

SKRIPSI

**RANCANG BANGUN SISTEM PENGAMAN PINTU
BRANKAS MENGGUNAKAN FINGERPRINT DAN
KEYPAD BERBASIS ARDUINO**



POLITEKNIK NEGERI BALI

Oleh :

Gerald Mogo Kadena

NIM. 1815344054

**PROGRAM STUDI D4 TEKNIK OTOMASI
JURUSAN TEKNIK ELEKTRO
POLITEKNIK NEGERI BALI
2023**

LEMBAR PERSETUJUAN UJIAN SKRIPSI

**RANCANG BANGUN SISTEM PENGAMAN PINTU
BRANKAS MENGGUNAKAN FINGERPRINT DAN
KEYPAD BERBASIS ARDUINO**

Oleh :

Geraldly Mogo Kadena

NIM. 1815344054

Skripsi ini telah melalui Bimbingan dan Pengujian Hasil, disetujui untuk
diujikan pada Ujian Skripsi
di

Program Studi D4 Teknik Otomasi
Jurusan Teknik Elektro - Politeknik Negeri Bali

Bukit Jimbaran, 2022

Disetujui Oleh :

Dosen Pembimbing 1:



Ir. Kadek Amerta Yasa, ST., MT.
NIP. 196809121995121001

Dosen Pembimbing 2:



I Gede Suputra Widharma, ST., MT.
NIP. 197212271999031004

LEMBAR PENGESAHAN SKRIPSI

RANCANG BANGUN SISTEM PENGAMAN PINTU
BRANKAS MENGGUNAKAN FINGERPRINT DAN
KEYPAD BERBASIS ARDUINO

Oleh :

Geraldly Mogo Kadena
NIM. 1815344054


Skripsi ini sudah melalui Ujian Skripsi pada tanggal 01 September 2023,
dan sudah dilakukan Perbaikan untuk kemudian disahkan sebagai Skripsi
di


Program Studi D4 Teknik Otomasi
Jurusan Teknik Elektro - Politeknik Negeri Bali

Bukit Jimbaran, 2023


Disetujui Oleh :

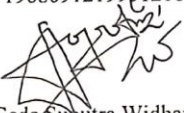
Tim Penguji :


1. Putri Alit Widyastuti Santuary, ST., MT
NIP. 197405172000122001


2. Dewa Ayu Indah Cahya Dewi, S.TI., MT.
NIP. 19911062020122005

Dosen Pembimbing :


1. Ir. Kadek Amerta Yasa, ST., MT.
NIP. 196809121995121001


2. I Gede Suputra Widharma, ST., MT.
NIP. 197212271999031004

Diketahui Oleh:

Ketua Jurusan Teknik Elektro


Ir. I Wayan Raka Ardana, MT.
NIP. 196705021993031005

HALAMAN PERNYATAAN KEASLIAN KARYA SKRIPSI

Saya yang bertanda tangan di bawah ini, menyatakan bahwa Skripsi dengan judul:

RANCANG BANGUN SISTEM PENGAMAN PINTU BRANKAS MENGUNAKAN FINGERPRINT DAN KEYPAD BERBASIS ARDUINO

adalah asli hasil karya saya sendiri.

Dengan ini saya menyatakan bahwa dalam naskah Skripsi ini tidak terdapat karya orang lain yang pernah diajukan untuk memperoleh gelar di suatu perguruan tinggi, dan atau sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah Skripsi ini, dan disebutkan dalam daftar pustaka.

Apabila saya melakukan hal tersebut di atas, dengan ini saya menyatakan menarik Skripsi yang saya ajukan sebagai hasil karya saya.

Bukit Jimbaran, 14 September 2023

Yang menyatakan



Geraldly Mogo Kadena

NIM. 1815344054

ABSTRAK

Pengamanan brankas menggunakan sistem penguncian, mekanik konvensional mudah dibobol dengan cara konvensional merupakan ilham untuk menciptakan sistem otomatisasi pengamanan brankas yang lebih teruji keandalannya. Perancangan otomatisasi pengamanan brankas dengan menggunakan kode password dan sidik jari berbasis mikrokontroler. Penguncian brankas yang diakses dengan masukan kode password dan sidik jari dengan mikrokontroler Arduino Wemos D1 sebagai pusat pengendali dan pengolah data. keypad sebagai media pemasuk password, LCD sebagai media penampil formasi, solenoid sebagai pembuka pintu brankas, dan power supply 12V, kesemuanya merupakan komponen pendukung rangkaian sistem. Alat ini dapat dibuka, apabila masukkan kode password dan sidik jari dengan benar Pada alat ini, Sistem bekerja setelah mendapat tegangan 12 volt. Sirkuit terhubung ke catu daya. Ketika sistem keamanan ini diaktifkan, layar pengaktifan akan muncul yang menyertakan frasa pengaktifan LCD. Sensor keamanan yang akan diaktifkan pertama kali adalah Fingerprint, Keypad, kemudian solenoid bekerja dengan perintah Relay, kemudian pintu brankas terbuka secara otomatis tanpa ada langkah yang ditentukan, kemudian buzzer akan langsung berbunyi dan LED akan menyala hijau, memberitahukan kepada pemilik. Sistem ini menggunakan aplikasi Blynk untuk mengecek siapa saja orang yang telah masuk sehingga meminimalisir tindak kejahatan yang dilakukan orang tidak dikenal. Penelitian ini telah berhasil merancang dan membangun sebuah sistem pengamanan ganda menggunakan *fingerprint* dan *keypad* pada pintu brankas yang secara keseluruhan sudah berfungsi dengan baik.

Kata Kunci: Pengaman brankas, keypad, LCD, sensor fingerprint, Arduino wemos D1

ABSTRACT

Safe security uses a locking system, conventional mechanics are easy to break into using conventional methods, which is an inspiration to create a safe security automation system that has more proven reliability. Safe security automation design using microcontroller-based password codes and fingerprints. Safe locking is accessed by entering a password code and fingerprint with the Arduino Wemos D1 microcontroller as the data processing and control center. keypad as a medium for entering passwords, LCD as a medium for displaying formations, solenoid as a safe door opener, and 12V power supply, all of which are supporting components for the system circuit. This device can be opened if you enter the password code and fingerprint correctly. In this device, the system works after receiving a 12 volt voltage. The circuit is connected to the power supply. When this security system is activated, a startup screen will appear that includes an LCD wake-up phrase. The security sensors that will be activated first are Fingerprint, Keypad, then the solenoid works with a Relay command, then the safe door opens automatically without any specified steps, then the buzzer will immediately sound and the LED will light green, notifying the owner. This system uses the Blynk application to check who has entered, thereby minimizing crimes committed by unknown people. This research has succeeded in designing and building a double security system using fingerprint and keypad on the safe door which overall functions well.

Keywords: Safe, keypad, LCD, fingerprint sensor, Arduino wemos D1

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa Pemurah dan Maha Penyayang, dengan limpah karunia-Nya Penulis dapat menyelesaikan penyusunan skripsi ini dengan judul Rancang Bangun Sistem Pengaman Pintu Brankas Menggunakan Fingerprint dan Keypad Berbasis Arduino.

Dalam penyusunan skripsi ini tidak terlepas dukungan dari berbagai pihak. Penulis secara khusus mengucapkan terima kasih yang sebesar-besarnya kepada semua pihak yang telah membantu. Penulis banyak menerima bimbingan, petunjuk dan bantuan serta dorongan dari berbagai pihak baik yang bersifat moral maupun material. Pada kesempatan ini penulis menyampaikan rasa terima kasih yang sebesar-besarnya kepada:

1. Tuhan Yang Maha Esa dengan segala rahmat serta karunia-Nya yang memberikan kekuatan bagi penulis dalam menyelesaikan skripsi ini.
2. Kepada kedua orang tua tercinta yang selama ini telah membantu penulis dalam bentuk perhatian, kasih sayang, semangat, serta doa yang tidak henti-hentinya mengalir demi kelancaran dan kesuksesan penulis dalam menyelesaikan skripsi ini.
3. Bapak I Nyoman Abdi, SE, M.eCom selaku Direktur Politeknik Negeri Bali
4. Bapak Ir. I Wayan Raka Ardana, M.T. selaku Ketua Jurusan Teknik Elektro Politeknik Negeri Bali.
5. Bapak IB. Irawan Purnama, ST, M.Sc, Ph.D. selaku Ketua Program Studi D IV Teknik Otomasi Politeknik Negeri Bali.
6. Bapak Ir. Kadek Amerta Yasa, ST., MT. selaku Dosen Pembimbing I Skripsi yang telah banyak memberikan bimbingan, arahan dan dukungan selama penyusunan skripsi ini..
7. Bapak I Gede Suputra Widharma, ST., MT. selaku Dosen Pembimbing II Skripsi yang telah banyak memberikan bimbingan, arahan dan dukungan selama penyusunan skripsi ini..

8. Segenap dosen dan seluruh staf Jurusan Teknik Elektro yang selalu membantu dalam memberikan fasilitas, ilmu, serta pendidikan hingga dapat menunjang dalam penyelesaian skripsi ini.
9. Sahabat dan teman - teman saya Yogi Koyoh, Yoyakim Oly, Adnan Yudha, Harmen Sorongan, Joshua Christo, Rizka Fergianti, Zefanya Agatha yang telah memberikan dukungan moral dan semangat.
10. Rekan seperjuangan D IV Teknik Otomasi Angkatan 2018, yang telah memberikan banyak masukan serta dukungan kepada saya.
11. Kepada Berliana Putri Alexander yang selama ini telah membantu penulis dalam bentuk perhatian, kasih sayang, semangat, serta doa yang tidak henti-hentinya mengalir demi kelancaran dan kesuksesan penulis dalam menyelesaikan skripsi ini.
12. Kepada CV. Pratama Cipta Waterproofing, yang telah memberikan banyak masukan serta dukungan kepada saya.
13. Serta masih banyak lagi pihak-pihak yang sangat berpengaruh dalam proses penyelesaian skripsi ini yang tidak bisa penulis sebutkan satu persatu

Skripsi ini saya persembahkan kepada Orang tua. Saya menyadari bahwa tidak ada sesuatu yang sempurna sehingga penulis mengharapkan kritik dan saran yang bersifat membangun. Pada kesempatan terakhir penulis berharap semoga tulisan ini dapat bermanfaat bagi pembaca. Semoga Tuhan Yang Maha Esa senantiasa membalas semua kebaikan yang telah diberikan.

Bukit Jimbaran, 27 Agustus 2022

Penulis

DAFTAR ISI

	Halaman
Halaman Persetujuan Ujian Skripsi.....	ii
Halaman Pengesahan Skripsi.....	iii
Halaman Pernyataan Keaslian Karya Skripsi.....	iv
Abstrak.....	v
Kata Pengantar.....	vii
Daftar Isi.....	ix
Daftar Gambar.....	xii
Daftar Tabel.....	xiv
Daftar Lampiran.....	xv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Ruang Lingkup Penelitian.....	2
1.4 Tujuan Penelitian.....	2
1.5 Manfaat Penelitian.....	2
1.6 Sistematika Penelitian.....	3
BAB II TINJAUAN PUSTAKA.....	4
2.1 Penelitian Sebelumnya.....	4
2.2 Landasan Teori.....	5
2.2.1 Brankas.....	5
2.2.2 Mikrokontroler.....	5
2.2.3 Arduino Uno (Wemos D1 R3).....	8
2.2.4 Sensor.....	9
2.2.5 Sensor Fingerprint.....	10
2.2.6 Liquid Crystal Display (LCD).....	12

2.2.7 Material Liquid Crystal Display (LCD).....	12
2.2.8 Pengendali / Kontroler LCD.....	12
2.2.9 Pin pada LCD.....	15
2.2.10 Prinsip Menggunakan LCD.....	15
2.2.11 Keypad Membran 4x4.....	16
2.2.12 Relay Single Channel.....	18
2.2.13 Selenoid Door Lock.....	19
2.2.14 Catu Daya (Power Supply).....	19
2.2.15 Buzzer.....	21
BAB III METODE PENELITIAN.....	22
3.1 Rancangan Sistem.....	22
3.1.1 Diagram Block Sistem.....	22
3.1.2 Perancangan Rangkaian.....	23
3.1.3 Prinsip Kerja Alat.....	26
3.1.4 Flowchart Sistem.....	27
3.2 Implementasi Sistem.....	28
3.2.1 Tahap Persiapan.....	28
3.2.2 Tahap Pembuatan Sistem.....	29
3.2.3 Tahap Pengukuran dan Analisis Sistem.....	29
3.2.4 Desain Perangkat Keras (Hardware).....	30
3.2.5 Desain Perangkat Lunak (Software).....	31
3.3 Pengolahan Data Dan Analisis.....	32
3.3.1 Tahap Persiapan.....	32
3.3.2 Tahap Pembuatan Sistem.....	33
3.3.3 Tahap Pengukuran dan Analisis Sistem.....	35
BAB IV HASIL DAN PEMBAHASAN.....	36
4.1 Pengujian Hasil Penelitian.....	36

4.1.1 Pengujian Keypad Matrix 4x4.....	36
4.1.2 Pengujian Fingerprint.....	38
4.1.3 Pengujian LCD 16x2.....	40
4.1.4 Pengujian Kombinasi Data Fingerprint dan Keypad.....	41
4.2 Analisa Hasil Pengujian.....	42
BAB V PENUTUP.....	44
5.1 Kesimpulan.....	44
5.2 Saran.....	44
DAFTAR PUSTAKA.....	45
LAMPIRAN.....	47

DAFTAR GAMBAR

	Halaman
Gambar 2.1: Brankas	5
Gambar 2.2: Arduino (Wemos D1 R3).....	9
Gambar 2.3: Sensor Fingerprint.....	9
Gambar 2.4: LCD I2C 16x2	11
Gambar 2.5: Keypad Membran 4x4	12
Gambar 2.6: Relay Single Channel.....	18
Gambar 2.7: Solenoid Door Lock	19
Gambar 2.8: Power Supply	20
Gambar 2.9: Buzzer	21
Gambar 3.1: Diagram Block Sistem	23
Gambar 3.2: Rangkaian Arduino Wemos D1	23
Gambar 3.3: Rangkaian Fingerprint	24
Gambar 3.4: Rangkaian Keypad	24
Gambar 3.5: Rangkaian LCD	25
Gambar 3.6: Rangkaian Relay	25
Gambar 3.7: Rangkaian Solenoid Door Lock	25
Gambar 3.8: Rangkaian Power Supply	26
Gambar 3.9: Flowchart Sistem	27
Gambar 3.10: Rangkaian Skematik Sistem	30
Gambar 3.11: Arduino IDE	31
Gambar 3.12: Blynk.....	32
Gambar 3.13: Pengujian Sensor Fingerprint.....	33
Gambar 3.14: Pengujian Keypad	34
Gambar 3.15: Pengujian LCD	35
Gambar 4.1: Tampilan Program Keypad Pada Arduino.....	36

Gambar 4.2: Hasil Tampilan Program Keypad	38
Gambar 4.3: Tampilan Program Sensor Fingerprint Pada Arduino	38
Gambar 4.4: Hasil Tampilan Program Sidik Jari.....	39
Gambar 4.5: Tampilan Program LCD Pada Arduino	40
Gambar 4.6: Hasil Tampilan Program LCD.....	41
Gambar 4.7: Hasil Tampilan Jika Sidik Jari dan Keypad Benar	42
Gambar 4.8: Hasil Tampilan pada Blynk	43
Gambar 4.9: Hasil Tampilan Jika Sidik Jari dan Keypad Salah.....	43

DAFTAR TABEL

	Halaman
Tabel 2.1: Spesifikasi Arduino Uno (Wemos D1 R3).....	9
Tabel 2.2: Konfigurasi Pin LCD.....	14
Tabel 2.3: Fungsi Pin LCD	15
Tabel 2.4: Kombinasi Keypad	17
Tabel 3.1: Pengujian Sensor FingerPrint	32
Tabel 3.2: Pengujian Tekan Keypad.....	33
Tabel 3.3: Pengujian Keypad	34
Tabel 4.1: Hasil Pencocokan Pengujian keypad / Password	37
Tabel 4.2: Hasil Pencocokan Pengujian Sidik Jari	39
Tabel 4.3: Hasil Kombinasi data Fingerprint dan keypad	42

DAFTAR LAMPIRAN

	Halaman
Lampiran 1 Coding Pendaftaran Fingerprint.....	47
Lampiran 2 Coding Program Keseluruhan	53
Lampiran 3 Gambar Alat Keseluruhan	62
Lampiran 4 Rangkaian Alat Keseluruhan.....	63
Lampiran 5 Proses Pengujian Rangkaian Alat Keseluruhan	63

BAB I

PENDAHULUAN

1.1 Latar Belakang

Segala upaya dilakukan demi mempermudah pekerjaan manusia dari waktu ke waktu yang membutuhkan mobilitas tinggi dalam melakukan pekerjaan serta otomatisasi sehingga manusia mendapat kemudahan dari teknologi tersebut. Dengan semakin pesatnya perkembangan teknologi di era industri modern sekarang ini, keamanan merupakan sebuah topik yang didalamnya sangat luas tetapi dalam hal ini kajian tentang keamanan hanya dalam lingkup keamanan rumah saja. Sebuah sistem keamanan rumah (*home security system*) dapat diartikan sebagai suatu himpunan atau kumpulan dari komponen, unsur atau variabel yang terorganisasi, saling berinteraksi dan bergantung satu sama lain serta terpadu yang menghasilkan aman dan nyaman untuk semua penghuni yang ada di rumah tersebut. Semua sistem keamanan yang beredar dipasaran memiliki tujuan atau fungsi yang sama yaitu menciptakan kondisi yang aman serta nyaman bagi pengguna. Dalam memilih sistem keamanan yang harus diperhatikan adalah ketepatan kita dalam memilih suatu sistem keamanan yang sesuai dengan kondisi serta dapat diterapkan dilingkungan pengguna. Pada masa globalisasi saat ini sistem harus dapat dimonitor dari jarak jauh merupakan suatu keharusan agar dapat memudahkan pengguna [1].

Sistem keamanan rumah salah satunya ada pada keamanan pada pintu kamar. Pada saat ini kunci yang beredar dipasaran memiliki tingkat keamanan yang rendah atau bisa dikatakan tidak aman lagi [2]. Hanya dalam hitungan menit saja seseorang dapat membuka pintu konvensional atau umum yang beredar dipasaran secara mudah menggunakan dua buah kawat saja [3]. Selain itu kunci yang umum beredar masih menggunakan anak kunci yang mengakibatkan sistem pengamanan ini menjadi kurang aman karena anak kunci mudah hilang dan juga dapat diduplikat oleh orang dalam yang membuat tindak kejahatan orang dalam tidak bisa diidentifikasi secara mudah [4]. Disinilah merupakan awal permasalahan yaitu sistem keamanan kunci yang lemah, tidak praktis dan juga tidak ada sistem untuk dapat melihat siapa saja seseorang yang telah masuk atau keluar ruangan didalam rumah. Perkembangan teknologi informasi dan elektronika yang sangat pesat menjadi salah satu solusi yang dapat diterapkan dan relevan untuk masa kini salah satunya otentikasi biometrik dengan pengenalan sidik jari.

Oleh karena itu, lahirlah ide untuk merancang sistem keamanan yang lebih aman " Rancang Bangun Sistem Pengaman Pintu Brankas Menggunakan Fingerprint dan Keypad Berbasis Arduino."

1.2 Rumusan Masalah

Berdasarkan uraian di atas, penulis tertarik untuk merancang sistem keamanan yang aman menggunakan sidik jari dan *keypad* berbasis Arduino Uno. Dimana dalam perancangan ini permasalahannya dinyatakan:

1. Bagaimanakah rancang bangun sistem pengaman brankas menggunakan *Keypad* dan sidik jari?
2. Bagaimanakah prinsip kerja dari sistem pengaman brankas menggunakan *Keypad* dan sidik jari?
3. Bagaimanakah prinsip kerja sidik jari dalam sistem pengaman brankas?

1.3 Batasan Masalah

Berdasarkan dari hasil penelitian yang telah dilakukan, maka batasan masalah dalam penelitian ini, yaitu:

1. Mikrokontroler yang digunakan adalah Arduino Uno (Wemos D1).
2. Alat ini hanya membuka kunci pada brankas.
3. Sensor Fingerprint hanya digunakan sebagai inputan dalam membaca sidik jari.

1.4 Tujuan Penelitian

1. Dapat menganalisa rancang bangun dari sistem pengaman brankas menggunakan *Keypad* dan sidik jari.
2. Dapat menganalisa prinsip kerja dari sistem pengaman brankas menggunakan *Keypad* dan sidik jari.
3. Dapat menganalisa prinsip kerja sidik jari dalam sistem pengaman brankas.

1.5 Manfaat Penelitian

Manfaat dari penelitian ini adalah:

1. Membuat kunci elektronik yang mudah digunakan.
2. Lebih efisien dalam membuka kunci brankas karena pengguna tinggal menempelkan sidik jari pada Fingerprint.

3. Mempermudah ketika hendak membuka kunci brankas dari dalam karena terdapat sensor sentuh.
4. Mahasiswa dapat mengembangkan dan mengaplikasikan ilmu yang sudah didapat dalam perkuliahan.
5. Penelitian ini dapat dijadikan sebagai bahan referensi untuk penelitian selanjutnya.

1.6 Sistematika Penulisan

Memfasilitasi pelaporan dan sistematisasi kinerja alat " Rancang Bangun Sistem Pengaman Pintu Brankas Menggunakan Fingerprint dan Keypad Berbasis Arduino ", dalam hal ini penulis membaginya menjadi beberapa bab dan memberikan gambaran singkat isi setiap babnya sebagai berikut:

BAB 1 : PENDAHULUAN

Bab ini memuat pendahuluan yang meliputi latar belakang, rumusan masalah, batasan masalah, tujuan masalah, manfaat peneletian.

BAB 2 : LANDASAN TEORI

Bab ini memuat landasan teori yang membahas tentang teori-teori yang mendukung dalam penyelesaian masalah.

BAB 3 : METODE PENELITIAN

Bab ini memuat tentang sistem perancangan alat yaitu diagram blok rangkaian, sistematik, dan cara kerja dari rangkaian.

BAB 4 : HASIL DAN PEMBAHASAN

Bab ini membahas hasil analisa dari rangkaian dan sistem kerja alat, penjelasan mengenai rangkaian-rangkaian yang digunakan.

BAB 5 : PENUTUP

Bab ini menjelaskan tentang kesimpulan dari pengujian dan saran masukan untuk mengembangkan dan melengkapi sistem yang sudah dibangun untuk masa yang mendatang.

DAFTAR PUSTAKA

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Sebelumnya

Semua sistem keamanan yang beredar dipasaran memiliki tujuan atau fungsi yang sama yaitu menciptakan kondisi yang aman serta nyaman bagi pengguna. Dalam memilih sistem keamanan yang harus diperhatikan adalah ketepatan kita dalam memilih suatu sistem keamanan yang sesuai dengan kondisi serta dapat diterapkan dilingkungan pengguna. Pada masa globalisasi saat ini sistem harus dapat dimonitor dari jarak jauh merupakan suatu keharusan agar dapat memudahkan pengguna [1]. Sistem keamanan rumah salah satunya ada pada keamanan pada pintu kamar. Pada saat ini kunci yang beredar dipasaran memiliki tingkat keamanan yang rendah atau bisa dikatakan tidak aman lagi [2].

Penelitian selanjutnya yang berjudul "Arsitektur Sistem Keamanan Rumah Dengan Menggunakan Teknologi Biometrik Sidik Jari Berbasis Arduino" [6], digunakan sidik jari sebagai kunci pintu. Hasil yang dicapai terdapat kekurangan, yaitu sistem tidak ada log sistem berupa web dan autentikasi kode. Penelitian tentang keamanan juga dibuat pada AKPOL dengan judul "Perancangan Sistem Keamanan Hak Akses Pintu AKPOL Semarang dengan RFID" [8], Sistem ini memiliki kekurangan karena dianggap kurang praktis. Berdasarkan kajian diatas, perbedaan utama penelitian ini dengan penelitian sebelumnya yaitu dari sisi sistem dan teknologi yang digunakan. Pada penelitian ini smart security system menggunakan sidik jari dan *keypad* menggunakan web sebagai alat monitoring data yang masuk secara *realtime* sehingga pemilik rumah mengetahui siapa saja yang masuk dengan sistem log.

Penelitian berikutnya dilakukan juga di Universitas PGRI Yogyakarta dengan judul "Rancang Bangun Aplikasi Smart Room di Fakultas Sains dan Teknologi Universitas PGRI Yogyakarta" [9]. Hasil penelitian ini adalah dosen bisa membuka pintu dan menyalakan ac atau kipas secara otomatis dengan RFID.

Penelitian berikutnya dari Eni Yuliza dan Toibah Umi Kalsum.2015. Alat Keamanan Pintu Brankas Berbasis Sensor Sidik Jari dan Password Digital dengan Menggunakan Mikrokontroler ATmega16 [14]. Salah satunya sistem keamanan pada pintu brankas yang rawan dan kemungkinan terjadinya pencurian sangat besar. Pada suatu brankas sekarang ini sangat diperlukan adanya sebuah keamanan, agar bisa mengantisipasi bahaya pencurian yang dapat terjadi secara tak terduga. Pintu brankas ini tidak semua orang bisa

membuka karena pintu brankas ini mempunyai suatu alat yaitu dengan cara sidik jari dan password.

2.2 Landasan Teori

2.2.1 Brankas

Brankas adalah lemari atau brankas yang digunakan untuk melindungi barang-barang berharga seperti uang, surat berharga dan lain-lain dari pencurian/penemuan seperti terlihat pada Gambar 2.1. Kunci pengaman untuk brankas umumnya berdasarkan sistem operasinya, ada dua jenis yaitu digital dan analog. Tidak jarang ada yang menggabungkan dua sistem, brankas sistem digital hadir dengan kunci analog (kunci manual) dan ada juga brankas yang menggunakan dua kunci analog (kunci kombinasi terbalik dan kunci panjang). Penggunaan brankas yang beredar dan dijual di pasaran tidak selalu dapat memenuhi kebutuhan keamanan setiap orang barang berharga miliknya. Dalam kasus brankas, hal ini terjadi karena beberapa keadaan, yaitu kunci yang tertindih, pintu brankas dibobol, dan masih banyak lagi kondisi lainnya, rata-rata terjadi ketika pemilik brankas tidak ada, sehingga siapapun bisa mencoba membukanya. aman. aman dan mengeluarkan isi brankas. aman karena membuka brankas relatif mudah. Hal ini menunjukkan bahwa brankas yang dijual di pasaran masih mencukupi dan memberikan perasaan lebih aman kepada penggunanya. [14]



Gambar 2.1 Brankas

(Sumber: priceza.co.id)

2.2.2 Mikrokontroler

Saat ini perkembangan teknologi semakin pesat berkat adanya teknologi mikrokontroler, sehingga rangkaian kendali atau rangkaian kontrol semakin banyak

dibutuhkan untuk mengendalikan berbagai peralatan yang digunakan manusia dalam kehidupan sehari-hari. Dari rangkaian kendali inilah akan terciptanya suatu alat yang dapat mengendalikan sesuatu. Rangkaian kendali atau rangkaian kontrol adalah rangkaian yang dirancang sedemikian rupa sehingga dapat melakukan fungsi–fungsi kontrol tertentu sesuai dengan kebutuhan. Bermula dari dibuatnya *Integrated Circuit* (IC). Selain IC, alat yang dapat berfungsi sebagai kendali adalah chip sama halnya dengan IC. Chip merupakan perkembangan dari IC, dimana chip berisikan rangkaian elektronika yang dibuat dari artikel silikon yang mampu melakukan proses logika. Chip berfungsi sebagai media penyimpan program dan data, karena pada sebuah chip tersedia RAM dimana data dan program ini digunakan oleh *logic chip* dalam menjalankan prosesnya. Chip lebih diidentikkan dengan dengan kata mikroprosesor. Mikroprosesor adalah bagian dari *Central Processing Unit* (CPU) yang terdapat pada komputer tanpa adanya memori, I/O yang dibutuhkan oleh sebuah sistem yang lengkap. Selain mikroprosesor ada sebuah chip lagi yang dikenal dengan nama mikrokomputer. Berbeda dengan mikroprosesor, pada mikrokomputer ini telah tersedia I/O dan memori. Dengan kemajuan teknologi dan dengan perkembangan chip yang pesat sehingga saat ini didalam sekeping chip terdapat CPU memori dan kontrol I/O. Chip jenis ini sering disebut mikrokontroler. Mikrokontroler merupakan sebuah sistem komputer di mana seluruh atau sebagian besar elemennya dikemas dalam satu chip IC (*Integrated Circuit*), sehingga sering disebut single chip microcomputer. Mikrokontroler ini juga merupakan sebuah sistem komputer yang memiliki satu atau beberapa tugas yang spesifik, berbeda dengan PC yang memiliki beragam fungsi. Perbedaan yang lain adalah perbandingan RAM dan ROM yang sangat besar antara mikrokontroler dengan komputer. Dalam mikrokontroler ROM jauh lebih besar dibanding RAM, sedangkan dalam komputer atau PC RAM jauh lebih besar dibanding ROM. Mikrokontroler memiliki kemampuan untuk mengolah serta memproses data sekaligus juga dapat digunakan sebagai unit kendali, maka dengan sekeping chip yaitu mikrokontroler kita dapat mengendalikan suatu alat. Mikrokontroler mempunyai perbedaan dengan mikroprosesor dan mikrokomputer. Suatu mikroprosesor merupakan bagian dari CPU tanpa memori dan I/O pendukung dari sebuah komputer, sedangkan mikrokontroler umumnya terdiri atas CPU, memori, I/O tertentu dan unit –unit pendukung lainnya. Pada dasarnya terdapat perbedaan sangat mencolok antara mikrokontroler dan mikroprosesor serta mikrokomputer yaitu pada aplikasinya, karena mikrokontroler hanya dapat digunakan pada aplikasi tertentu saja. Kelebihan lainnya yaitu terletak pada perbandingan Random Access Memory (RAM) dan Read Only Memory (ROM). Sehingga

ukuran board mikrokontroler menjadi sangat ringkas atau kecil, dari kelebihan yang ada terdapat keuntungan pemakaian mikrokontroler dengan mikroprosesor yaitu pada mikrokontroler sudah terdapat RAM dan peralatan I/O pendukung sehingga tidak perlu menambahnya lagi. Pada dasarnya struktur dari mikroprosesor memiliki kemiripan dengan mikrokontroler. Mikrokontroler biasanya dikelompokkan dalam satu keluarga, masing-masing mikrokontroler memiliki spesifikasi tersendiri namun cocok dalam pemrogramannya misalnya keluarga MCS-51 yang diproduksi ATMEL seperti AT89C51, AT89S52 dan lainnya sedangkan keluarga AVR seperti Atmega 8535 dan lain sebagainya. [15]

Mikrokontroler adalah sebuah sistem komputer fungsional dalam sebuah chip. Di dalamnya terkandung sebuah inti prosesor, memori (sejumlah kecil RAM, memori program, atau keduanya), dan perlengkapan input output. Dengan kata lain, mikrokontroler adalah suatu alat elektronika digital yang mempunyai masukan dan keluaran serta kendali dengan program yang bisa ditulis dan dihapus dengan cara khusus, cara kerja mikrokontroler sebenarnya membaca dan menulis data. Sekedar contoh, bayangkan diri Anda saat mulai belajar membaca dan menulis, ketika Anda sudah bisa melakukan hal itu Anda bisa membaca tulisan apapun baik buku, cerpen, artikel dan sebagainya, dan Andapun bisa pula menulis hal-hal sebaliknya. Begitu pula jika Anda sudah mahir membaca dan menulis data maka Anda dapat membuat program untuk membuat suatu sistem pengaturan otomatis menggunakan mikrokontroler sesuai keinginan Anda. Mikrokontroler merupakan komputer didalam chip yang digunakan untuk mengontrol peralatan elektronik, yang menekankan efisiensi dan efektifitas biaya. Secara harfiahnya bisa disebut “pengendali kecil” dimana sebuah sistem elektronik yang sebelumnya banyak memerlukan komponen-komponen pendukung seperti IC TTL dan CMOS dapat direduksi/diperkecil dan akhirnya terpusat serta dikendalikan oleh mikrokontroler ini. Mikrokonktroler digunakan dalam produk dan alat yang dikendalikan secara otomatis, seperti sistem kontrol mesin, remote controls, mesin kantor, peralatan rumah tangga, alat berat, dan mainan. Dengan mengurangi ukuran, biaya, dan konsumsi tenaga dibandingkan dengan mendesain menggunakan mikroprosesor memori, dan alat input output yang terpisah, kehadiran mikrokontroler membuat kontrol elektrik untuk berbagai proses menjadi lebih ekonomis. Dengan penggunaan mikrokontroler ini maka:

- a) Sistem elektronik akan menjadi lebih ringkas.

b) Rancang bangun sistem elektronik akan lebih cepat karena sebagian besar dari sistem adalah perangkat lunak yang mudah dimodifikasi.

c) Pencarian gangguan lebih mudah ditelusuri karena sistemnya yang kompak.

Agar sebuah mikrokontroler dapat berfungsi, maka mikrokontroler tersebut memerlukan komponen eksternal yang kemudian disebut dengan sistem minimum. Untuk membuat sistem minimal paling tidak dibutuhkan sistem *clock* dan reset, walaupun pada beberapa mikrokontroler sudah menyediakan sistem *clock* internal, sehingga tanpa rangkaian eksternal pun mikrokontroler sudah beroperasi. Yang dimaksud dengan sistem minimal adalah sebuah rangkaian mikrokontroler yang sudah dapat digunakan untuk menjalankan sebuah aplikasi. Sebuah IC mikrokontroler tidak akan berarti bila hanya berdiri sendiri. Pada dasarnya sebuah sistem minimal mikrokontroler AVR memiliki prinsip yang sama. [4] Pada pembahasan ini Mikrokontroler yang digunakan adalah Arduino Uno seperti pada Gambar 2.2.

2.2.3 Arduino Uno (Wemos D1 R32)

Sehubungan dengan pembahasan untuk saat ini mikrokontroler yang digunakan adalah Wemos D1 R32 yang merupakan dasar (base) Arduino Uno, walaupun masih ada beberapa software lain yang sangat berguna selama pengembangan Arduino. Gambar 2.2 menunjukkan gambar Wemos D1 R32 base Arduino Uno.

IDE Arduino adalah software yang sangat canggih ditulis dengan menggunakan Java. IDE Arduino terdiri dari:

1. Editor program, sebuah window yang memungkinkan pengguna menulis dan mengedit program dalam bahasa Processing.
2. Compiler, sebuah modul yang mengubah kode program (bahasa Processing) menjadi kode biner. Bagaimanapun sebuah mikrokontroler tidak akan bisa memahami Bahasa Processing. Yang bisa dipahami oleh mikrokontroler adalah kode biner. Itulah sebabnya compiler diperlukan dalam hal ini.
3. Uploader, sebuah modul yang memuat kode biner dari komputer ke dalam memory di dalam papan Arduino. [5]



Gambar 2.2 Arduino Uno (Wemos D1 R32)

(Sumber: nshopvn.com)

Tabel 2.1 Spesifikasi Arduino Uno (Wemos D1 R32)

Mikrokontroller	ESP-8266X
Input Tegangan	3.3V
Pin I/O Digital	11
Pin Analog	1
Kecepatan Clock	80MHz/160MHz
Flash	4 Mbytes

2.2.4 Sensor

Sensor adalah piranti yang menerima sebuah stimulus atau rangsangan sebagai kuantitas sifat atau kondisi tertentu yang dapat diubah menjadi sinyal listrik. Output dari sensor dapat berupa arus dan sering digunakan untuk pendeteksian saat melakukan pengukuran dan pengendalian. Karakteristik sensor berfungsi untuk mengetahui kinerja dari sensor yang dirancang. Ciri-ciri karakteristik statis dari sebuah sensor yaitu:

1. Akurasi

Untuk mengetahui ketidak akuratan dari sebuah sensor, sebagai perbedaan nilai perhitungan dengan nilai eksperimen.

2. Non linearitas

Digunakan untuk sensor yang memiliki fungsi transfer dari pendekatan linier. Non linearitas juga sebagai pendekatan maksimum dan dapat dilakukan untuk sensor dengan fungsi transfer non linear yang dapat digunakan dengan metode terminal point dan metode transfer.

3. Saturasi

Setiap sensor memiliki batasan operasi meskipun memiliki fungsi transfer linier tetapi pada input tertentu memiliki transfer non linear.

4. Resolusi

Merupakan kemampuan sensor dalam mendeteksi sinyal input minimum. Saat sensor diberikan input secara berlanjut, maka sinyal output pada sensor tidak akan memberikan hasil yang sempurna. Dengan kondisi demikian biasanya terjadi perubahan output, maka sensor tersebut dikatakan memiliki resolusi yang kecil.

5. Repeatabilitas

Terjadi karena sensor tidak mampu untuk menghasilkan nilai yang sama pada kondisi yang sama, biasanya dikarenakan gangguan temperatur dan kondisi lingkungan lainnya.

[18]

2.2.5 Sensor Fingerprint

Sidik jari adalah garis-garis pada kulit ujung jari. Sidik jari memberikan gaya gesek yang lebih besar sehingga jari dapat menggenggam benda dengan lebih kuat. Sistem keamanan sidik jari dimulai di Amerika oleh seorang pria bernama E. Henry pada tahun 1902. Henry menggunakan metode sidik jari untuk mengidentifikasi pekerja agar mendapatkan upah ganda. Sistem Henry menggunakan pola umum (jambul = tonjolan pada kulit, baik pada tangan) yang berfokus pada pola jari-jari, khususnya jari telunjuk. Untuk mendapatkan gambar pola timbul, gerakkan jari yang bertinta di atas kartu yang dicetak untuk membuat pola timbul yang unik untuk setiap orang. Para ahli telah membuktikan bahwa tidak ada dua individu yang memiliki pola sisir yang sama. Pola yang dimunculkan tidak diwariskan. Pola tulang rusuk terbentuk selama perkembangan janin dan tidak berubah sepanjang hidup. Perubahan puncak hanya dapat terjadi karena trauma seperti cedera, luka bakar, penyakit, atau sebab lainnya. Sistem biometrik sidik jari saat ini merupakan sistem yang paling banyak digunakan karena tingkat akurasi yang tinggi. [16] Bentuk fisik sensor sidik jari ditunjukkan pada Gambar 2.3.



Gambar 2.3 *Sensor Fingerprint*

(Sumber: *Rahmad Syam, M. H., 2010.*)

Spesifikasi Sensor fingerprint:

1. Tegangan suplay: 3,6 - 6,0VDC
2. Arus operasi: maks 120mA
3. Arus puncak: 150mA
4. maksimal Waktu pencitraan sidik jari: <1,0 detik
5. Area jendela: 14mm x 18mm
6. File tanda tangan: 256 byte
7. File template: 512 byte
8. Penyimpanan kapasitas: 162 templat
9. Peringkat keselamatan (1-5 keselamatan rendah ke tinggi) Tingkat Penerimaan Palsu: <0,001% (Tingkat Keamanan 3)
10. Tingkat Penolakan Salah: <1,0% (Tingkat 3)
11. Antarmuka: Serial TTL
12. Tingkat Baud: 9600, 19200, 28800, 38400, 57600 (standarnya adalah 57600)
13. Peringkat suhu kerja: -20C hingga + 50C
14. Kelembaban kerja: 40% - 85% RH
15. Dimensi Penuh: 56 x 20 x 21,5mm Dimensi Terkena (bila ditempatkan di kotak): 21mm x 21mm x 21mm segitiga
16. Berat: 20 gram

2.2.6 *Liquid Crystal Display (LCD)*

LCD adalah salah satu jenis display elektronik yang dibuat dengan teknologi logika CMOS yang bekerja dengan tidak menghasilkan cahaya tetapi memantulkan cahaya yang ada di sekelilingnya terhadap front-lit atau mentransmisikan cahaya dari back-lit. IIC *Modul Serial Interface Backpack* IIC LCD Backpack bekerja layaknya shift register yang memiliki pin antarmuka lebih sedikit sehingga dapat mengatasi masalah habisnya digital pin pada arduino. IIC LCD backpack modul mempunyai 16 pinout yang bisa dihubungkan dengan pin LCD secara langsung dan memiliki 4 pin input (VCC, GND, SDA, SCL). [11]

2.2.7 *Material Liquid Crystal Display (LCD)*

LCD merupakan lapisan campuran organik antara lapisan kaca transparan dengan elektroda indium oksida transparan berbentuk tampilan 7 segmen dan lapisan elektroda di bagian belakang kaca, seperti terlihat pada Gambar 2.4. Ketika elektroda diaktifkan oleh medan listrik (tegangan), molekul organik berbentuk silinder panjang sejajar dengan elektroda yang tersegmentasi. Lapisan sandwich dilengkapi dengan polarizer vertikal di bagian depan dan polarizer horizontal di bagian belakang, diikuti oleh lapisan reflektif. Cahaya yang dipantulkan tidak dapat menembus molekul yang selaras dan segmen yang diaktifkan menjadi gelap dan membentuk fitur data yang ditampilkan. [11]



Gambar 2.4 LCD I2C 16x2

(Sumber: Sinukaban, GK.2019.)

2.2.8 **Pengendali / Kontroler LCD**

Dalam modul LCD terdapat mikrokontroler yang berfungsi sebagai pengendali tampilan karakter LCD. *Microntroller* pada suatu LCD dilengkapi dengan memori dan *register*. Memori yang digunakan *microcontroler* internal LCD adalah:

- 1) DDRAM (*Display Data Random Access Memory*) merupakan memori tempat karakter yang akan ditampilkan berada.
- 2) CGRAM (*Character Generator Random Access Memory*) merupakan memori untuk menggambarkan pola sebuah karakter dimana bentuk dari karakter dapat diubah-ubah sesuai dengan keinginan.
- 3) CGROM (*Character Generator Read Only Memory*) merupakan memori untuk menggambarkan pola sebuah karakter dimana pola tersebut merupakan karakter dasar yang sudah ditentukan secara permanen oleh pabrikan pembuat LCD (*Liquid Cristal Display*) tersebut sehingga pengguna tinggal mengambilnya sesuai alamat memorinya dan tidak dapat merubah karakter dasar yang ada dalam CGROM.

Berikut Register control yang terdapat dalam suatu LCD diantaranya adalah:

- 1) Register perintah yaitu register yang berisi perintah-perintah dari mikrokontroler ke panel LCD pada saat proses penulisan data atau tempat status dari panel LCD dapat dibaca pada saat pembacaan data.
- 2) Register data yaitu register untuk menuliskan atau membaca data dari atau ke DDRAM. Penulisan data pada register akan menempatkan data tersebut ke DDRAM sesuai dengan alamat yang telah diatur sebelumnya.

Pin kaki atau jalur input dan kontrol dalam suatu LCD diantaranya adalah:

- 1) Pin data adalah jalur untuk memberikan data karakter yang ingin ditampilkan menggunakan LCD dapat dihubungkan dengan bus data dari rangkaian lain seperti mikrokontroler dengan lebar data 8 bit.
- 2) Pin RS (Register Select) berfungsi sebagai indikator atau yang menentukan jenis data yang masuk, apakah data atau perintah. Logika low menunjukkan yang masuk adalah perintah, sedangkan logika high menunjukkan data.
- 3) Pin R/W (Read Write) berfungsi sebagai instruksi pada modul jika low tulis data sedangkan high baca data.
- 4) Pin E (Enable) digunakan untuk memegang data baik masuk atau keluar.

- 5) Pin VLCD berfungsi mengatur kecerahan tampilan (kontras) dimana pin ini dihubungkan dengan trimpot 5 Kohm, jika tidak digunakan dihubungkan ke ground, sedangkan tegangan catu daya ke LCD sebesar 5 Volt.

LCD yang digunakan adalah jenis LCD yang menampilkan data dengan 2 baris tampilan pada display. Keuntungan dari LCD ini adalah:

- 1) Dapat menampilkan karakter ASCII, sehingga dapat memudahkan untuk membuat program tampilan.
- 2) Mudah dihubungkan dengan port I/O karena hanya menggunakan 8 Bit Data dan 3 Bit Control.
- 3) Ukuran modul yang proporsional.
- 4) Data yang digunakan relative sangat kecil.

Operasi dasar pada LCD terdiri dari empat, yaitu instruksi mengakses proses internal, instruksi menulis data, instruksi membaca kondisi sibuk, dan instruksi membaca data. ROM pembangkit sebanyak 192 tipe karakter, tiap karakter dengan huruf 5x7 dot matrik. Kapasitas pembangkit RAM 8 tipe karakter (membaca program), maksimum pembacaan 80x8 bit tampilan data. Perintah utama LCD adalah *Dsisplay Clear*, *Cursor Home*, *Dsisplay ON/OFF*, *Display Character Blink*, *Cursor Shift*, dan *Display Shift*.

Tabel 2.2 Konfigurasi Pin LCD

No. Pin	Keterangan	Konfigurasi Hubung
1	GND	Ground
2	VCC	Tegangan +5VDC
3	VEE	Ground
4	RS	Kendali RS
5	RW	Ground
6	E	Kendali E/Eneble
7	D0	Bit 0
8	D1	Bit 1
9	D2	Bit 2
10	D3	Bit 3
11	D4	Bit 4
12	D5	Bit 5
13	D6	Bit 6
14	D7	Bit 7
15	A	Anoda (+5VDC)
16	K	Katoda (Ground)

LCD memerlukan voltase rendah dan daya rendah, menjadikannya sangat populer dalam aplikasi seperti kalkulator dan jam digital. LCD biasanya dikemas dalam bentuk paket dual in-line (DIP) dan dapat menampilkan beberapa kolom dan baris secara bersamaan menggunakan metode penyaringan. Cara penyaringannya adalah dengan mengaktifkan perpotongan kolom dan baris secara bergantian dan cepat agar tampak seolah-olah semuanya aktif. Tujuan penggunaan cara ini adalah untuk menyimpan jalur pengaktifan panel LCD. Fungsionalitas LCD juga telah ditingkatkan dari monokrom menjadi ribuan warna.

2.2.9 Pin pada LCD

LCD memiliki pin dari 1 hingga 16 pin. Masing-masing pulpen ini mempunyai kegunaannya masing-masing. Antarmukanya bisa menggunakan sistem 4-bit atau 8-bit. Menggunakan sistem 4-bit menghemat 4 port mikrokontroler. Penggunaan masing-masing pin LCD ditunjukkan pada Tabel 2.3 di bawah ini.

Tabel 2.3 Fungsi Pin LCD

No. Pin	Nama	Fungsi
1	GND	Ground untuk LCD
2	VCC	+5 untuk LCD
3	Vreff	Tegangan Pengatur <i>brightness</i>
4	RS	Bit pemilih intruksi / data
5	R/W	Bit pemilih <i>Read / Write</i>
6	E	Bit <i>Enable</i>
7	D0	Data Bit 0
8	D1	Data Bit 1
9	D2	Data Bit 2
10	D3	Data Bit 3
11	D4	Data Bit 4
12	D5	Data Bit 5
13	D6	Data Bit 6
14	D7	Data Bit 7
15	Back Light (+)	Optimal
16	Back Light (-)	Optimal

2.2.10 Prinsip Menggunakan LCD

Modul LCD memiliki tiga jalur kontrol bernama RS, R/W, dan E. RS digunakan untuk memberitahu LCD apakah data yang ditentukan adalah kata perintah atau kata data. RS

harus diset ke 0 untuk mengirim instruksi dan diset ke logika 1 untuk mengirim data. Sedangkan garis R/W digunakan untuk memilih baca atau tulis. “Read” berarti membaca data dari LCD dan “write” berarti menulis data ke LCD. Dalam hal ini, Anda dapat menjaga baris ini tetap rendah (logika 0) karena Anda hanya menulis data ke LCD. Terakhir ada baris E (aktifkan) yang bila logikanya tinggi (1), memungkinkan penulisan ke LCD. Kata perintah yang dikirim ke LCD memberitahu pengontrol LCD apa yang harus dilakukan.

2.2.11 Keypad Membran 4x4

Keypad adalah tombol-tombol yang disusun secara matriks yang berfungsi untuk memasukkan data seperti masukkan pintu otomatis, masukkan absensi, masukkan data logger dan sebagainya. Konstruksi keypad matrix 4×4 cukup sederhana, yaitu terdiri dari 4 baris dan 4 kolom dengan keypad berupa saklar push buton yang diletakan disetiap persilangan kolom dan barisnya. Kolom dan baris dihubungkan ke port pada mikrokontroler. Menekan tombol akan menghubungkan baris dan kolom yang sesuai. Membaca suatu baris dilakukan dengan mengatur semua kolom ke logika rendah. Pada saat ini, port yang terhubung ke kolom menjadi output dan port yang terhubung ke baris menjadi input. Pembacaan dilakukan dengan cara memindai (membaca) setiap baris dan setiap kolom. Jika sakelar tidak ditekan, semua baris terbaca logis 1. Jika salah satu baris menunjukkan nilai 0, berarti saklar ditekan pada baris tersebut (terhubung ke kolom logis 0). Langkah selanjutnya adalah mencari saklar mana yang sebenarnya ditekan, yaitu mencari kolom yang terhubung dengan saklar tersebut. Ketika saklar ditekan, mikrokontroler membaca logika 0. Mengetahui kolom mana yang berlogika 0 pada saat itu, mikrokontroler mengetahui kolom mana yang saklarnya ditekan. [15] Gambar 2.7 menunjukkan tampilan fisik keyboard matriks 4x4.



Gambar 2.5 Keypad Membran 4x4

(Sumber: Trimarsiah, Y. 2016.)

Saat tombol ditekan, maka baris dan kolom akan terhubung ke *ground* sehingga kondisi pada baris dan kolom tersebut akan menjadi *low*. Apabila tombol ‘1’ ditekan, maka baris 1 dan kolom 1 akan terhubung ke *ground* sehingga kondisi baris dan kolom tersebut akan berubah menjadi *low*, demikian pula pada tombol ‘2’ dan seterusnya sehingga terbentuk tabel berikut. Tabel 2.6 Kombinasi Keypad.

Tabel 2.4 Kombinasi Keypad

Tombol	K4 PB. 7	K3 PB. 6	K2 PB. 5	K1 PB. 4	B4 PB. 3	B3 PB. 2	B2 PB. 1	B1 PB. 0	Hexa
1	0	1	1	1	0	1	1	1	77
2	0	1	1	1	1	0	1	1	7B
3	0	1	1	1	1	1	0	1	7D
A	0	1	1	1	1	1	1	0	7E
4	1	0	1	1	0	1	1	1	B7
5	1	0	1	1	1	0	1	1	BB
6	1	0	1	1	1	1	0	1	BD
B	1	0	1	1	1	1	1	0	BE
7	1	1	0	1	0	1	1	1	D7
8	1	1	0	1	1	0	1	1	DB
9	1	1	0	1	1	1	0	1	DD
C	1	1	0	1	1	1	1	0	DE
*	1	1	1	0	0	1	1	1	E7
0	1	1	1	0	1	0	1	1	EB
#	1	1	1	0	1	1	0	1	ED
D	1	1	1	0	1	1	1	0	EE

Pengambilan data dari keypad dilakukan dengan menunggu adanya penekanan tombol keypad. Kondisi tidak ada penekanan tombol adalah *high* untuk semua pin keypad kecuali *common* yang terhubung ke *ground* atau FFh pada port mikrokontroler. Walaupun kondisi port mikrokontroler bukan lagi FFh, penekanan keypad masih belum *valid*, hal ini disebabkan adanya *bouncing*, atau getaran secara mekanis dalam tombol keypad yang terjadi. Oleh karena itu, apabila pengambilan data keypad langsung dilakukan saat itu, maka akan sering kali terjadi kesalahan. Data keypad akan *valid* apabila salah satu baris telah terhubung dengan salah satu kolom dan *common*. Hal ini ditandai dengan adanya hanya dua buah logika 0 pada kaki-kaki keypad. Contohnya pada penekanan tombol ‘2’,

maka data dari *keypad* hanya akan *valid* bila baris dua dan kolom dua sudah terhubung ke *ground* atau berlogika 0.

Untuk mengetahui kondisi ini, dapat dilakukan dengan memasukkan data *keypad* ke akumulator dan memeriksa kondisi *flag parity*. Apabila jumlah logika 0 dalam akumulator adalah genap, maka *flag parity* akan *clear*, dan apabila jumlah logika 0 dalam akumulator adalah ganjil, maka *flag parity* akan *set*. Program akan terus menerus mengambil data dari *keypad* hingga jumlah logika 0 dalam akumulator adalah genap atau *flag parity clear*. Setelah data *valid* diambil, maka program akan menunggu tombol *keypad* dilepas dengan menunggu adanya kondisi FFh kembali serta melakukan konversi berdasarkan tabel *keypad* setelah kondisi tersebut terpenuhi. Proses konversi tabel *keypad* dilakukan dengan menganggap data-data dari *keypad* sebagai suatu alamat memori di mana isi dari alamat tersebut adalah berupa data yang dianggap sebagai tanda saat tombol tersebut ditekan. Contohnya pada tombol '1' yang menghasilkan data B7h pada *keypad*. Program akan mengambil data di alamat Tabel *Keypad* + B7h di mana pada alamat tersebut dapat diisi 01H atau 31H (ASCII dari bilangan 1) tergantung kebutuhan pengguna.

2.2.12 Single Channel Relay Module

Relay adalah saklar yang dioperasikan secara listrik dan merupakan komponen elektromekanikal yang terdiri dari dua bagian utama yakni elektromagnet (*coil*) dan mekanikal (seperangkat kontak saklar/*switch*). Relay menggunakan prinsip elektromagnetik untuk menggerakkan kontak saklar sehingga dengan arus listrik yang kecil (*low power*) dapat menghantarkan listrik yang bertegangan lebih tinggi. [13]



Gambar 2.6 Relay Single Channel

(Sumber : tokopedia.com)

2.2.13 Solenoid Door Lock

Pengunci pintu adalah perangkat elektronik kunci pintu dengan menggunakan tegangan listrik sebagai pengendalinya. Alat ini banyak diaplikasikan pada pintu otomatis. Solenoid pengunci pintu bekerja jika diberi tegangan. Dalam keadaan normal tuas pada solenoid pengunci pintu akan memanjang, dan jika diberi tegangan tuas pada alat ini akan memendek. Tegangan listrik yang diberikan akan membuat medan magnet sehingga tuas pada solenoid pengunci pintu akan tertarik oleh medan magnet. [19] [26]



Gambar 2.7 Solenoid Door Lock

(Sumber: tokopedia.com)

Didalam solenoid terdapat kawat melingkar pada inti besi. Ketika arus listrik melalui kawat ini, maka terjadi medan magnet untuk menghasilkan energi yang bisa mendorong inti besi. Poros dalam dari solenoid adalah piston seperti silinder terbuat dari besi atau baja, yang disebut *pluger* (setara dengan sebuah dinamo) medan magnet kemudian menerapkan kekuatan untuk *pluger* ini, baik menarik atau *repling* (kembali posisi). Ketika medan magnet dimatikan, pegas *pluger* kemudian kembali ke posisi semula.

2.2.14 Catu Daya (Power Supply)

Catu daya (Power Supply) adalah suatu alat listrik yang dapat menyediakan energi listrik untuk perangkat listrik ataupun elektronika lainnya. Pada dasarnya Power Supply atau Catu daya ini memerlukan sumber energi listrik yang kemudian mengubahnya menjadi energi listrik yang dibutuhkan oleh perangkat elektronika lainnya. Oleh karena itu, Power Supply kadang-kadang disebut juga dengan istilah Electric Power Converter. Maka didalam alat yang saya gunakan terdapat 2 elemen power supply yang berbeda fungsi, yakni sebagai berikut:

1. Switch-Mode Power Supply

Switch-Mode Power Supply (SMPS) adalah jenis Power Supply yang langsung menyearahkan (*rectrify*) dan menyaring (*filter*) tegangan Input AC untuk mendapatkan tegangan DC. Tegangan DC tersebut kemudian di-switch ON dan OFF pada frekuensi tinggi dengan sirkuit frekuensi tinggi sehingga menghasilkan arus AC yang dapat melewati Transformator Frekuensi Tinggi. Bisa kita lihat bentuk fisik dari pada Power Supply pada gambar 2.9 dibawah ini: [26]

2. Pengatur Tegangan Tetap (*Fixed Voltage Regulator*)

IC jenis Pengatur Tegangan Tetap (*Fixed Voltage Regulator*) ini memiliki nilai tetap yang tidak dapat disetel (*di-adjust*) sesuai dengan keinginan Rangkaiannya. Tegangannya telah ditetapkan oleh produsen IC sehingga Tegangan DC yang diatur juga Tetap sesuai dengan spesifikasi IC-nya. Misalnya IC Voltage Regulator 7805, maka Output Tegangan DC-nya juga hanya 5 Volt DC. Terdapat 2 jenis Pengatur Tegangan Tetap yaitu Positive Voltage Regulator dan Negative Voltage Regulator. Jenis IC Voltage Regulator yang paling sering ditemukan di Pasaran adalah tipe 78XX. 9 Tanda XX dibelakangnya adalah Kode Angka yang menunjukkan Tegangan Output DC pada IC Voltage Regulator tersebut. Contohnya 7805, 7809, 7812 dan lain sebagainya. IC 78XX merupakan IC jenis Positive Voltage Regulator. IC yang berjenis Negative Voltage Regulator memiliki desain, konstruksi dan cara kerja yang sama dengan jenis Positive Voltage Regulator, yang membedakannya hanya polaritas pada Tegangan Outputnya. Contoh IC jenis Negative Voltage Regulator diantaranya adalah 7905, 7912 atau IC Voltage Regulator berawalan kode 79XX. [26]



Gambar 2.8 Power Supply

(*Sumber: teknikelektronika.com*)

2.2.15 Buzzer

Buzzer adalah sebuah komponen elektronika yang berfungsi untuk mengubah getaran listrik menjadi getaran suara. Pada dasarnya prinsip kerja buzzer hampir sama dengan loud speaker. [27]



Gambar 2.9 Buzzer

(Sumber: [pinterest.com](https://www.pinterest.com))

BAB III

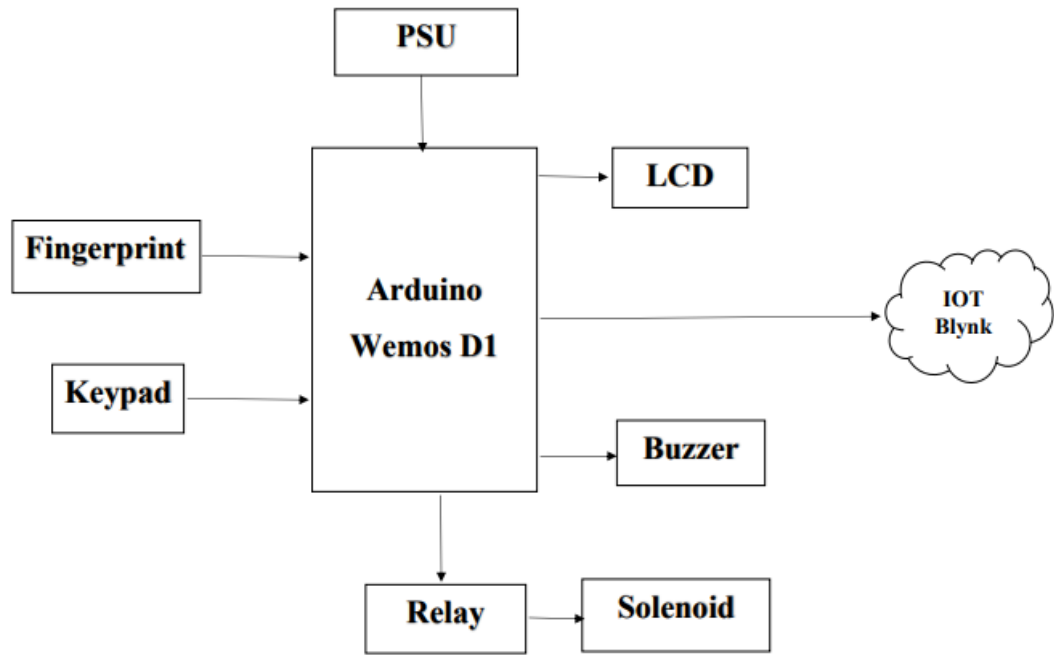
METODE PENELITIAN

Metode penelitian ini adalah penelitian kuantitatif. Penelitian Kuantitatif adalah penelitian yang mengambil data dalam jumlah yang banyak. Bisa puluhan, ratusan, atau mungkin ribuan. Proses dan makna lebih ditonjolkan dalam penelitian kuantitatif. Pengumpulan Data Referensi Tahap ini mempelajari berbagai informasi yang berkaitan dengan rancang bangun sistem pengamanan pintu brankas menggunakan *Fingerprint*, *keypad* dan *solenoid door lock* baik dari internet, buku, jurnal, hingga bisa menjadi referensi untuk membantu pembuatan alat. Jenis-jenis data dalam penelitian kualitatif adalah data non-numerik atau angka. Data ini biasanya berisi analisa kondisi saat ini pada organisasi sehingga membantu peneliti dalam menentukan permasalahan. Penelitian kuantitatif cocok digunakan untuk meneliti masalah yang sudah jelas, memiliki populasi luas, dan bermaksud untuk menguji hipotesis. [10]

3.1 Rancangan Sistem

3.1.1 Diagram Blok Sistem

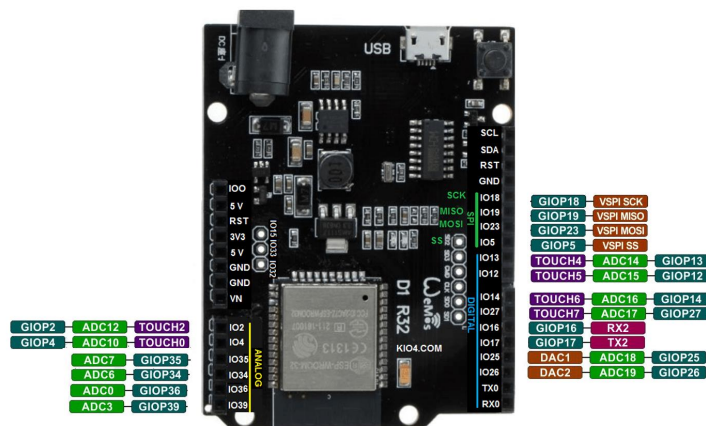
Sesuai dengan judul yang telah disusun oleh penulis yaitu mengenai sistem keamanan pada pintu brankas menggunakan *Fingerprint* dan *keypad*, maka pada penelitian ini akan merancang dan membangun sebuah alat dan aplikasi *logging* yang dapat membantu pemilik brankas dalam mengamankan barang berharga didalam rumah dan mengetahui siapa saja yang membuka dan mengakses brankas. Secara keseluruhan sistem terdiri atas beberapa bagian yang dapat digambarkan berdasarkan diagram blok Rancang Bangun Sistem Pengaman Pintu Brankas Menggunakan *Fingerprint* dan *Keypad* Berbasis Arduino berikut ini, pada Gambar 3.1 di bawah ini.



Gambar 3.1 Diagram Blok Sistem

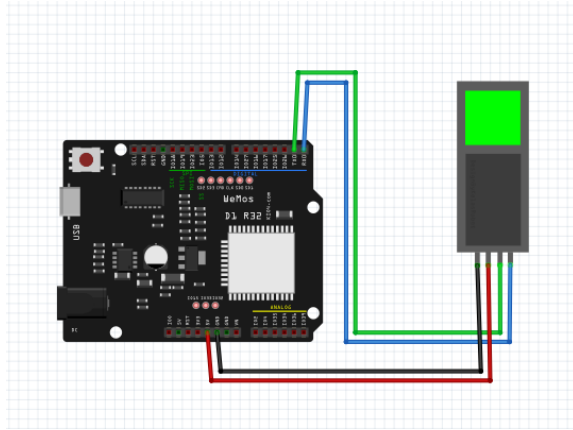
3.1.2 Perancangan Rangkaian

1. Arduino Uno (wemos D1) berfungsi untuk memudahkan pengguna dalam melakukan prototyping, memprogram mikrokontroler serta menciptakan berbagai alat canggih berbasis mikrokontroler.



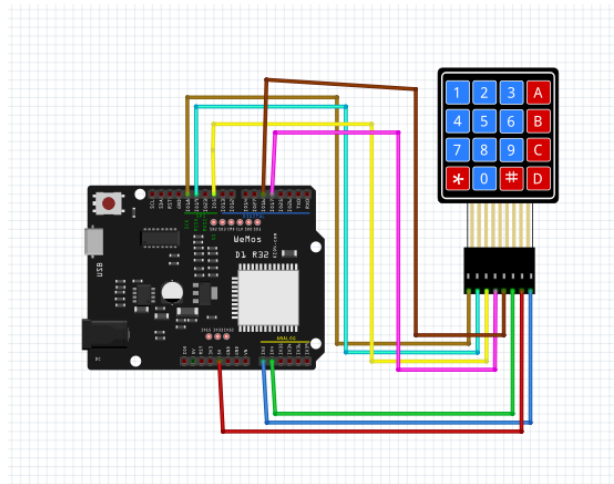
Gambar 3.2 Rangkaian Arduino Wemos D1

2. Sensor sidik jari yang berfungsi sebagai input sistem pengaman dengan menggunakan sidik jari, yang dimana sidik jari setiap orang berbeda, hanya sidik jari yang sudah terdaftar pada sistem saja yang dapat membuka kunci pengaman pada brankas.



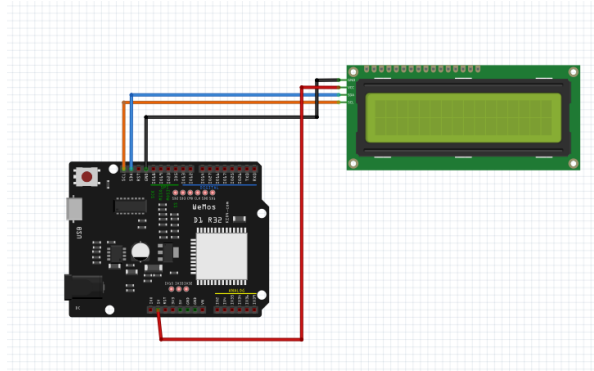
Gambar 3.3 Rangkaian Fingerprint

3. *Keypad* berfungsi sebagai *interface* antara perangkat (mesin) elektronik dengan manusia atau dikenal dengan istilah *Human Machine Interface (HMI)*.



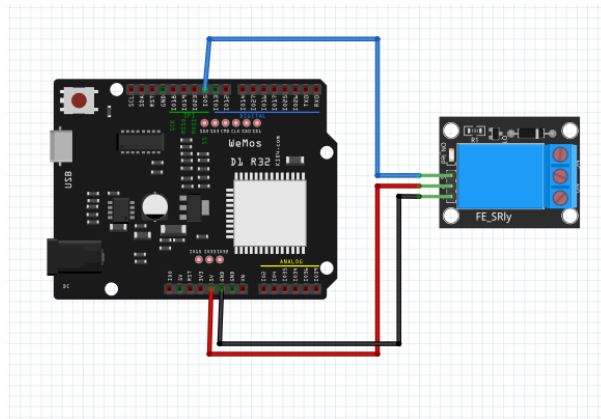
Gambar 3.4 Rangkaian Keypad

4. LCD berfungsi untuk menampilkan karakter angka, huruf ataupun simbol dengan konsumsi arus yang rendah. Pada alat ini, display yang digunakan adalah *Liquid Crystal Display (LCD)* 16 x 2. Untuk blok ini tidak ada komponen tambahan karena mikrokontroler dapat memberi data langsung ke LCD, pada LCD Hitachi – M1632 sudah terdapat driver untuk mengubah data ASCII output mikrokontroler menjadi tampilan karakter. Pemasangan potensio sebesar 5 K Ω untuk mengatur kontras karakter yang tampil.



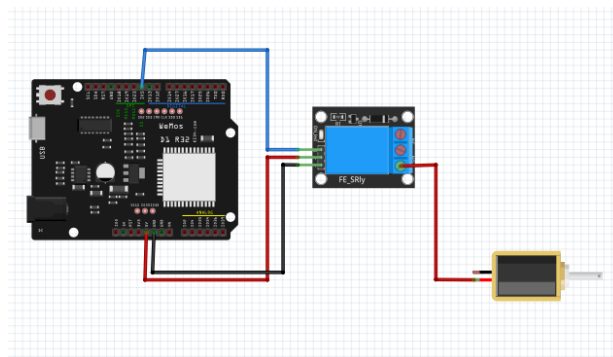
Gambar 3.5 Rangkaian LCD

5. *Relay 1 Channel* memiliki fungsi sebagai memutus dan menyambung tegangan sesuai dengan perintah yang ditentukan.



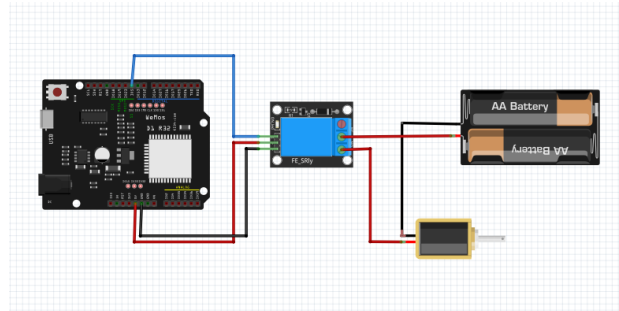
Gambar 3.6 Rangkaian Relay

6. *Solenoid Door Lock* adalah salah satu solenoid yang difungsikan khusus sebagai solenoid untuk pengunci pintu secara elektronik. Solenoid ini mempunyai dua sistem kerja, yaitu *Normaly Close* (NC) dan *Normaly Open* (NO).



Gambar 3.7 Rangkaian Selenoid Door Lock

7. *Power supply* merupakan perangkat yang berfungsi sebagai pengubah tegangan masukan AC menjadi DC.



Gambar 3.8 Rangkaian Power Supply

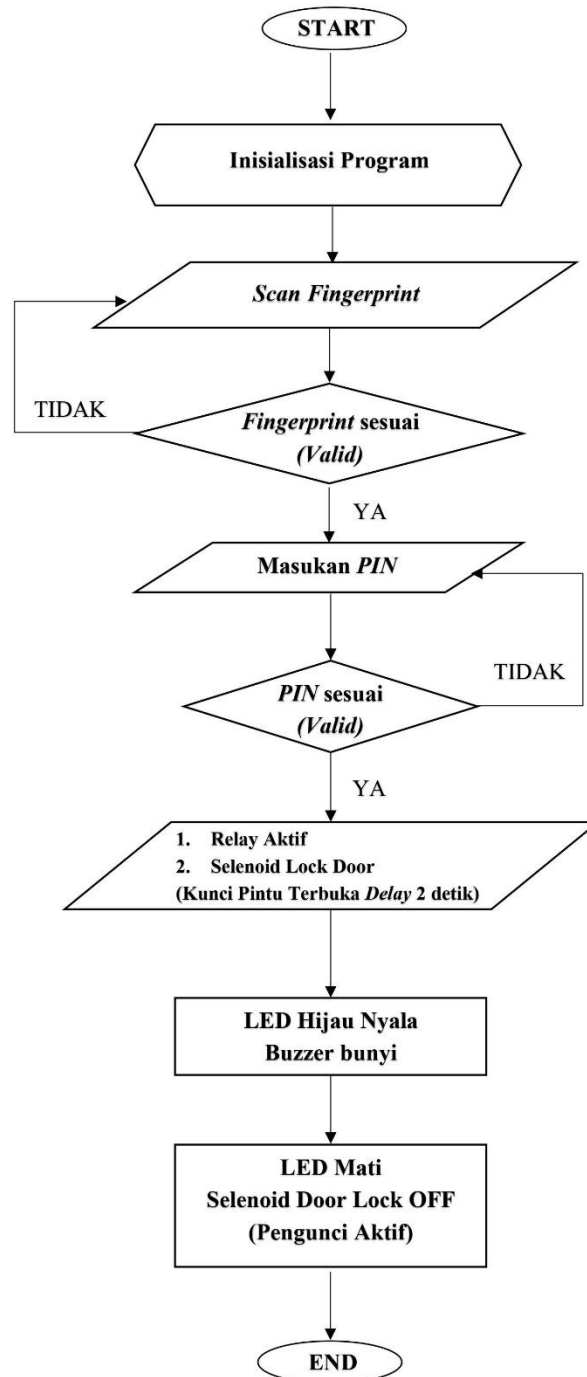
3.1.3 Prinsip Kerja Alat

Pada saat alat dihidupkan atau diaktifkan, brankas akan terkunci. Untuk membuka pertama harus dilakukan pencocokan sidik jari pada fingerprint, jika sidik jari tepat, maka akan muncul respon di layar lcd, dan akan ada perintah berikutnya yaitu memasukkan password yang telah ditentukan pada tombol keypad. Jika password yang dimasukkan tepat maka kunci solenoid akan bergerak membuka pintu brankas.

Untuk kondisi salah pada saat pencocokan sidik jari yang tidak sesuai, maka buzzer akan berbunyi sebagai tanda ada yang mencoba membuka brankas secara illegal. alat ini juga bisa dikendalikan melalui komunikasi usart serial melalui port serial pada komputer, dengan menggunakan aplikasi Arduino IDE.

3.1.4 Flowchart Sistem

Flowchart Sistem pengamanan menggunakan sensor *Fingerprint* dan *Keypad*. Seperti pada Gambar 3.9.



Gambar 3.9 Flowchart Sistem

3.2 Implementasi Sistem

Tahap implementasi ini menjelaskan mengenai perangkat keras dan perangkat lunak yang digunakan dalam membangun sistem ini. Pengujian alat dan program dilakukan untuk mengetahui apakah alat dan program yang dibuat agar bekerja dengan baik sesuai proyek yang dibuat. Tahap akhir dalam pembuatan laporan didapatkan dari hasil analisis data pada alat yang dibuat berdasarkan pengujian-pengujian yang dilakukan.

3.2.1 Tahap Persiapan

Metode Pelaksanaan dalam penelitian ini secara umum dibagi kedalam 5 tahap yang diperlihatkan oleh diagram berikut:

Tahap 1: Pendesainan Prototipe Alat

Pada tahap ini kegiatan yang dilakukan adalah mendesain Alat dengan menggunakan *software paint* pada laptop. Pada tahap ini akan di desain komponen-komponen alat yaitu desain ruang alat sebagai tempat sensor.

Tahap 2: Pembuatan Prototipe

Brankas ini terbuat dari bahan kayu, dan juga dengan menggunakan triplek sebagai wadah tempat objek pada alat.

Tahap 3: Pembuatan rangkaian alat

Pada tahap ini akan dilakukan pembuatan rangkaian alat yang berfungsi untuk mendapatkan hasil data secara otomatis yang diperoleh sensor.

Tahap 4: Pengukuran

Pada tahap ini akan dilakukan pencocokkan objek antara Sidik Jari dan *Password*.

Tahap 5: Analisa Data

Pada tahap ini akan dilakukan analisa terhadap data yang diperoleh dari hasil pencocokkan. Analisa ini meliputi keccokan *password*, dan sidik jari untuk dapat membuka brankas

3.2.2 Tahap Pembuatan Sistem

Selama fase ini, seperangkat alat akan dikembangkan untuk memungkinkan pengumpulan data otomatis yang dikumpulkan oleh sensor. Tahapan – Tahapan pelaksanaan tahap ini adalah:

- a. Mendesain layout rangkaian dengan *software Fritzing Beta Versi 0.9.3*.
- b. Mencetak hasil layout pada kertas foto dengan menggunakan printer.
- c. Memasang komponen elektronik sesuai dengan jalur pada layout.

3.2.3 Tahap Pengukuran dan Analisis Sistem

Dalam menyusun skripsi ini penulis melakukan beberapa penerapan metode penelitian untuk menyelesaikan permasalahan. Metode penelitian yang dilakukan adalah dengan cara:

- a. Studi pustaka untuk mengumpulkan, mempelajari serta menyeleksi bahan-bahan tentang pemrograman berbasis mikrokontroler Arduino.
- b. Pengumpulan data yang berhubungan dengan tugas akhir. Data yang dibutuhkan adalah data-data tentang komponen-komponen elektronika yang akan digunakan dalam perancangan alat.
- c. Analisis Sistem.

Melakukan analisis terhadap program yang akan dibuat serta komponen - komponen elektronika yang digunakan.

- d. Perancangan sistem.

Merancang suatu sistem pengaman brankas menggunakan *fingerprint* dan *keypad* sebagai pengaman utama. Termasuk interface aplikasi dan perancangan susunan rangkaian elektronika.

- e. Impelentasi Sistem (Coding).

Menyusun kode program untuk sistem tempat sampah yang dapat memisahkan sampah organic dan sampah non organic berbasis mikrokontroler arduino.

- f. Testing

Melakukan pengujian sistem yang telah dibuat sehingga dapat melakukan perbaikan sistem apabila ditemukan kesalahan pada sistem.

- g. Dokumentasi Sistem.

Pembuatan dokumentasi sistem lengkap dengan analisis yang telah diperoleh.

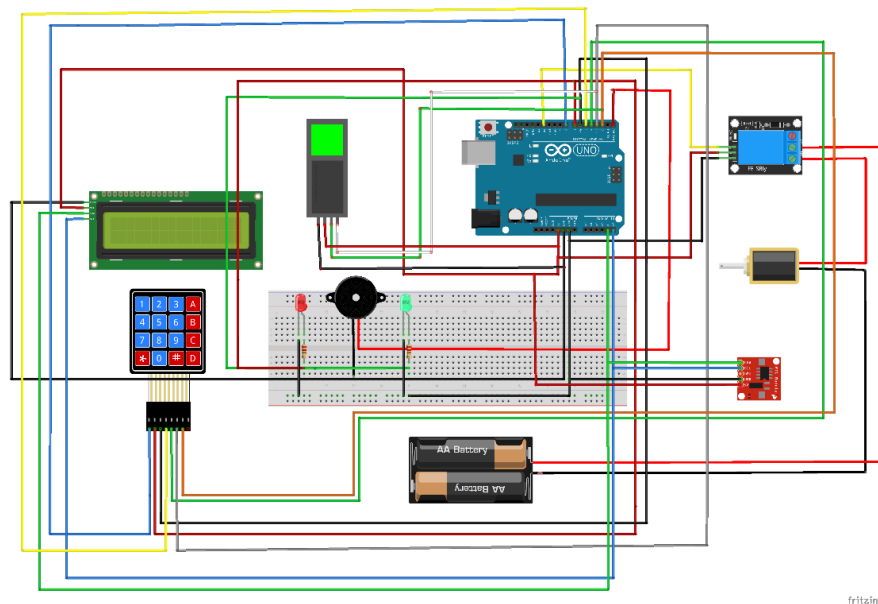
Analisis Sistem Sedang Berjalan

Bahasa Pemrograman yang digunakan pada alat adalah Bahasa Pemrograman Arduino yang dimana bahasa pemrograman Arduino adalah sebuah kit elektronik open source yang dirancang khusus dengan kemampuan komputasi yang dapat berinteraksi secara lebih

dekat dengan dunia nyata dibandingkan komputer biasa, untuk memudahkan bagi para seniman, desainer, dan siapapun yang tertarik dalam menciptakan objek atau mengembangkan perangkat elektronik yang dapat berinteraksi dengan bermacam-macam sensor dan pengendali Arduino dapat dioperasikan dengan cara menginstal terlebih dahulu software atau aplikasi pendukung untuk memprogram mikrokontroler arduino berupa program Sederhana, dengan programming environment turunan dari bahasa pemrograman C yang mudah dimengerti.

3.2.4 Desain Perangkat Keras (*Hardware*)

Pada tahap ini penulis menjelaskan bagaimana rancangan koneksi antar perangkat system yang akan dibuat. Setiap komponen akan memiliki fungsi masing – masing, ada bagian input, process dan output yang akan digunakan pada sistem ini.



Gambar 3.10 Rangkaian Skematik Sistem

Perancangan perangkat keras secara keseluruhan dibuat dengan menggunakan komponen-komponen yang sesuai dengan fungsinya berdasarkan diagram blok yang telah dibuat pada Gambar 3.1 diatas. Berikut ini adalah skematik rangkaian dari komponen yang digunakan pada Rancang Bangun Sistem Pengaman Pintu Brankas Menggunakan *Fingerprint* dan *Keypad* Berbasis Arduino menggunakan aplikasi Fritzing Beta Versi 0.9.3. Saat sistem keamanan ini diaktifkan maka akan muncul tampilan awal berisi kalimat pembuka pada layar LCD. Sensor keamanan yang pertama kali aktif adalah *Fingerprint*,

Keypad, lalu *Solenoid* akan bekerja melalui perintah *Relay*, maka pintu brankas akan terbuka secara otomatis tanpa melalui tahapan yang sudah disetel, yang kemudian akan langsung membunyikan buzzer serta lampu LED akan menyala berwarna hijau. Alat ini hanya dapat membuka pintu brankas dari satu sisi saja. Selain itu jika memasukkan password pada keypad salah maka sistem akan langsung mereset kembali ke awal program, ini dimaksudkan agar keamanan dapat lebih terjamin. Implementasi sistem ini menghasilkan antarmuka pengguna yang menghubungkan admin dengan perangkat elektronik. Dalam antarmuka pengguna ini, admin dapat melihat data pengguna yang membuka brankas tersebut.

3.2.5 Desain Perangkat Lunak (*Software*)

Arduino IDE adalah *software* yang digunakan untuk membuat *sketch* pemrograman atau dengan kata lain arduino IDE sebagai media untuk pemrograman pada *board* yang ingin diprogram. Arduino IDE ini berguna untuk mengedit, membuat, meng-*upload* ke *board* yang ditentukan, dan meng-*coding* program tertentu. Arduino IDE dibuat dari bahasa pemrograman JAVA, yang dilengkapi dengan *library* C/C++(*wiring*), yang membuat operasi *input/output* lebih mudah.



Gambar 3.11 Arduino IDE

(*Sumber: kmtech.id*)

Bahasa pemrograman yang digunakan pada alat ini adalah bahasa pemrograman Arduino. Ini adalah perangkat elektronik sumber terbuka yang dirancang khusus dengan kemampuan komputasi yang memungkinkan Anda berinteraksi lebih dekat dengan dunia nyata dibandingkan komputer biasa. Ini tersedia untuk seniman, desainer, dan siapa saja yang tertarik untuk membuat objek atau mengembangkan elektronik. Perangkat yang dapat berinteraksi dengan berbagai sensor dan pengontrol Arduino dapat bekerja dengannya dengan terlebih dahulu menginstal perangkat lunak atau aplikasi pendukung untuk bekerja dengan Arduino. Programkan mikrokontroler Anda dalam bentuk program sederhana

dengan menggunakan lingkungan pemrograman yang berasal dari bahasa pemrograman C yang mudah dipahami.

Blynk adalah aplikasi yang dapat diunduh secara gratis untuk iOS dan Android yang berfungsi mengontrol Arduino, Raspberry Pi dan sejenisnya melalui Internet. *Blynk* dirancang untuk *Internet of Things* dengan tujuan dapat mengontrol *hardware* dari jarak jauh, dapat menampilkan data sensor, dapat menyimpan data, visual dan melakukan banyak hal canggih lainnya. Ada tiga komponen utama dalam yaitu *Blynk App*, *Blynk Server*, dan *Blynk Library*.



Gambar 3.12 *Blynk*

(Sumber: nyebarilimu.com)

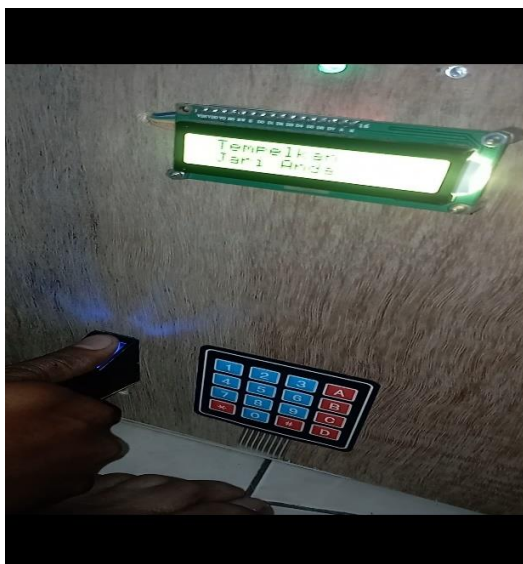
3.3 Pengujian dan Analisa Sistem

3.3.1 Pengujian Sensor *Fingerprint*

Pengujian ini dilakukan dengan tujuan untuk mengetahui kemampuan modul fingerprint dalam *scanning* sidik jari. *Fingerprint* harus bisa membedakan sidik jari yang sudah terdaftar di EEPROM dan sidik jari yang belum terdaftar.

Tabel 3.1 *Pengujian Sensor Fingerprint*

No.	Sidik Jari	Hasil
1	Aldy (Sidik Jari Pemilik)	Sidik Jari Cocok
2	Christo	Sidik Jari Tidak Cocok
3	Dimas	Sidik Jari Tidak Cocok
4	Putri	Sidik Jari Tidak Cocok
5	Rizka	Sidik Jari Tidak Cocok



Gambar 3.13 Pengujian Sensor Fingerprint

3.3.2 Pengujian Keypad

Pengujian ini untuk memastikan sistem keamanan dengan sandi berjalan dengan baik. Pengujian dilakukan dengan membuat rangkaian *keypad*, LCD dan Arduino.

Tabel 3.2 Pengujian Tekan keypad

No.	Penekanan Pada <i>Keypad</i>	Tampilan Pada LCD
1	Tekan 0	"0"
2	Tekan 1	"1"
3	Tekan 2	"2"
4	Tekan 3	"3"
5	Tekan 4	"4"
6	Tekan 5	"5"
7	Tekan 6	"6"
8	Tekan 7	"7"
9	Tekan 8	"8"
10	Tekan 9	"9"
11	Tekan A	"A"
12	Tekan B	"B"
13	Tekan C	"C"
14	Tekan D	"D"
15	Tekan *	"*"
16	Tekan #	"#"

Tabel 3.3 Pengujian keypad

No.	Password	Hasil
1	181818 (Password Pemilik)	Cocok
2	123456	Tidak Cocok
3	121212	Tidak Cocok
4	111111	Tidak Cocok
5	252525	Tidak Cocok



Gambar 3.14 Pengujian Keypad

3.3.3 Pengujian Tampilan LCD

Pengujian ini bertujuan memastikan bahwa LCD dapat berkomunikasi dengan arduino. Pin untuk komunikasi arduino adalah D8, D9, D10, D11, D12, dan D13 sedangkan pin untuk komunikasi \ LCD adalah DB4, DB5, DB6, DB7, E, dan RS. Potensio 10 k Ω berfungsi untuk mengatur pencahayaan pada LCD.



Gambar 3.15 Pengujian LCD

BAB IV

HASIL DAN PEMBAHASAN

4.1 Pengujian Hasil Penelitian

Pengujian alat ini secara keseluruhan adalah memastikan bahwa alat yang Anda buat berjalan dengan benar. Masukan yang berbeda diberikan dalam bentuk sidik jari yang berbeda dan password yang berbeda untuk menguji sistem keamanan yang aman. Periksa apakah alat ini berfungsi sebagaimana mestinya.

4.1.1. Pengujian Keypad matriks 4x4

Secara umum, prinsip kerja dari keypad 4x4 adalah memberikan masukan pada perangkat memproses melalui kombinasi kolom dan baris dari tombol-tombol yang ada pada keypad. Proses pembacaan tombol keypad ini dinamakan proses *scanning*. Program yang digunakan adalah Bascom-AVR sehingga proses *scanning keypad* lebih mudah. Hal ini dikarenakan didalam program Bascom-AVR telah terdapat perintah program yang berfungsi untuk *scanning keypad*. Berikut adalah potongan listing program *scanning keypad*.



```
sketch_0ct14a | Arduino 1.8.19
File Edit Sketch Tools Help
sketch_0ct14a.g
#include <Keypad.h>

//Mendefinisikan jumlah baris dan kolom keypad yang digunakan
const byte jumlahBaris = 4;
const byte jumlahKolom = 4;

//Memetakan tombol keypad
char petaTombol[jumlahBaris][jumlahKolom] =
{
  {'1','2','3','A'},
  {'4','5','6','B'},
  {'7','8','9','C'},
  {'*','0','#','D'}
};

//Mendefinisikan koneksi pin antara arduino dan keypad 4x4
byte pinBaris[jumlahBaris] = {5, 8, 7, 6};
byte pinKolom[jumlahKolom] = {5, 4, 3, 2};

//Mendefinisikan (Membuat) keypad
Keypad tombol = Keypad(macroMap(petaTombol), pinBaris, pinKolom, jumlahBaris, jumlahKolom);

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

void loop() {
  //Variabel untuk menyimpan data jika terjadi penekanan pada tombol keypad
}
```

Gambar 4.1 Tampilan Program Keypad pada Arduino

Pada gambar 4.1 diatas, listing program *scanning keypad* dimulai dengan mendeklarasikan variabel. Terdapat 2 variabel yang digunakan dalam *scanning keypad*, yaitu *Keypad* dan *Key*. Variabel *Keypad* nantinya akan digunakan untuk membaca nilai dari keypad, sedangkan variabel *Key* akan digunakan untuk menyimpan nilai yang diperoleh pada label data ketika adanya penekanan tombol keypad. Kemudian dilanjutkan dengan mengkonfigurasi *port* mikrokontroler yang akan digunakan untuk *scanning keypad* tersebut. Setelah itu dilanjutkan dengan

listing program utama yang berguna untuk *scanning keypad*. Terdapat perintah *Getkbd* dan *Lookup*, perintah *Getkbd* digunakan untuk *scanning keypad*, sedangkan *Lookup* digunakan untuk menentukan nilai yang terdapat pada label Data ketika adanya penekanan tombol *keypad*. Ketika tidak ada tombol *keypad* yang ditekan, maka nilai dari *Key* adalah 16. Sedangkan ketika ada tombol *keypad* yang ditekan, maka nilai hasil penekanan *keypad* akan diambil dari label Data dan menampilkannya pada layar LCD. Nilai yang terdapat pada tag data disesuaikan dengan karakter numerik dari keyboard itu sendiri. Cara ini lebih mudah dibandingkan pemindaian manual. Saat memindai secara manual, salah satu baris dan kolom keyboard harus digunakan sebagai keluaran dan yang lainnya sebagai masukan. Pemindaian manual bekerja dengan mengirimkan bit-bit pada setiap kolom (kolom keyboard yang terhubung ke pin mikrokontroler) yaitu. '000001' maka bit logika '1' digeser ke kiri setiap 1 siklus dan ketika bit logika '1' sudah mencapai bit paling kiri ('100000'), maka proses scan kembali ke awal (logika '1' kembali ke lokasi aslinya, yaitu '000001'). Sedangkan untuk baris, seluruh pin mikrokontroler yang terhubung ke baris keyboard digunakan sebagai input dengan nilai awal yang rendah (logika '0'). Jadi, ketika tombol keyboard ditekan, kombinasi baris dan kolom tombol keyboard yang ditekan memiliki logika '1'. Jadi hasilnya 1.000.000 kali lipat kombinasinya.

Tabel 4.1 Hasil Pencocokan Pengujian keypad / Password

No.	Password	Hasil
1	181818 (Password Pemilik)	Kunci Terbuka
2	123456	Kunci Tidak Terbuka
3	121212	Kunci Tidak Terbuka
4	111111	Kunci Tidak Terbuka
5	252525	Kunci Tidak Terbuka



Gambar 4.2 Hasil Tampilan Program Keypad

4.1.2. Pengujian *Fingerprint*

Pengujian ini dilakukan untuk mengetahui kemampuan modul sidik jari dalam memindai sidik jari. Sidik jari harus bisa membedakan antara sidik jari yang terdaftar di EEPROM dan sidik jari yang tidak. Dalam pengujian ini sidik jari yang terdaftar adalah sidik jari pemilik, Di sisi lain, sidik jari dari bukan pemilik yang belum terdaftar akan dicocokkan. Selain itu juga diuji apakah waktu yang diperlukan untuk menyelesaikan pemindaian kurang dari 1 detik sesuai datasheet. Gambar 4.2 dibawah ini menggunakan arduino untuk menguji modul sidik jari.

```

sketch_oct14e | Arduino 1.8.19
File Edit Sketch Tools Help
sketch_oct14e.g
#include "Adafruit_Fingerprint.h"
#include "SoftwareSerial.h"
SoftwareSerial mySerial(2, 3);
Adafruit_Fingerprint finger = Adafruit_Fingerprint(mySerial);

void setup() {
  Serial.begin(9600);
  finger.begin(57600);

  if (finger.verifyPassword()) {
    Serial.println("Fingerprint Sensor Ditemukan!");
  } else {
    Serial.println("Fingerprint Sensor Tidak Ditemukan! :(");
    while (1) {
      delay(1);
    }
  }
}

void loop() {
  FINGERPRINT();
}

//-----FINGERPRINT-----//
void FINGERPRINT() {
  getFingerprintID();
  delay(50);
}

```

Gambar 4.3 Tampilan Program Sensor Fingerprint pada Arduino

LED area pemindaian sidik jari akan berkedip terus menerus bila ada sidik jari yang datanya tidak ada di EEPROM, namun bila area pemindaian mendeteksi sidik jari yang datanya ada di EEPROM maka LED akan mati untuk melakukan operasi selanjutnya. Confidence adalah banyaknya data sidik jari yang didapat dari hasil scan. Dari hasil pengujian dapat disimpulkan kondisi sidik jari dalam keadaan baik dan tidak ada error, waktu pemindaian sidik jari cukup baik hampir semuanya dibawah 1 detik, namun dilakukan tanpa beban terkoneksi ke Arduino, yang bukan sidik jari.

Tabel 4.2 Hasil Pencocokan Pengujian Sidik Jari

No.	Sidik Jari	Hasil
1	Aldy (Sidik Jari Pemilik)	Kunci Terbuka
2	Christo	Kunci Tidak Terbuka
3	Dimas	Kunci Tidak Terbuka
4	Putri	Kunci Tidak Terbuka
5	Rizka	Kunci Tidak Terbuka



Gambar 4.4 Hasil Tampilan Program Sidik Jari

4.1.3. Pengujian LCD 16x2

Pengujian karakter Liquid Crystal Display (LCD) menggunakan pengujian karakter 16 x 2 untuk mendapatkan tampilan dalam format karakter yang diinginkan (huruf, angka, dll). Dimana karakter tersebut ditampilkan pada layar LCD. IC mikrokontroler diprogram agar layar kristal cair dapat menampilkan karakter (huruf, angka, dll). Program yang tertanam pada IC mikrokontroler berguna untuk menampilkan karakter (huruf, angka, dll) pada layar LCD. Berikut ini adalah kutipan dari daftar program pengujian LCD.



```
sketch_oct4a | Arduino 1.8.19
File Edit Sketch Tools Help

sketch_oct4a$
#include <LiquidCrystal.h> //menginkludakan library LCD

//untuk urutannya RS, E, 4, 5, 6, 7
LiquidCrystal lcd(12, 11, 5, 4, 3, 2); //pin yang disambungkan antara arduino dengan LCD

void setup()
{
  //prosedur pemanggilan fungsi LCD
  lcd.begin(16, 2); //16 = Baris, 2 = kolom
}

void loop() {
  lcd.setCursor(00,00); //set pada baris 1 dan kolom 1
  lcd.print("nyebarilmu.com"); //memuliskan "nyebarilmu.com"
  lcd.setCursor(00,1); //set pada baris 2 dan kolom 1
  lcd.print("Tutorial LCD16x2");
  delay(3000); //waktu tunda 3 detik
  lcd.clear();

  //Tampilan kedua
  lcd.setCursor(00,00); //set pada baris 1 dan kolom 1
  lcd.print("Halo Dunia!"); //memuliskan "Halo Dunia"
  lcd.setCursor(00,1); //set pada baris 2 dan kolom 1
  lcd.print("coba coba ya");
  delay(3000); //waktu tunda 3 detik
  lcd.clear();
}
}
```

Gambar 4.5 Tampilan Program LCD pada Arduino

Dalam memprogram LCD 16x2 kita harus terlebih dahulu mengkonfigurasi tipe LCD. Dalam perancangan ini digunakan LCD dengan tipe 16x2. Kemudian mensetting Cursor Off Noblink, hal ini diperlukan agar tidak ada kursor pada layar LCD yang berkedip. Kemudian dilanjutkan dengan mengkonfigurasi pin-pin mikrokontroler yang akan dihubungkan pada LCD. Pada perancangan ini digunakan mode 4-bit yaitu Db4, Db5, Db6 dan Db7 untuk mengirimkan data dari mikrokontroler ke LCD. Selain itu, pin Rs dan E pada LCD harus dihubungkan ke port mikrokontroler. Fungsi pin Rs merupakan instruksi khusus dimana Rs harus diset ke logika “1” atau high ketika karakter ditampilkan pada LCD. Sedangkan pin E digunakan untuk memberitahu LCD bahwa mikrokontroler telah mengirimkan data ke LCD. Setelah mengkonfigurasi pin-pin yang digunakan, maka dilanjutkan dengan listing program utama untuk penampil karakter pada LCD. Fungsi dari syntax Cls digunakan untuk membersihkan tampilan layar LCD. Sintaks untuk menampilkan

karakter pada LCD adalah LCD diikuti dengan tanda kutip (“) dan dilanjutkan dengan huruf atau angka yang nantinya akan ditampilkan pada LCD, kemudian diakhiri dengan tanda kutip (“).



Gambar 4.6 Hasil Tampilan Program LCD

4.1.4. Pengujian Kombinasi data Fingerprint dan Keypad

Hasil pengujian menunjukkan bahwa sistem keamanan hanya mendeteksi sidik jari yang terdaftar di program sistem keamanan, jika sidik jari tidak terdaftar di sistem, sistem mendeteksi bahwa itu adalah pencuri dan pintu brankas. tetap terkunci. Saat pengujian sistem keamanan dengan sensor sidik jari ini, pengenalan sidik jari dilakukan dengan sangat detail, sehingga kesalahan dalam pembacaan sidik jari sangat jarang terjadi selama sidik jari ditempatkan dengan benar. Pada percobaan kali ini kami melakukan 5 kali pengujian dengan satu sidik jari yang terdaftar di program. Pada pengujian pertama yaitu menggunakan sidik jari pemilik yang terdaftar di sistem keamanan dengan nama pengguna Aldy, ketika pengguna teridentifikasi oleh sensor sidik jari, sistem mendeteksi sidik jari dan memasukkan kata sandi “181818”, kemudian melakukan aktivasi. relay yang menyuplai tegangan 12 volt ke solenoid untuk membuka pintu brankas sehingga brankas dapat dibuka. Pada upaya kedua, ketiga, keempat dan kelima, sidik jari tidak terdaftar di sistem keamanan. Ketika

sensor sidik jari mendeteksi sidik jari, sistem mendeteksi kesalahan dan sistem menampilkan pesan kesalahan, dan solenoid kunci pintu tetap dalam posisi terkunci.

Tabel 4.3 Hasil Kombinasi data Fingerprint dan Keypad

No.	Sidik Jari	Password	Memasukkan data	Hasil
1	Aldy	181818	Berhasil	Kunci Terbuka
2	Christo	123456	Eror	Kunci Tidak Terbuka
3	Dimas	121212	Eror	Kunci Tidak Terbuka
4	Rizka	252525	Eror	Kunci Tidak Terbuka
5	Putri	111111	Eror	Kunci Tidak Terbuka



Gambar 4.7 Hasil Tampilan Jika Sidik Jari dan Keypad Benar

4.2 Analisa Hasil Pengujian

Berdasarkan informasi yang diterima, dapat disimpulkan bahwa selain sidik jari yang didaftarkan, juga akan ditampilkan hasil “sidik jari teman”. Jika Anda memasukkan kata sandi selain "181818", pintu tidak akan terbuka. Oleh karena itu, jika pintu brankas dibuka dengan menggunakan sidik jari yang sesuai dan terdaftar serta memasukkan password "181818" untuk membuka pintu brankas, maka pintu brankas tidak akan terbuka tanpa sidik jari pemiliknya.



Gambar 4.8 Hasil tampilan pada Blynk

Pada alat ini, Sistem bekerja setelah mendapat tegangan 12 volt. Sirkuit terhubung ke catu daya. Ketika sistem keamanan ini diaktifkan, layar pengaktifan akan muncul yang menyertakan frasa pengaktifan LCD. Sensor keamanan yang akan diaktifkan pertama kali adalah Fingerprint, Keypad, kemudian solenoid bekerja dengan perintah Relay, kemudian pintu brankas terbuka secara otomatis tanpa ada langkah yang ditentukan, kemudian buzzer akan langsung berbunyi dan LED akan menyala hijau, memberitahukan kepada pemilik bahwa posisi pintu terbuka. Brankas hanya bisa dibuka dari satu sisi dengan alat ini. Selain itu, jika sidak jari tidak cocok, maka sistem akan segera kembali ke awal program demi keamanan LED akan menyala merah.



Gambar 4.9 Hasil Tampilan Jika Sidik Jari Salah

BAB V

PENUTUP

5.1 Kesimpulan

Penelitian ini telah berhasil merancang dan membangun sebuah sistem pengamanan ganda menggunakan *fingerprint* dan *keypad* pada pintu brankas yang secara keseluruhan sudah berfungsi dengan baik. Sistem keamanan yang menggunakan sidik jari dan dioperasikan oleh Arduino Uno sebagai pusat kendali rangkaian. Rangkaian dapat berfungsi dengan baik untuk membuka pintu brankas dan memonitoring seseorang saat membuka dan menutup pintu brankas yang *controll* melalui aplikasi *blynk*.

1. Perancangan sistem keamanan brankas ini terdiri dari keyboard untuk memasukkan password. Sensor sidik jari sebagai input sidik jari untuk membuka keamanan yang merupakan keamanan utama. Arduino Uno sebagai pengontrol, penerima dan pengolah data. Brankas ini menggunakan kunci pintu solenoid sebagai pembuka pintu otomatis dan LCD sebagai layarnya.
2. Prinsip kerjanya adalah dapat menggunakan sidik jari dan keyboard untuk membuka brankas dengan aman. Memasukkan sidik jari dan kata sandi yang benar akan membuka brankas menggunakan sensor sidik jari yang dimasukkan menggunakan sidik jari manusia.
3. Pemindai sidik jari bekerja dengan cara “menangkap” sidik jari seseorang lalu menyimpan pola yang menjadi ciri khasnya. Identifikasi dilakukan dengan menggabungkan data yang disimpan. Jika diatur sama, akses akan terbuka secara otomatis.

5.2 Saran

Penggunaan sensor sidik jari masih memiliki kelemahan seperti sensor yang terlalu sensitif terhadap sidik jari, namun biasanya permukaan jari basah dan permukaan sensor kotor sehingga kurang maksimal. Untuk mencegah hal ini, tindakan seperti menjaga jari tetap kering dan membersihkan permukaan sidik jari harus dilakukan saat menggunakan.

Daftar Pustaka

- [1] A. F. S. Rahman, M. W. Kasrani, and A. W. P. Cristhobe, "Penggunaan Bluetooth & Gsm Modul Untuk Sistem Pengontrolan Smart Home," *Pros. Semin. Nas. Din. Inform.* 2018, vol. 2, no. 1, 2018.
- [2] S. Lumban Tobing, "Rancang Bangun Pengaman Pintu Menggunakan Sidik Jari (Fingerprint) Dan Smartphone Android Berbasis Mikrokontroler Atmega8," *Tek. Elektro Univ Tanjungpura Pontianak*, vol. 1, no. Rancang Bangun Pengaman Pintu, p. 2, 2015, doi: 10.1017/CBO9781107415324.004.
- [3] J. D. Purbani, "Pembuatan Mesin Identifikasi Sidik Jari Sebagai Kunci Pengaman Pintu," p. 48, 2010.
- [4] Ade Septryanti and Fitriyanti, "Berbasis Mikrokontroler Arduino Menggunakan," *Ranc. Bangun Apl. Kunci Pintu Otomatis Berbas. Mikrokontrol Arduino Menggunakan Smartphone Android*, vol. 2, no. 2, pp. 59–63, 2017, [Online]. Available: https://scholar.googleusercontent.com/scholar?q=cache:Wdcs4FzN0ZcJ:scholar.google.com/+pintu+otomatis+menggunakan+arduino&hl=en&as_sdt=0,5.
- [5] A Riantiarto, D Suryadi - *Jurnal Teknik Elektro Universitas ... - jurnal.untan.ac.id* Sistem monitoring arus dan kontrol beban listrik rumah tangga berbasis Arduino Uno dan Web berfungsi untuk memantau keadaan arus, tegangan, dan daya serta dapat mengontrol
- [6] A. Siswanto, A. Yulianti, and L. Costaner, "Arsitektur Sistem Keamanan Rumah Dengan Menggunakan Teknologi Biometrik Sidik Jari Berbasis Arduino," *Arsit. Sist. Keamanan Rumah Dengan Menggunakan Teknol Biometrik Sidik Jari Berbasis Arduino Apri*, no. November, pp. 3–7, 2017.
- [7] I Nyoman Sukarma, I Gede Suputra Widharma, Ade Surya Wiguna, 2016, https://www.researchgate.net/publication/346511868_Design_of_Safe-Deposit_Box_Security_System_Using_Both_Password_and_Finger_Print_Combination_with_Microcontroller_ATMEGA328_Based, Januari, 2016.
- [8] P. Gambiro, A. Triwiyatno, and B. Setiyono, "Perancangan Sistem Keamanan HakAkses Pintu AKPOL Semarang dengan RFID, 2014."
- [9] R. H. Hardyanto and W. I. Hamzah, "Rancang Bangun Aplikasi Smart Room di Fakultas Sains dan Teknologi Universitas PGRI Yogyakarta," *Semin. Nas. Din. Inform.* 2020 Univ. PGRI Yogyakarta, pp. 213–217, 2020.
- [10] Ms. Prof. Dr. Suryana, "Metodologi Penelitian: Metodologi Penelitian Model Praktis Penelitian Kuantitatif dan Kualitatif," *Univ. Pendidik. Indones.*, pp. 1–243, 2012, doi: 10.1007/s13398-014-0173-7.2.
- [11] Agarwal, P., "Algorithm to Create Multi Line Display from Two Line LCD". 2016.
- [12] Anton Yudhaa, Sunardi, Priyatno.2018. Perancangan Pengaman Pintu Rumah Berbasis Sidik Jari Menggunakan Metode UML.*Jurnal Teknologi Universitas Muhammadiyah Jakarta*.10:132-133
- [13] M. Saleh, and. M. Haryanti, "Rancang Bangun Sistem Keamanan Rumah Menggunakan Relay," *Jurnal Teknologi Elektro* vol. 8, pp. 87- 94. 2017

- [14] Eni Yuliza dan Toibah Umi Kalsum.2015. Alat Keamanan Pintu Brankas Berbasis Sensor Sidik Jari dan Password Digital dengan Menggunakan Mikrokontroler ATmega16.Jurnal Media Infotama.11:1-2
- [15] Trimarsiah, Yunita.2016. Pengaman Pintu Otomatis Menggunakan Keypad Matiks Berbasis Mikrokontroler AT89S52 pada Laboratorium STMIK-MURA LUBUK LINGGAU Hal:45-46
- [16] Setiawan, Idris.2015. Sistem pengaman Pintu Rumah Menggunakan Sensor Sidik Jari [Tugas Akhir]. Semarang: Universitas Negeri Semarang.Program Diploma Teknik Elektro.Hal:9,12,14,15
- [17] Arafat.2016. Sistem Pengaman Pintu Rumah Berbasis Internet of Things (IoT) dengan ESP8266. Technologia.7:264
- [18] Sinukaban, GK.2019.penggunaan Thermocouple Type K pada Oven Pemanggang Kue sebagai Sensor Temperatur Berbasis Mikrokontroller ATmega328 [Tugas Akhir]. Medan: Universitas Sumatera Utara
- [19] Spring Loaded Elektromagnet. Solenoid 12V Pull Type. [Online]. Tersedia: <http://www.engineeringshock.com/12v-pull-type-solenoid.html> [2 Agustus 2012]
- [20] Rajagukguk YG, 2018. Rancang Bangun Sistem Pengingat Lampu Sein Pada Sepeda Motor Berbasis Mikrokontroler ATmega8. [Tugas Akhir]. Medan: Universitas Sumatera Utara, Program Diploma.
- [21] Rahmad Syam, M. H., 2010. Determining the Standard Value of Acquisition Distortion of Fingerprint Images Based on Image Quality. ITB J. ICT Vol. 4, No. 2., 115-132.
- [22] Setianingrum, E.C. dan Purnama, B.E. 2013. "Sistem Pengaman Brankas dengan Menggunakan Handphone Berbasis Mikrokontroler AT89S51". Seminar Riset Unggulan Nasional Informatika dan Komputer FTI UNSA. Vol. 2, No. 1, pp. 1–7.
- [23] Prasetyo, D.E. 2014. Sistem Keamanan Berlapis untuk Lemari Brankas dengan Menggunakan 3 Kombinasi Password. Skripsi. AMIKOM Yogyakarta.
- [24] Yumono, F. 2013. "Sistem Pengamanan Brankas Kantor Perbankan Menggunakan Aktivasi Password Digital Berbasis Mikrokontrol Atmega 8535". Jurnal Cendekia. Vol. 11, No. 1, pp. 42– 45.
- [25] Ariessanti, H.D., Radiyanto dan Yuswanto, A.S. 2015. "Pengaman Brankas Menggunakan Voice dengan Media Bluetooth Berbasis Mikro-kontroller Atmega 328". Techno.COM. Vol. 9, No. 1, pp. 27–32.
- [26] Amit, Manoj Kumar. Design and Implementation of Multiple Output Switch Mode Power Supply. International Jurnal of Engineering Trends and Technology (IJETT), 4 (10). 2013
- [27] Gusmanto. 2016. Rancang Bangun Sistem Peringatan Dini dan Pelacakan Pada Kendaraan Sepeda Motor dengan Menggunakan Mikrokontroler Arduino nano

LAMPIRAN

Lampiran 1 coding pendaftaran fingerprint

```
#include <Adafruit_Fingerprint.h>

#define MODEM_RX 16
#define MODEM_TX 17
#define mySerial Serial2 // use for ESP32

Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);

uint8_t id;

void setup()
{
  Serial.begin(9600);
  while (!Serial); // For Yun/Leo/Micro/Zero/...
  delay(100);
  Serial.println("\n\nAdafruit Fingerprint sensor enrollment");

  // set the data rate for the sensor serial port
  finger.begin(57600);

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

```

uint8_t readnumber(void) {
    uint8_t num = 0;

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

void loop()          // run over and over again
{
    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) { // ID #0 not allowed, try again!
        return;}

    Serial.print("Enrolling ID #");
    Serial.println(id);

    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;
}
}

```

```
// OK success!
```

```

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;
    }
}

// OK success!

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;
}

// OK converted!
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);

```

```

p = finger.storeModel(id);
if (p == FINGERPRINT_OK) {
  Serial.println("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;
}
}

```

Lampiran 2 Coding Program Keseluruhan

```

#define BLYNK_PRINT Serial
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
#include <LiquidCrystal_I2C.h>
#include <Adafruit_Fingerprint.h>
#include <Keypad.h>

#define BLYNK_TEMPLATE_ID "TMPL6F8osfhu"
#define BLYNK_TEMPLATE_NAME "Brankas"

```

```

#define BLYNK_AUTH_TOKEN "EfFyCuIdUx5hZHqBx-Lyo62VBFTuev8M"

#define mySerial Serial2 // use for ESP32

Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);

LiquidCrystal_I2C lcd(0x27, 16, 2);

char ssid[] = "Gear360";
char pass[] = "Aldy123456";

//PIN

#define RedLED 12
#define GreenLED 5
#define buzzerPin 2
#define Solenoid 4

// Initializing pins for keypad
const byte rows = 4;
const byte columns = 4;
byte row_pins[rows] = {26, 25, 27, 14};
byte column_pins[columns] = {13, 23, 19, 18};

char initial_password[6] = {'1', '8', '1', '8', '1', '8'}; // Variable to store initial password
char password[6]; // Variable to store users password
boolean FingerprintMode = true; // boolean to change modes
int FingerprintState = -1;
char key_pressed = 0; // Variable to store incoming keys
uint8_t i = 0; // Variable used for counter

```

```

// Keypad pin map
char hexaKeys[rows][columns] = {
  {'1', '2', '3', 'A'},
  {'4', '5', '6', 'B'},
  {'7', '8', '9', 'C'},
  {'*', '0', '#', 'D'}
};

// Create instance for keypad
Keypad keypad_key = Keypad( makeKeymap(hexaKeys), row_pins, column_pins, rows,
  columns);

void setup() {
  Serial.begin(9600);
  Serial.println ("Begin");
  // Arduino Pin configuration
  pinMode(buzzerPin, OUTPUT);
  pinMode(RedLED, OUTPUT);
  pinMode(GreenLED, OUTPUT);
  pinMode(Solenoid, OUTPUT);

  Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass, "blynk.cloud", 8080);

  lcd.init(); // LCD screen
  lcd.backlight();
  lcd.clear(); // Clear LCD screen

  finger.begin(57600);
  delay(5);
  if (finger.verifyPassword()) {

```

```

    Serial.println("Fingerprint siap diginakan");
}
else{
    Serial.println("Fingerprint eror :( ");
    Serial.println("Coba cek pengkabelan ");
    while (1) { delay(1); }
}
finger.getTemplateCount();
Serial.print("Terdapat "); Serial.print(finger.templateCount); Serial.println(" sidik jari
yang terdaftar");

}

void loop() {
    // System will first look for mode
    Blynk.run();
    if (FingerprintMode == true) {
        Blynk.virtualWrite(V0, 0);
        Serial.println("Silahkan tempelkan jari anda");
        lcd.setCursor(0, 0);
        lcd.print(" Tempelkan");
        lcd.setCursor(0, 1);
        lcd.print(" Jari Anda ");
        FingerprintState = getFingerprintIDez();

        //Checking the card
        if (FingerprintState > 0){
            // If Finger is matched.
            Serial.println("Sidik Jari Cocok");
            lcd.clear();

```

```

lcd.print("Jari Cocok");
digitalWrite(GreenLED, HIGH);
digitalWrite(buzzerPin, HIGH);
delay(1000);
digitalWrite(GreenLED, LOW);
digitalWrite(buzzerPin, LOW);
lcd.clear();
lcd.print("Masukan Password:");
Serial.println("Masukkan Password:");
lcd.setCursor(0, 1);
FingerprintMode = false; // Make Finger mode false
FingerprintState = 0;
} else if(FingerprintState == -3) {
    // If UID of tag is not matched.
    Blynk.virtualWrite(V0, 1);
    lcd.clear();
    lcd.setCursor(0, 0); lcd.print("Sidik jari");
    lcd.setCursor(0, 1); lcd.print("Tidak Cocok");
    Serial.println("Sidik Jari Tidak Cocok");
    digitalWrite(buzzerPin, HIGH);
    digitalWrite(RedLED, HIGH);
    delay(3000);
    digitalWrite(buzzerPin, LOW);
    digitalWrite(RedLED, LOW);
    lcd.clear();
}
}

// If Finger mode is false, it will look for keys from keypad

```



```

if (FingerprintMode == false) {
  key_pressed = keypad_key.getKey(); // Storing keys
  if (key_pressed){
    password[i++] = key_pressed; // Storing in password variable
    lcd.print("*");
    Serial.print(password);
  }

  if (i == 6) { // If 6 keys are completed
    delay(200);
    if (!(strcmp(password, initial_password, 6))){ // If password is matched
      Serial.println("Password Benar");
      lcd.clear();
      digitalWrite(GreenLED, HIGH);
      lcd.setCursor(0, 1);
      Serial.println();
      Serial.println("Pintu Terbuka");
      lcd.print("Pintu Terbuka");
      digitalWrite(Solenoid, HIGH);
      Serial.print("Solenoid ON");
      delay(3000);
      digitalWrite(Solenoid, LOW);
      Serial.println("Pintu Terkunci");
      digitalWrite(GreenLED, LOW);
      lcd.clear();
      i = 0;
      FingerprintMode = true; // Make Finger mode true
    } else { // If password is not matched
      Blynk.virtualWrite(V0, 1);
    }
  }
}

```

```

    lcd.clear();

    lcd.print("Password Salah");
    Serial.println("Password Salah");
    digitalWrite(buzzerPin, HIGH);
    digitalWrite(RedLED, HIGH);
    delay(3000);
    digitalWrite(buzzerPin, LOW);
    digitalWrite(RedLED, LOW);

    lcd.clear();

    i = 0;

    FingerprintMode = true; // Make Finger mode true
}
}
}
}

```

```

uint8_t getFingerprintID() {
    uint8_t p = finger.getImage();
    switch (p) {
        case FINGERPRINT_OK:
            Serial.println("Image taken");
            break;
        case FINGERPRINT_NOFINGER:
            Serial.println("No finger detected");
            return p;
        case FINGERPRINT_PACKETRECEIVEERR:
            Serial.println("Communication error");
            return p;
        case FINGERPRINT_IMAGEFAIL:

```

```

Serial.println("Imaging error");

return p;

default:

Serial.println("Unknown error");

return p;

}

// OK success!

p = finger.image2Tz();

switch (p) {

case FINGERPRINT_OK:

Serial.println("Image converted");

break;

case FINGERPRINT_IMAGEMESS:

Serial.println("Image too messy");

return p;

case FINGERPRINT_PACKETRECIIVEERR:

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;

}

```

```

// OK converted!

p = finger.fingerFastSearch();
if (p == FINGERPRINT_OK) {
    Serial.println("Found a print match!");
}
else if (p == FINGERPRINT_PACKETRECEIVEERR) {
    Serial.println("Communication error");

    return p;
}
else if (p == FINGERPRINT_NOTFOUND) {
    Serial.println("Did not find a match");

    return p;
}
else {
    Serial.println("Unknown error");

    return p;
}

// found a match!

Serial.print("Found ID #"); Serial.print(finger.fingerID);

Serial.print(" with confidence of "); Serial.println(finger.confidence);

return finger.fingerID;
}

int getFingerprintIDez() {
    uint8_t p = finger.getImage();

    if (p != FINGERPRINT_OK) return -1;

    p = finger.image2Tz();
}

```

```

if (p != FINGERPRINT_OK) return -2;

p = finger.fingerFastSearch();

if (p != FINGERPRINT_OK) return -3;

// found a match!

Serial.print("Terdeteksi jari no : "); Serial.print(finger.fingerID);

Serial.print(" Dengan keakuratan : "); Serial.print(finger.confidence-8);

Serial.println("%");

if (finger.confidence>90){

}

return finger.fingerID;

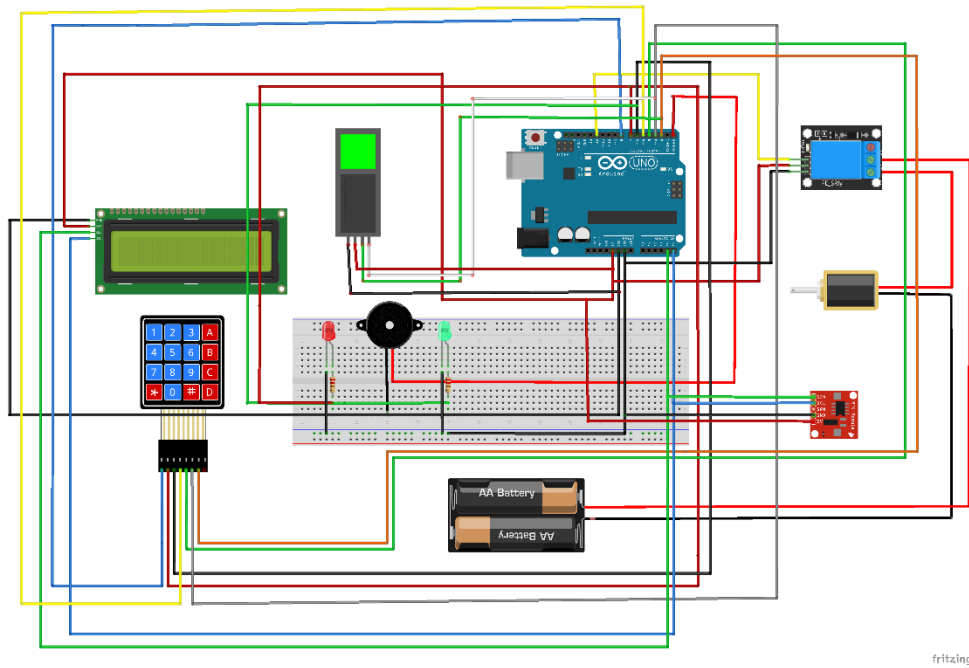
}

```

Lampiran 3 Gambar Alat Keseluruhan



Lampiran 4 Rangkaian Alat Keseluruhan



Lampiran 5 Proses Pengujian Rangkaian Alat Keseluruhan

