lindawati, ST, M.T.I

Dosen Pembimbing II :Asriyadi, S.T., M.T

No.	Waktu	Dokumentasi	Keterangan	Prese ntasi	TTD Pem. I	TTD Pem. II
1.	Minggu, 5 Mei 2024		Diagram Blok			
2.	Kamis, 9 Mei 2024		Skema Rangkaian			
3.	Senin, 27 Mei 2024		Flowchart			

4.	Sabtu, 1 Juni 2024		Pemrograman Perangkat			
5.	Minggu, 2 Juni 2024		Pemrograman Perangkat			
6.	Jumat, 7 Juni 2024		Pemrograman Perangkat			
7.	Jumat, 7 Juni 2024		Perangkat Lengkap Sebelum Koding			
8.	Senin, 24 Juni 2024		Perangkat Setelah Koding			

9.	Rabu, 26 Juni 2024		Perangkat sudah di housing			
10.	Kamis, 27 Juni 2024		Perakitan alat dan percobaan Alat			

