BAB I

PENDAHULUAN

A. Latar Belakang

Pada era globalisasi saat ini perkembangan di bidang teknologi sangat pesat dan telah berpengaruh terhadap kehidupan manusia. Hal ini bisa kita lihat dengan banyaknya kemudahan yang ditawarkan dari perkembangan teknologi. Dengan itu maka diperlukan sumber daya manusia yang mampu memanfaatkan kemajuan teknologi tersebut, supaya tidak ketinggalan dengan kemajuan teknologi. Perkembangan teknologi tersebut bisa kita lihat darim banyaknyapemanfaatan arduino pada alat elektronik yang penggunaanya menjadi otomatis untuk membantu pekerjaansehari-hari. Begitu juga halnya dengan keamanan saat ini banyak pengembangan arduino untuk keamanan atau security seperti halnya di suatu universitas, perguruan tinggi, instansi pendidikan atau perusahaan yang membutuhkan keamanan unuk sebuah ruangan atau tempat tertentu

Saat ini telah banyak dikembangkan sebuah sistem pengamanan akses masuk ke sebuah rumah atau ruangan dengan beberapa verifikasi identitas dengan sistem komputer, baik dengan menggunakan kunci, kartu, *password*, dan sebagainya. Keamanan yang berkembang saat ini seperti pengenalan sidik jari, pengenalan retina mata, pengenalan iris mata, dan sebagainya mengharuskan seseorang memposisikan tubuh mereka pada posisi tertentu yang sesuai

dengan posisi sensor ataupun kamera yang membuat teknologi ini terkesan kaku. Teknologinya mengharuskan orang tersebut untuk tidak bergerak dalam beberapa waktu tertentu selama proses identifikasi untuk membuat sistem pembacaannya akurat. Untuk itu diperlukan sebuah sistem identifikasi yang lebih fleksibel dan bersifat otomatis yang dapat mencegah kejahatan. Oleh karena itu tugas akhir ini akan membuat *Prototype* sistem keamanan akses pintu masuk menggunakan pendeteksi wajah berbasis *Internet Of Things* untuk mencari titik pada wajah yang nantinya akan dilakukan pencocokan antara *Sample* dan database. Dari latar belakang yang telah dijelaskan diatas, maka penulis memilih judul **"RANCANG BANGUN *PROTOTYPE* SISTEM KEAMANAN AKSES PINTU MASUK MENGGUNAKAN PENDETEKSI WAJAH BERBASIS *INTERNET OF THINGS*"**

B. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan sebelumnya maka didapatkan rumusan masalah yaitu bagaimana merancang dan membangun *Prototype* sistem keamanan akses pintu masuk mengunakan pendeteksi wajah berbasis *Internet Of Things*?

C. Batasan Masalah

Untuk menghindari pembahasan diluar permasalahan, berikut ini adalah batasan permasalahan pada perancangan *Prototype* sistem keamanan akses pintu masuk mengunakan pendeteksi wajah berbasis *Internet Of Things*:

1. Perangkat yang di buat masih berbentuk *Prototype*

2. Bahasa pemrograman yang digunakan pada *arduino* adalah bahasa C dan untuk membangun sistem keamanan menggunakan bahasa pemrograman *PHP*
3. Sistem yang dibuat terkoneksi jaringan internet

D. Tujuan Penelitian

Adapun tujuan yang hendak dicapai dalam penyusunan ini adalah untuk merancang dan membuat elektronika *Prototype* menggunakan pendeteksi wajah berbasis *Internet Of Things*

E. Manfaat Penelitian

1. Manfaat bagi penulis.

Agar penulis dapat mengaplikasikan ilmu teori yang telah didapatkan dibangku perkuliahan, serta dapat menambah pengetahuan mengenai rancang bangun *Prototype*sistem keamanan akses pintu masuk mengunakan pendeteksi wajah berbasis *Internet Of Things* secara mendalam.

2. Manfaat bagi masyarakat.

Diharapkan dapat menjadi *Alternatif* bagi masyarakat agar selalu tetap waspada akan kejahatan.

F. Sistematika Penulisan

Dalam menyusun sistematika penulisan, penulis menguraikan ke dalam lima bab yaitu :

BAB I : PENDAHULUAN

Menguraikan tentang latar belakang penulisan, rumusan masalah, tujuan penelitian, manfaat penelitian,dan batasan masalah, dan juga sistematika penulisan.

BAB II :TINJAUAN PUSTAKA

Pada Bab ini berisi uraian - uraian tentang tinjauan penelitian sebelumnya dan teori - teori yang digunakan sebagai referensi dalam rancang bangun *Prototype* sistem keamanan akses pintu masuk mengunakan pendeteksi wajah berbasis *Internet Of Things*

BAB III: METODE PENELITIAN

Membahas tentang waktu dan tempat penelitian, metode penelitian, jenis penelitian, alat dan bahan yang digunakan dalam penelitian

BAB IV: PERANCANGAN DAN IMPLEMENTASI APLIKASI

Bab ini membahas tentang Analisis Sistem Dengan DFD, Perancangan aplikasi, Implementasi, dan Pengujian sistem.

BAB V : PENUTUP

Membahas tentang kesimpulan dan saran.

BAB II

TINJAUAN PUSTAKA

A. Tinjauan Hasil Penelitian Sebelumnya

1. (Muhammad, 2018) Pada peneltian yang berjudul **"Sistem Keamanan Akses Pintu Masuk Menggunakan *Face Recognition* Berbasis Raspaberry Pi 3"**. Pada peracangan sistem ini digunakan metode *SURF* untuk mencari titik penting dari citra *Sample* dan citra *database* untuk dilakukan pencocokan citra. Dalam pembuatan sistem ini menggunakan bahasa pemrograman *phyton*.
2. (Mifta & Arifin, 2016) Penelitian ini berjudul **"Pengaman Laptop Menggunakan Pengenalan Wajah Berbasis *Triangel Face*"**. Dalam peneltian ini berfokus pada keamanan untuk mengakses *laptop*. Sistem pencocokan citra menggunakan *triangel face* yaitu ukuran jarak antar mata, hidung dan mulut.
3. (Piarsa & Wibawa, 2017) Penelitian ini berjudul **"*Prototipe* Deteksi dan Pengenalan Wajah pada Sistem *Monitoring* dan Kontrol *Visual* Keamanan Rumah"** penelitian ini berfokus pada sistem *monitoring* keamanan rumah yang dapat mendeteksi wajah yang sudah terdaftar di *database*. Sistem ini juga menggunakan perangkat *Raspaberry* pi serta *library open* CV.

Sistem keamanan akses pintu ini menggunakan deteksi wajah sebagai sistem keamanan agar dapat maengatasi masalah yang sering terjadi seperti lupa letak menyimpan kunci, kunci tercecer dan penyalahgunaan kunci oleh orang yang tidak bertanggung jawab.

B. Kajian Teori

1. *Prototype*

(Fauzan & Adiputri, 2019) *Prototype* adalah salah satu pendekatan dalam rekayasa perangkat lunak yang secara langsung mendemonstrasikan bagaimana sebuah perangkat lunak atau komponen – komponen perangkat lunak akan bekerja dalam lingkungannya sebelum tahapan konstruksi aktual dilakukan

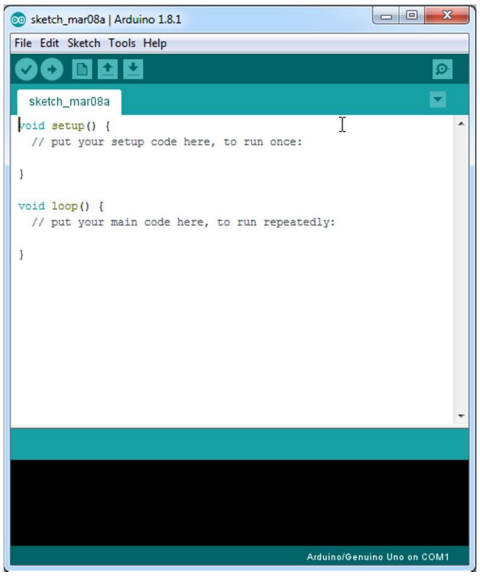
Selain itu juga *Prototype* adalah tahapan yang ditunjukan untuk mentransformasi sifat – sifat abstrak dari sebuah ide menjadi lebih berwujud. Tahapan ini tidak hanya berupa proses visualisasi ide tetapi juga proses pembangunan ide. Secara umum, *Prototype* memiliki dua kategori: *low-fidelity* dan *high-fidelity*. Proses *Prototype* yang digunakan di dalam sistem keamanan pintu masuk berbasis *IOT* ini adalah *low-fidelity* atau *Rapid Prototyping* karena proses pembuatan yang cepat, mudah, murah dan *basic*.

2. *IDE Arduino*

(Alam, Parinduri, Hutagalung, Hutagalung, & Masri, 2020) *IDE Arduino* adalah *Software* yang sangat canggih yang akan digunakan sebagai *editor Script* dalam pembuatan sistem keamanan pintu masuk menggunakan pedeteksi wajah berbasis *IOT*. *IDE Arduino* terdiri dari:

1. *Editor* Program, sebuah *window* yang memungkinkan pengguna menulis dan meng*edit* program dalam bahasa *Processing*.
2. *Compiler*, sebuah modul yang mengubah kode program (bahasa *Processing*) menjadi kode *biner*. Bagaimana sebuah *microcontroller* tidak akan bisa memahami bahasa *Processing*. Yang bisa dipahami oleh *microcontroller* adalah *code* biner. Itulah sebabnya compiler diperlukan dalam hal ini.

   Uploader, sebuah modul yang memuat kode *biner* dari komputer ke dalam *memory* di dalam papan *Arduino*



Gambar 2.1 *IDE Arduino*

3. Esp32cam

(Wicaksono & Rahmatya, 2020) Modul *ESP32-Cam* adalah modul kamera yang dilengkapi dengan *wifi* dan *bluetooth*. Modul ini yang akan digunakan dalam pengambilan gambar wajah untuk keamanan akses pintu masuk berbasis *IOT*. Harganya sangat murah sehingga peminatnya sangat

banyak, modul ini sangat cocok untuk projek *IOT* sehingga banyak aplikasi *IOT* menggunakan modul kamera ini, misalkan perangkat rumah pintar, kontrol nirkabel industri, sistem keamanan, identifikasi kode *QR*, dan aplikasi *IOT* lainya.



Gambar 2.2 Esp32-Cam

Tabel 2. 1 Spesifikasi ESP32-Cam

Module Model	*ESP 32-CAM*
Package	DIP-16
Size	27*40.5*4.5 (±0.2) mm
SPI Flash	*Default* 32*Mbit*
RAM	*Internal* 520*KB + External* 4*M PSRAM*
Bluetooth	*Bluetooth* 4.2 *BR/EDR and BLE standard*
Wi-fi	802.11 *b/g/n/e/i*
Support Interface	*UART, SPI, I2C, PWM*
Support TF Card	*Maximum support 4G*
IO Port	9
Serial Port Rate	*Default 115200 bps*
Image Output Format	*JPEG (Only OV*2640 *support), BMP, GRAYSCALE*
Spectrum Range	2412 ~ 2484MHz
Antenna Form	*Onboard PCB antenna, gain 2dBi*
Transmit Power	802.11b: 17 ± 2 *dBm (@*11*Mbps)* 802.11g: 14 ± 2 *dBm (@*54*Mbps)* 802.11n: 13 ± 2 *dBm (@MCS7)*
Safety	*WPA/WPA2/WPA2-Enterprise/WPS*
Power Supply Range	*5V*
Operating Temperature	-20 ℃ ~ 85 ℃
Storage Environment	-40 ℃~ 90℃, < 90%RH

4. Kabel *Jumper*

(Razor, 2020) Kabel *Jumper* menjadi salah satu komponen yang sangat penting dalam membuat rangkaian. Kabel *Jumper* adalah kabel elektrik yang memiliki pin konektor di setiap ujungnya untuk menghubungkan dua komponen tanpa harus memerlukan solder. Konektor pada kabel *Jumper* terdapat dua jenis yaitu konektor jantan (*male connector*) dan konektor betina (*female connector*).Ukuran panjang kebel *Jumper* juga bervarian. Seperti yang diketahui bahwa kabel *Jumper* memiliki beberapa warna yang berbeda, perbedaan warna tidak berpengaruh pada fungsi kabel perbedaan warna tersebut hanya untuk memudahkan saat membuat rangkaian. Pada pembuatan *Prototype* sistem keamanan pintu masuk menggunakan pendeteksi wajah yang dibuat oleh penulis, kabel *Jumper* yang digunakan adalah kabel *Jumper* jenis *Male to Female.*



Gambar 2.3 Kabel *JumperMale to Female*

Tabel 2. 2 Spesifikasi Kabel *Jumper*

Jenis	*Male To Female*
Panjang Kabel	±10 Cm
Ukuran *Pitch*	2.54 mm

5. *Relay Module*

(Malliwang, 2020) *Relay* adalah Saklar (*Switch*) yang dioperasikan secara listrik dan merupakan komponen *Electromechanical* yang terdiri dari 2 bagian utama yakni Elektromagnet (*Coil*) dan Mekanika (seperangkat Kontak Saklar/*Switch*). *Relay* menggunakan Prinsip Elektromagnetik untuk menggerakan Kontak Saklar sehingga dengan arus listrik yang kecil (*low power*) dapat menghantarkan listrik bertegangan lebih tinggi. Modul *Relay* sendiri dapat digunakan sebagai *Switch* untuk menjalankan berbagai peralatan elektronik. Misalkan Lampus listrik, Motor listrik, dan sebagai peralatan elektronik lainnya. *Relay* yang digunakan pada pembuatan sistem keamanan pintu masuk menggunakan deteksi wajah yaitu *Relay Module* 1 *channel* 5v.



Gambar 2.4 *Relay Module* 1*Channel*

Tabel 2. 3 Spesifikasi *Relay Module*

Bahan	Papan Sirkuit
Model	1 *channel*
Sinyal Kontrol	TTL
Nilai Beban	AC125 ~ 250V/10A, DC28 ~ 30V/10A
Max. Switch tegangan	250VAC 30 V
Memicu Tegangan	0-1.5 V (RENDAH); 3-5 V (Tinggi)
Pemicu saat ini	5mA
Arus Maksimum	90mA
Ukuran	50mm * 26mm * 18.5mm (1.9 inci * 1.02 inci * 0.72 inci)

6. Adaptor

Adaptor adalah sebuah rangkaian yang berguna untuk mengubah tegangan AC yang tinggi menjadi DC yang rendah. Dan *output* dari adaptor bisa di atur sesuai kebutuhannya misalkan 3V, 4.5V, 5V, 9V, 12V dan seterusnya. Sedangkan yang digunakan dalam perencanaan sistem keamanan pintu masuk adalah adaptor dengan tegangan 12V.

Adapun rincian fungsi dari adaptor yaitu :

a. Adaptor yang kita kenal kebanyakan yaitu mengubah dari listrik PLN 220 *Volt* (Arus AC) menjadi tegangan 5 *Volt* DC, 12 Volt DC, 19 Volt DC, 24 *Volt* DC dan sebagainya tergantung keperluan perangkat apa yang digunakan.
b. Ada juga adaptor yang mengubah dari listrik PLN 20 Volt AC menjadi tegangan listrik lebih kecil namun arusnya tetap AC, misalnya menjadi 9 *Volt* AC, atau 24 *Volt* AC.
c. Adaptor disebut juga charger.



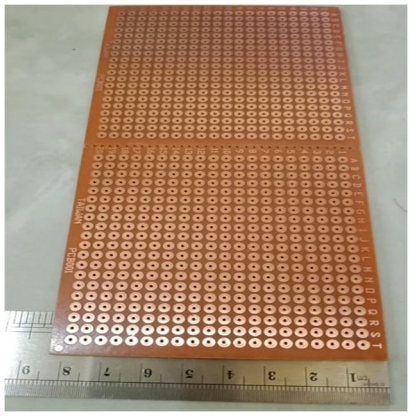
Gambar 2. 5 *Relay Module* 1*Channel*

Tabel 2. 4 Spesifikasi Adaptor

Model	LIN601-1G-12C20
Input	AC100 ~ 240V 50/60Hz
Output	DC12V – 2A

7. Papan PCB

(Hakiem, 2015)Papan PCB adalah sebuah papan yang digunakan untuk mendukung semua komponen-komponen elektronika yang berada diatasnya, papan PCB juga memiliki jalur-jalur konduktor yang terbuat dari tembaga dan berfungsi untuk menghubungkan antara satu komponen dengan lainnya.



Gambar 2.6 Papan PCB

Tabel 2. 5 Spesifikasi Papan PCB

Jenis	Bolong/Berlubang
Ukuran	16.4 cm x 8.4 cm

Bahan yang digunakan untuk membuat PCB adalah jenis fiber sebagai media isolasinya yang dilapisi cat berwarna hijau, sedangkan untuk jalur konduktor menggunakan tembaga. Ada beberapa macam jenis PCB menurut kegunaannya yaitu PCB 1 *side* (bisa digunakan pada rangkaian

elektronika seperti radio, TV, dll) PCB *double side* (maksudnya kedua sisi PCB digunakan untuk menghubungkan komponen) dan PCB *multi side* (bagian PCB luar maupun dalam digunakan sebagai media penghantar, Misalnya pada rangkaian-rangkaian PC) jadi kita tidak perlu melubangi pcb tinggal langsung menyoldernya saja.

8. Lampu *LED*

Light Emittng Diode atau sering disingkat dengan *LED* adalah komponen elektronika yang dapat memancarkan cahaya monokromatik ketika diberikan tegangan maju. *LED* merupakan keluarga *Dioda* yang terbuat dari bahan semiknduktor. Warna-warna cahaya yang dipancarkan oleh *LED* tergantung pada jenis bahan semikonduktor yang dipergunakannya. *LED* memiliki dua kutub yaitu kutub *Positif* (P) dan kutub *negatif* (N). *LED* hanya memancarkan cahaya apabila dialiri tegangan maju (bias *forward*) dari *Anoda* menuju *katoda*.

Lampu *LED* pada pembuatan *Prototype* sistem keamanan pintu masuk menggunakan deteksi wajah berbasis iot ini memakai 2 lampu *LED* yaitu *LED* berwarna Kuning dan *LED* berwarna Merah. Jika lampu *LED* berwarna merah yang menyala menandakan bahwa kunci pintu sedang terbuka dan jika *LED* berwarna kuning yang menyala menandakan bahwa pintu sedang terkunci.



Gambar 2.7 Lampu *LED*

Tabel 2. 6 Spesifikasi Papan PCB

Nama	Lampu LED
Ukuran	5mm

9. *Solenoid Door Lock*

(Ariandi, Yesputra, & Risnawati, 2021) *Solenoid door lock* adalah salah satu *Solenoid* yang difungsikan khusus sebagai *Solenoid* untuk pengunci pintu secara elektronik. *Solenoid* akan bekerja sebagai pengunci dan akan aktif ketika diberi tegangan. *Solenoid* ini mempunyai dua sistem kerja yaitu *normalyclose* (NC) dan normal *open* (NO). Perbedaannya adalah jika cara kerja *Solenoid* nc apabila diberi tegangan, maka *Solenoid* akan memanjang (tertutup) dan sebaliknya untuk sistem kerja *Solenoid* no.



Gambar 2.8 *Solenoid Door Look*

Tabel 2. 7 Spesifikasi *Solenoid Door Lock*

Panjang Baut	15 mm
Berat	0.17 kg

10. Pemrograman Bahasa C

Bahasa Pemrograman C adalah sebuah bahasa pemrograman komputer yang bisa digunakan untuk membuat berbagai aplikasi (*general-purpose programminglanguage*), mulai dari sistem operasi (seperti *Window*s atau *Linux*), *antivirus*, *Software* pengolah gambar (*imageProcessing*), hingga *compiler* untuk bahasa pemrograman, dimana C banyak digunakan untuk membuat bahasa pemrograman lain yang salah satunya adalah *PHP*.

Meskipun termasuk *general-purpose programming language*, yakni bahasa pemrograman yang bisa membuat berbagai aplikasi, bahasa pemrograman C paling cocok merancang aplikasi yang berhubungan langsung dengan Sistem Operasi dan *Hardware.* Ini tidak terlepas dari tujuan awal bahasa C dikembangkan.

```
#include <stdlib.h>
#include <stdio.h>

int main()
{
   int r;
   float phi, luas;
   phi=3.14;

   printf("Masukan jari-jari lingkaran (cm) :",r);
   scanf("%i",&r);

   luas=phi*r*r;

   printf("Jadi luas lingkaran tersebut adalah :%f \n",luas);
   system ("pause");
   return 0;
}
```

Gambar 2.9 Contoh Pemrograman C

Ritchie membuat bahasa pemrograman C untuk mengembangkan sistem operasi *UNIX*. Sebelumnya, sistem operasi *UNIX* dibuat menggunakan bahasa *assembly* (*assemblylanguage*). Akan tetapi bahasa *assembly* sendiri sangat rumit dan susah untuk dikembangkan.

Berikut beberapa fitur serta keunggulan bahasa pemrograman C jika dibandingkan dengan bahasa pemrograman lain:

a. **C sebagai bahasa pemrograman Prosedural**

   Konsep pemrograman prosedural adalah sebuah metode pemrograman yang setiap baris perintah diproses secara berurutan dari baris paling atas hingga baris paling bawah. Selain itu bisa terdapat fungsi tambahan (*function*) yang digunakan untuk menyelesaikan berbagai tugas.

b. **Bahasa C sangat cepat dan efisien**

   Aplikasi yang dibuat dengan bahasa C bisa di eksekusi dengan sangat cepat serta berukuran kecil. Ini karena C bisa langsung berkomunikasi dengan *Hardware*, sebuah fitur yang jarang tersedia di bahasa pemrograman *modern* seperti *JAVA*, *PHP*, maupun *Phyton*.

c. **C adalah portabel *language***

   Bahasa pemrograman C bisa di-*compile* ulang supaya berjalan di berbagai sistem operasi tanpa perlu mengubah kode-kode yang ada. Aplikasi yang dibuat di *Window*s dengan bahasa C, bisa dipindahkan ke *Linux* dengan sedikit atau tanpa modifikasi.

d. **C merupakan "induk" dari bahasa pemrograman *modern***

Bahasa pemrograman C banyak menginspirasi bahasa pemrograman lain, seperti C++, C#, *Objective* C, *PHP*, *JAVA*, *JavaScript* dan masih banyak lagi. Dengan mempelajari bahasa C, anda akan *familiar* dan lebih mudah saat berpindah ke bahasa pemrograman lain yang merupakan turunan C.

11. *UML (Unified Modelling Language)*

UML (Unified Modelling Language) menurut menurut Adi nugroho (2010:6), "*Unified Modelling Language* adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berpradigma berorientasi objek". Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami.

Metode *Unified Modelling Language* (*UML*) menggunakan tiga bangunan dasar untuk mendeskripsikan sistem atau perangkat lunak yang akan dikembangkan, yaitu:

1. Sesuatu (*things*)

Ada empat *things* dalam *Unified Modelling Language* (*UML*):

a. *Structural things*, bagian yang relatif statis dapat berupa elemen-elemen yang bersifat fisik maupun konseptual.
b. *Behavorial things,* bagian dinamis biasanya merupakan kata kerja dari model *UML* yang mencerminkan perilaku sepanjang waktu
c. *Grouping things,* bagian pengorganisasian dalam *UML*. Dalam penggambaran model *UML* yang rumit diperlukan penggambaran paket yang menyederhanakan model. Paket-paket ini kemudian dapat didekomposisi

lebih lanjut. Paket berguna bagi pengelompokan sesuatu, misalnya model-model serta *subsitem-subsistem*.

d. *An notational things*, merupakan bagian yang meperjelas model *UML*. Dapat berisi komentar yang menjelaskan fungsi serta ciri-ciri tiap element dalam model *UML*.

2. Relasi (*relationship*)

Ada empat *relationship* (hubungan) dalam *Unified Modelling Language* (*UML*):

a. Ketergantungan (d*ependency*) adalah hubungan dimana perubahan yang terjadi pada suatu elemn *independent* akan mempengaruhi elemen yang bergantung padanya.

b. *Asosiasi* adalah apa dan bagaimana yang mengubungkan antara objek satu dengan yang lainnya. Suatu bentuk asosiasi adalah agregasi yang menampilkan hubungan suatu objek dengan bagian-bagiannya.

c. *Generalisasi* adalah hubungan dimana objek anak berbagai perilaku dan struktur data dari objek yang ada diatasnya (objek induk). Arah dari objek induk ke objek anak dinamakan spesialisasi sedangkan arah sebaliknya dinamakan generalisasi.

d. Realisasi adalah operasi yang benar-benar dilakukan oleh suatu objek.

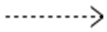
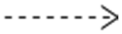
3. Diagram, *Unified Modelling Language* (*UML*) menyediakan Sembilan jenis diagram yang dapat dikelompokkan berdasarkan sifatnya (statis dan dinamis).

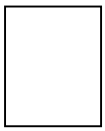
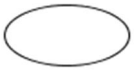
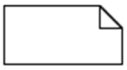
1. diagram *use case*

Diagram *use case* menyajikan interaksi antara *use case* dan aktor, dimana aktor dapat berupa orang, peralatan, atau sistem lain yang berinteraksi dengan sistem yang sedang di bangu. *Use case* menggambarkan fungsionalitas sistem atau persyaratan-persyaratan yang harus dipenuhi sistem dari pandangan pemakai.(*Sholiq, 2006*)

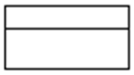
Adapun simbol-simbol *Use Case* Diangram antara lain :

Tabel 2. 8 Simbol *Use Case Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		*Actor*	Menspesifikasikan himpuan peran yang pengguna mainkan ketika berinteraksi dengan *use case*.
2		*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (*independent*).
3		*Generalization*	Hubungan dimana objek anak (*descendent*) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*).
4		*Include*	Menspesifikasikan bahwa *use case* sumber secara *eksplisit*.
5		*Extend*	Menspesifikasikan bahwa *use case* target memperluas perilaku dari *use case* sumber pada suatu titik yang diberikan.
6		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya.

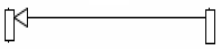
NO	GAMBAR	NAMA	KETERANGAN
7		*System*	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		*Use Case*	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
9		*Collaboration*	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan prilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		*Note*	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

Tabel 2. 9 *Simbol Class Diagram*

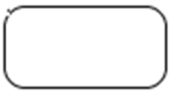
NO	GAMBAR	NAMA	KETERANGAN
1		*Generalization*	Hubungan dimana objek anak *(descendent)* berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*).
2		*Nary Association*	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		*Class*	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.

NO	GAMBAR	NAMA	KETERANGAN
4		*Collaboration*	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		*Realization*	Operasi yang benar-benar dilakukan oleh suatu objek.
6		*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri

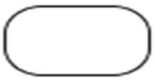
Tabel 2. 10 Simbol *Sequence Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		*LifeLine*	Objek *entity*, antarmuka yang saling berinteraksi.
2		*Message*	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3		*Message*	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

Tabel 2. 11 Simbol *State Chart Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		*State*	Nilai atribut dan nilai *link* pada suatu waktu tertentu, yang dimiliki oleh suatu objek.
2		*Initial Pseudo State*	Bagaimana objek dibentuk atau diawali
3		*Final State*	Bagaimana objek dibentuk dan dihancurkan
4		*Transition*	Sebuah kejadian yang memicu sebuah state objek dengan cara memperbaharui satu atau lebih nilai atributnya
5		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya.
6		*Node*	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

Tabel 2. 12 Simbol *Actifity Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		*Actifity*	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		*Action*	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		*Initial Node*	Bagaimana objek dibentuk atau diawali.
4		*Actifity Final Node*	Bagaimana objek dibentuk dan dihancurkan
5		*Fork Node*	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

12. *Flowchart*

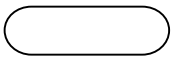
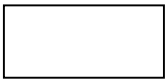
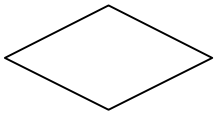
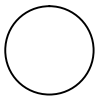
Flowchart atau bagan alir adalah suatu bentuk grafik atau diagram dari algoritma dimana simbol-simbol standar mewakili tampilan operasi yang perlu dan memperlihatkan urutan pelaksanaanya. Langkah-langkah dan elemen-elemen aktivitas pengolahan data dapat digambarkan dengan suatu bagan alir. *Flowchart* menggambarkan urutan logika dari suatu prosedur pemecahan masalah, sehingga *Flowchart* merupakan langkah-langkah penyelesaian masalah yang dituliskan dalam simbol-simbol tertentu.

Gambar atau skema bagan alir akan membantu analis memahami prosedur kerja dengan baik juga digunaka untuk membantu menganalisis dan mengembangkan prosedur itu sendiri. *Flowchart* ini selain dibutuhkan sebagai alat komunikasi, juga diperlukan sebagai dokumentasi. Adapun aturan-aturan dalam perancangan diagram alir tersebut, yaitu:

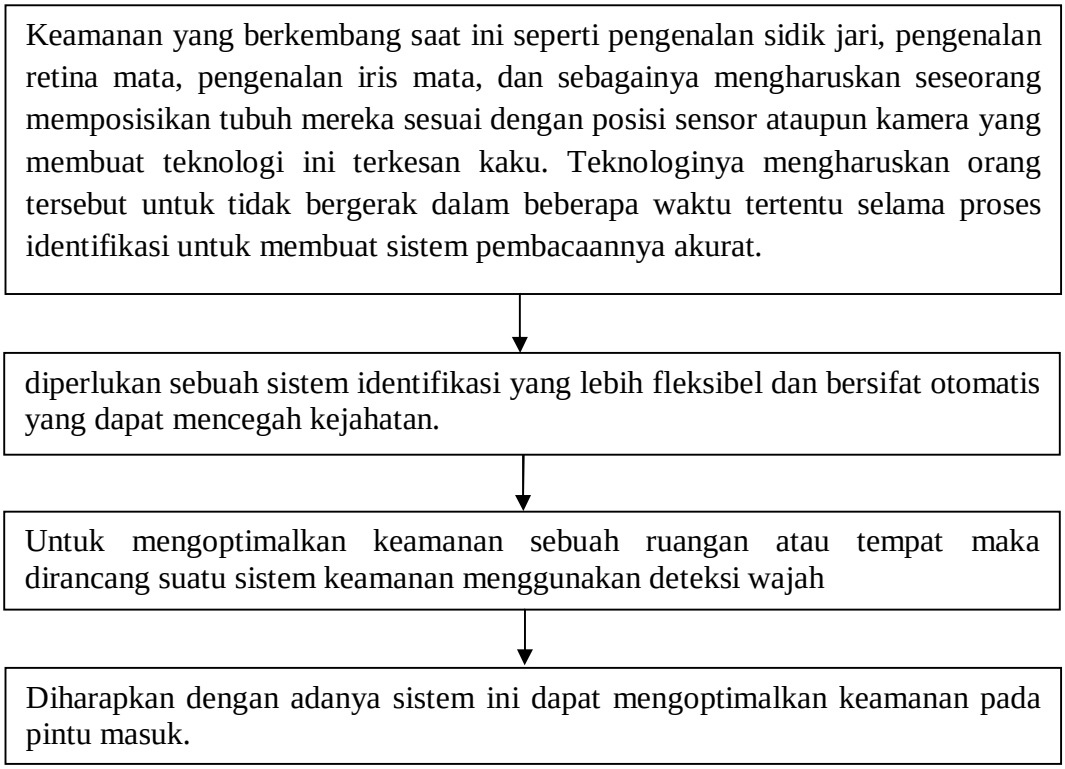
1. Diagram alir digambarkan dengan orientasi dari atas kebawah dan dari kiri ke kanan.
2. Setiap kegiatan.proses dalam diagram alir harus dinyatakan secara eksplisit.
3. Setiap diagram alir harus dimulai dari satu start *state* dan berakhir pada satu atau lebih terminal akhir/*terminator*/*haltstate*.
4. Gunakan *connector* dan *off-pageconnector state* dengan label yang sama untuk menunjukkan keterhubungan antarpath algoritma yang terputus/terpotong, misalnya sebagi akibat/ganti halaman.

Tujuan dari *Flowchart* adalah untuk menggambarkan suatu tahapan penyelesaian masalah secara sederhana, terurai, rapi dan jelas menggunakan simbol-simbol standar.

Tabel 2.13 Simbol-simbol *Flowchart*

No	Simbol-simbol	Keterangan simbol
1		Terminal symbol (*terminal symbol*), digunakan untuk menunjukkan awal dan akhir dari program
2		Simbol persiapan (*Preparation symbol*), digunakan untukmemberikan nilai awal pada suatu *variable*
3		Simbol pengolahan*(Processing symbol)* digunakan untuk pengolahan arthmatika dan pemindahan data.
4		Simbol keputusan *(Decision symbol),* digunakan untuk mewakili operasi perbandingan logika
5		Simbol proses terdefinsi *(predefined process symbol)* digunakan untuk proses yang detailnya dijelaskan terpisah, misalnya dalam bentuk *subroutne*.
6		Simbol penghubung *(Connector symbol),* digunakan untuk menunjukkan hubungan arus proses yang terputus masih dalam halaman yang sama.

C. Kerangka Pikir



BAB III

METODE PENELITIAN

A. Jenis Penelitian

Jenis penelitian yang digunakan pada Rancang Bangun *Prototype* Sistem Keamanan Pintu Masuk Menggunakan Pendeteksi Wajah Berbasis *IOT* adalah :

1. Penelitian Deskriptif

   Penelitian Deskriptif memberikan gambaran fenomena yang sesungguhnya terjadi dalam menggambarkan fakta-fakta dan informasi secara sistematis.

2. Penelitian Kepustakaan

   Penelitian Kepustakaan adalah jenis penelitian dengan mencari informasi terkait masalah pada buku, catatan, serta berbagai laporan untuk mencocok dan mendapatkan pemecahan dari masalah yang dikaji.

B. Metode Pengumpulan Data

Untuk memperoleh data-data yang dibutuhkan dalam rangka melakukan penelitian, maka penulis mengumpulkan data melalui beberapa cara yaitu :

1. Pengumpulan Data

   Metode pengumpulan data menggunakan Metode Kepustakaan yaitu metode atau teknik pengumpulan data yang bersumber dari literatur buku-buku penunjang dan jurnal untuk konsep teori yang berhubungan dengan objek permasalahan penelitian.

2. Analisis Data

   Menganalisa data-data yang sebelumnya telah dikumpulkan dengan caramenganalisis cara kerja sistem yang akan dirancang, mengidentifikasi masalah, dan menganalisa kebutuhan sistem.

3. Perancangan Program

   Sebagai pedoman dalam penulisan program atau kode-kode agar berjalan sesuai rencana.

4. Uji Coba Program

   Pengujian program dilakukan untuk memastikan bahwa program yang dibuat dapat berjalan dengan baik dengan cara membuat *Flowchart* Sistem dan kemudian akan digunakan untuk membuat Desain Sistem.

5. Evaluasi

   Sistem yang telah selesai dibangun perlu adanya evaluasi untuk menemukan kelemahan yang terdapat pada program yang telah dibangun tadi, yang nantinya bisa digunakan sebagai acuan untuk memperbaiki program sehingga lebih sempurna.

C. Alat dan Bahan

Dalam melakukan penelitian, maka diperlukan alat dan bahan penelitian yang mendukung kegiatan penelitian tersebut. Alat dan bahan yang diperlukan antara lain :

1. *Hardware*

Alat yang digunakan selama proses penelitian yaitu :

a. *Laptop HP* 240 G2 *Notebook PC* dengan spesifikasi:

1. *Processor* : *Intel*(R) *CORE*(TM) i3-3110M *CPU @* 2.40GHz (4 CPUs) ,~2.4 GHz
2. *RAM* : 4 GB
3. *Hard Disk Drive* : 500 GB
4. *Monitor* : 14.0 inci
5. Sistem Operasi : *Windows 10 PRO 64-bit*

2. Software

Software yang digunakan dalam pembuatan aplikasi yaitu :

a. Windows 10

b. *Arduino Uno*

c. *Browser*

D. Tahapan Penelitian

Tahap-tahap penelitian yang dimaksud dalam penelitian ini adalah berkenaan dengan proses pelaksanaan penelitian.

1. Persiapan Penelitian

Tahap persiapan adalah tahap yang dilakukan sebelum melakukan penelitian.Pada tahapan ini dimulai dengan mengkaji permasalahan yang telah ada kemudian melakukan studi literatur mengenai permasalahan yang sedang diteliti.

2. Studi Literatur

Tahap pelaksanaan merupakan tahap penerapan penelitian. Pada tahap ini dilakukan beberapa kegiatan yang mengenai dengan penelitian yang akan dilakukan. Pada tahapan ini ada empat langkah yang harus dipenuhi guna untuk mencapai hasil maksimal dalam penelitian yaitu tahapan pengumpulan data

dengan menggunakan beberapa metode pengumpulan data yang telah dijelaskan pada poin diatas, pengolahan data, analisis data dan selanjutnya penafsiran hasil analisis. Setelah kegiatan tersebut, proses selanjutnya yaitu dengan melakukan tugas lapangan dalam rangka mengumpulkan data, untuk kemudian akan diproses. Proses yang dimaksud meliputi penyuntingan, penerapan masalah dalam aplikasi program, serta analisis sebagai penarikan kesimpulan hasil akhir.

3. Pengumpulan Data

Pada tahap ini peneliti melakukan pencarian data-data di berbagai sumber untuk dikumpulkan lalu dikaji lebih lanjut.

4. Analisis

Pada tahap analisis, peneliti melakukan analisa terhadap permasalahan yang diteliti kemudian merumuskan masalah yang menjadi pokok penelitian sehingga dapat dibuat alternatif pemecahan masalah.

5. Perancangan

Peneliti kemudian merancang aplikasi yang ingin dibuat berdasarkan alternatif pemecahan masalah.

6. Pengujian

Setelah melakukan perancangan, peneliti kemudian menguji hasil perancangan yang telah dibuat.Jika hasil perancangan terdapat kekurangan atau kelemahan maka kembali ke tahap analisis.

7. Implementasi

Setelah pada perancangan tidak terdapat kekurangan maka aplikasi siap untuk di gunakan oleh *user*.

8. Tahap Penyelesaian

Tahap penyelesaian merupakan tahap akhir yang dilakukan pada penelitian yang dilakukan.Pada tahap ini penyusunan laporan penelitian disusun.

E. Jenis Data

Jenis data yang dikumpulkan dalam penelitian ini adalah data primer dan data sekunder yang relevan dengan masalah yang akan dibahas.

1. Data Primer

   Data Primer adalah data yang berasal atau data yang diperoleh langsung dari sumber-sumber yang berada di lapangan yang berasal dari wawancara dengan pihak pihak terkait.

2. Data Sekunder

   Data sekunder adalah data yang diperoleh tidak secara langsung dari objek penelitian seperti buku dan jurnal terkait dengan metodeyang digunakan.

BAB IV

HASIL DAN PEMBAHASAN

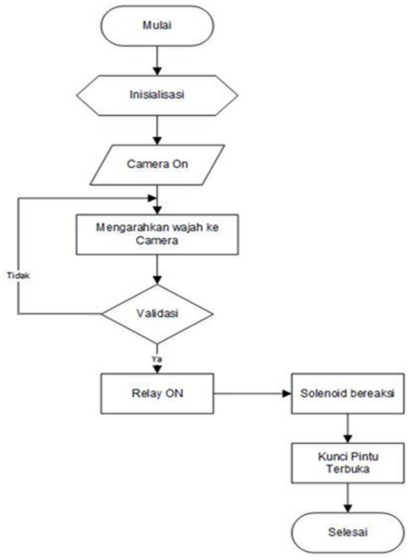
A. Analisis Sistem Yang Sedang Berjalan

Adapun Analisis sistem yang sedang berjalan dapat dilihat pada urutan proses berikut:

1. Mulai.
2. Kunci Pintu Manual
3. Menurunkan *Handle* Pintu
4. Pintu Terbuka
5. Selesei.

Urutan proses diatas menjelaskan gambaran secara umum ketika menggunakan kunci pintu manual untuk membuka pintu.

B. Analisis Sistem Yang Sedang Dibuat



Gambar 4.1 *Flowchart* Sistem Yang diusulkan

Penjelasan dari *flowchart* tersebut adalah pertama melakukan *konfigurasi serial*, apabila jaringan telah terhubung maka alat akan mengaktifkan esp32cam untuk pengambilan sample / mendeteksi wajah, kemudian akan muncul tulisan *ID* [0] sample [1] dan akan berlangsung hingga *ID* [1] sample [5] setelah proses pengambilan sample berhasil maka akan muncul tulisan *Hello Friend* yang menandakan bahwa pendeteksi wajah berhasil mengenali wajah dan setelah itu *solenoid* akan otomatis terbuka.

Pemodelan dalam pengembangan sistem informasi menggunakan *UML*. *UML* merupakan suatu metode pemodelan secara visual untuk sarana perancangan sistem berorientasi objek. Tipe *UML* yang digunakan dalam pengembangan sistem informasi adalah *Use Case*. Diagram dibentuk sebagai berikut:

1. *Use Case*

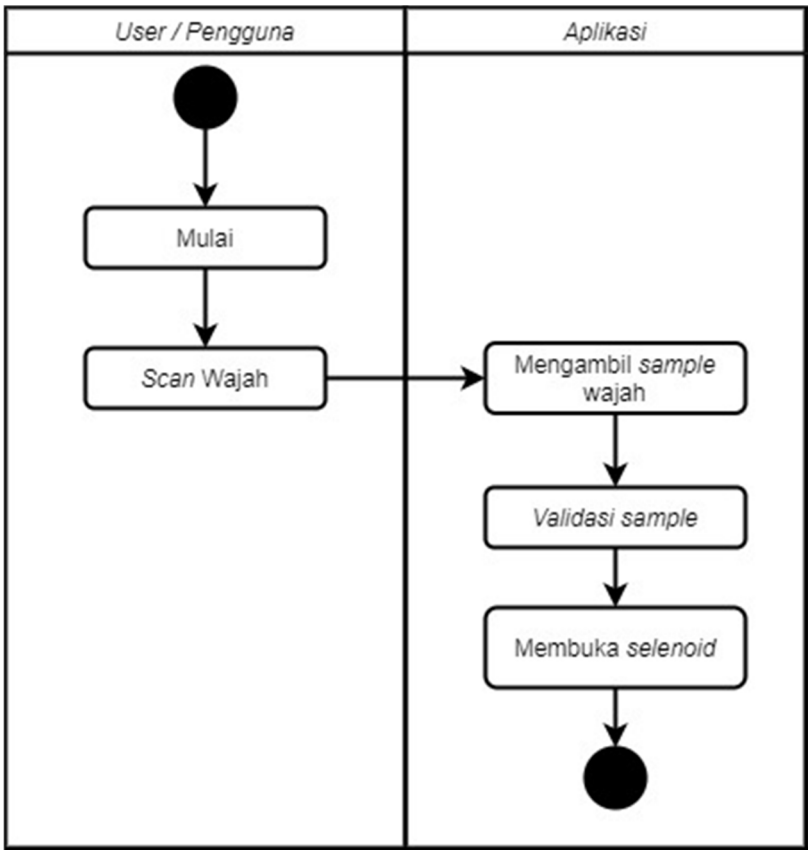
   use case dibentuk sebagai langkah awal perancangan sistem yang menggambarkan interaksi antara sistem dan aktor. *Use Case* juga dapat mendeskripsikan tipe intraksi sipemakai sistem dengan sistemnya. Berikut ini gambaran use case :



Gambar 4.2 *Diagram Use Case*

Penggunadapat membuka *solenoid* secara otomatis jika pengambilan sample wajah dengan menggunakan esp32cam berhasil.

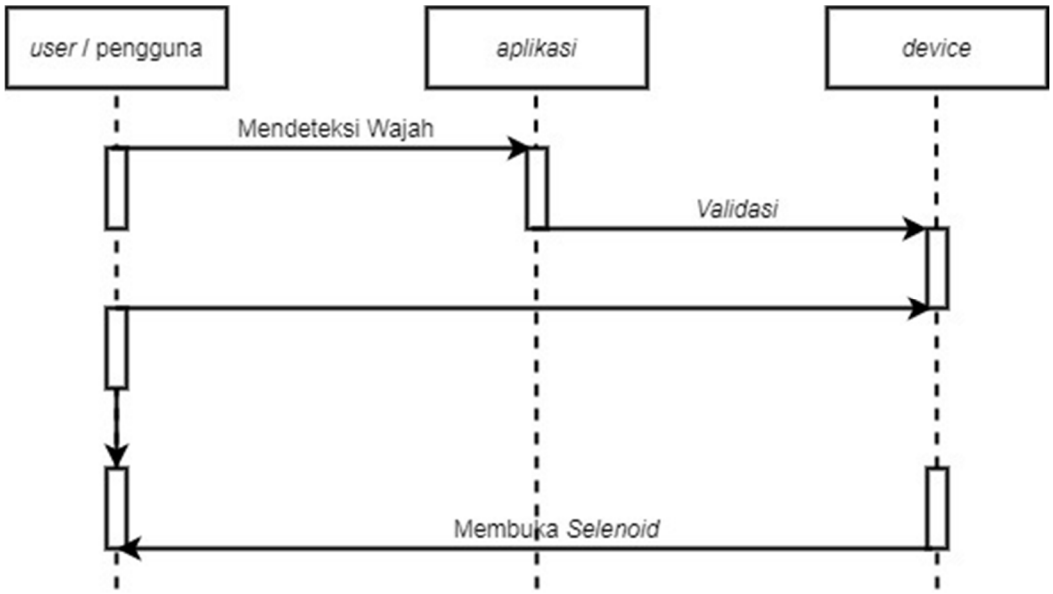
2. *Activity diagram*



Gambar 4. 3 *Activity diagram*

3. *Sequence diagram*

Sequence Diagram merupakan aliran antara objek yang membentuk proses, berikut adalah diagram *sequence*.



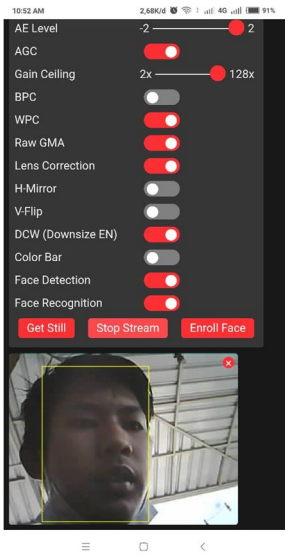
Gambar 4. 4 *Sequence diagram*

C. Rancangan Sistem

Berikut adalah rancangan dari sistem Rancang Bangun *Prototype* Sistem Keamanan Akses Pintu Masuk menggunakan Pedeteksi Wajah Berbasi *IOT*.

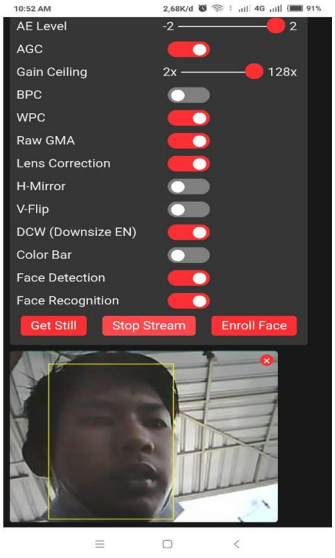
1. Rancangan *Software*

Software sistem keamanan akses pintu masuk menggunakan deteksi wajah dirancang untuk membantu dalam pengaturan pengambilan rekaman wajah *user*. Sistem pengenalan dan pendeteksi wajah akan diaktifkan melalui *Software* ini kemudian status rekaman wajah pun juga akan di tampilkan pada *Software* ini.



Gambar 4. 5 Tampilan Deteksi Wajah

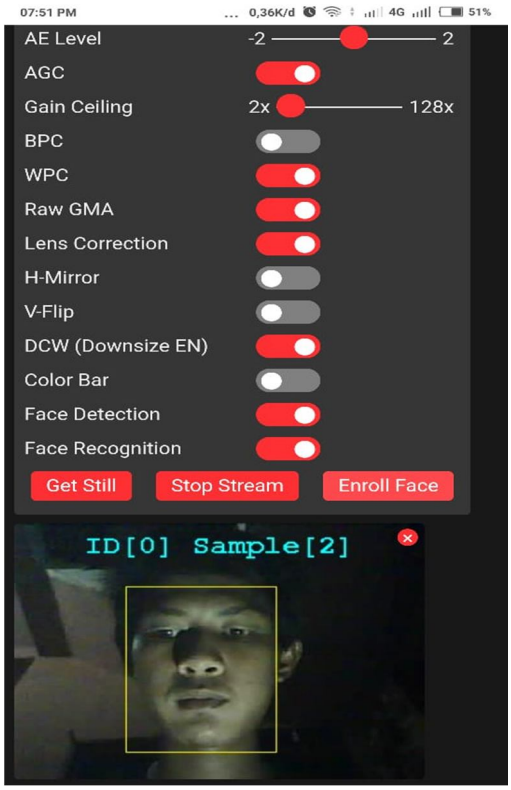
a. Tampilan Awal



Gambar 4. 6 Tampilan Awal

Tampilan Awal ini merupakan tampilan pada saat akan memulai pengambilan *Sample* wajah yang ditandai dengan tulisan .

b. Tampilan Pengambilan *Sample*



Gambar 4. 7 Tampilan pengambilan *Sample*

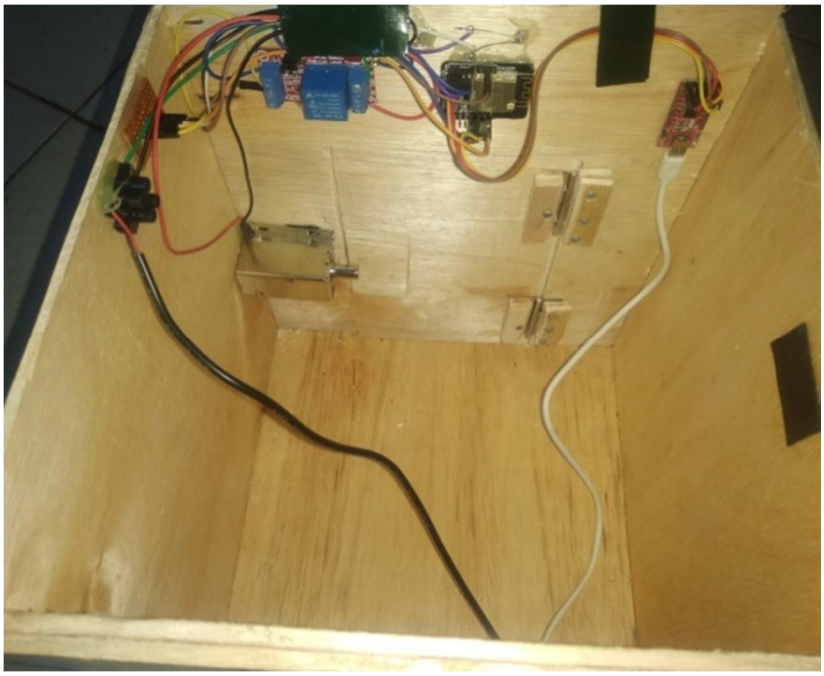
Pada saat *button enroll face* di tekan maka proses pengambilan *Sample* wajah dimulai. Pengambilan *Sample* angka dimulai dengan angka 1 sampai angka 5 dengan warna tulisan berwarna biru setelah itu proses pengambilan *Sample* selesai.

2. Rancangan *Hardware*

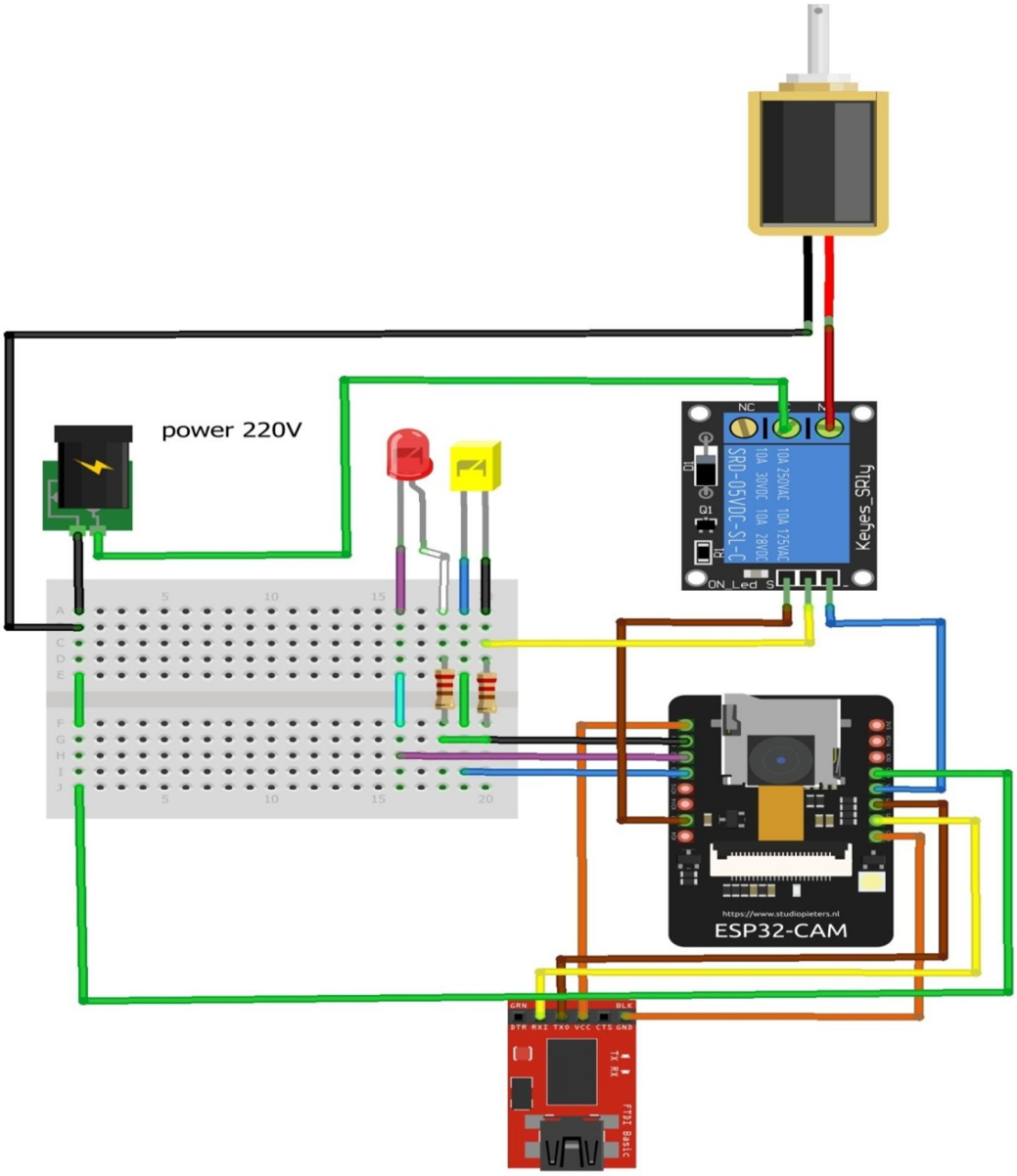
Rancangan *Hardware* dirangkai dalam bentuk Miniatur sederhana seperti yang terlihat pada gambar dibawah.



Gambar 4. 8 Bentuk Rangkaian Diluar *BOX*



Gambar 4. 9 Bentuk Rangkaian Dalam *BOX*



Gambar 4. 10 Rangkaian Alat Perkabelan

Pada perancangan perangkat keras meliputi penjelasan komponen-komponen yang digunakan. Adapun uraian penghubungan pada setiap alat adalah sebagai berikut :

a. Koneksi *Pin* FTDI

Table 4. 1 Koneksi *Pin* FTDI

NO	*PIN*	Keterangan
1	Gnd	Dihubungnkan ke GND Esp32
2	VCC	Dihubungkan ke 5V Esp32
3	TX	Dihubungkan ke *VOR Esp32cam*
4	RX	*VOT Esp32cam*

b. Koneksi Pin Esp32Cam

Tabel 4.1 Koneksi *Pin* Esp32Cam

NO	*PIN*	Keterangan
1	1012	Dihubungkan ke Vcc Lampu 1
2	1013	Dihubungkan ke Vcc Lampu 2
3	102	Dihubungkan ke IN *Relay*
4	Vcc	Dihubungkan ke Vcc *Relay* (D+)
5	GND	Dihubungkan ke (-) *Breadboard*

c. Koneksi *Pin Negatif Board*

Tabel 4.2 Koneksi *Pin Negatif Board*

NO	PIN	Keterangan
1	*Negatif Board*	Dihubungkan ke *Negatif* Lampu 1
2	*Negatif Board*	Dihubungkan ke *Negatif* Lampu 2
3	*Negatif Board*	Dihubungkan ke *Negatif Relay* (-D)

d. Koneksi *Solenoid door lock*

Tabel 4.3 Koneksi *Solenoid door lock*

NO	*PIN*	Keterangan
1	Kabel Merah	Dihubungkan ke *ON Relay*
2	*Negatif Board*	Dihubungkan ke Adaptor

D. Pengujian Sistem

Teknik pengujian sistem merupakan proses pengeksekusian program secara keseluruhan. Pengujian sistem ini berfokus untuk mengetahui proses dan cara kerja aplikasi. Adapun teknik pengujian yang digunakan dalam Rancang Bangun *Prototype* Sistem Keamanan Akses Pintu Masuk Menggunakan Deteksi

Wajah Berbasis *IOT* adalah Pengujian Alat secara langsung, Pengujian *Black Box* dan Pengujian *White Box*.

1. Pengujian Alat Secara langsung

   a. Pengujian dalam ruangan terang

      1. Pengujian Pengambilan *Sample* Wajah

         Pengujian ini dilakukan dalam sebuah ruangan dengan menggunakan pencahayaan lampu ruangan serta jarak objek dengan kamera minimal kurang lebih 30 cm dan jarak maksimal kurang lebih 70cm. Pengujian dimulai dengan pengambilan *sample* setelah menekan tombol *enroll face* kemudian akan muncul tulisan *ID* [0] sample [1] dan akan berlangsung hingga *ID* [1] sample [5] setelah proses pengambilan sample berhasil maka akan muncul tulisan *Hello Friend*. seperti pada gambar dibawah.



Gambar 4. 11 Pengambilan *sample*

      2. Pengujian Membuka *Solenoid* Menggunakan Deteksi Wajah

         untuk membuka pintu menggunakan pendeteksi wajah dibutuhkan ruangan dengan pencahayaan yang cukup agar sistem dapat mengenal wajah dengan baik. Saat wajah di dihadapkan ke *ESP32cam* maka akan muncul tulisan *Hello Friend* yang menandakan bahwa pendeteksi wajah berhasil mengenali wajah dan setelah itu *solenoid* akan otomatis terbuka.



Gambar 4. 12 Deteksi wajah untuk membuka pintu

b. Pengujian dalamruangan kurang cahaya

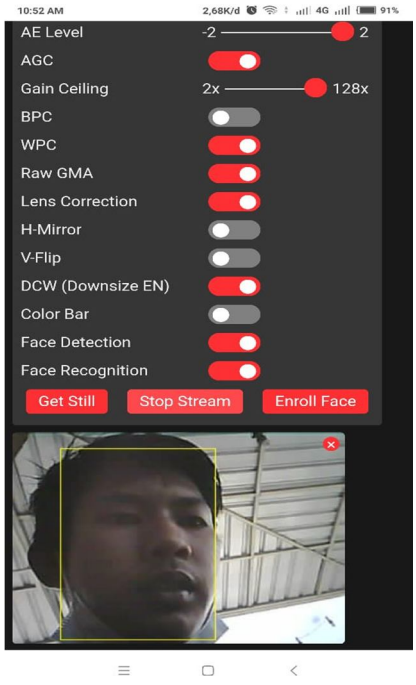


Gambar 4. 13 Pengujian alat dalam ruangan kurang cahaya

Sistem keamanan pintu masuk yang di uji pada ruangan kurang cahaya tidak dapat mendeteksi wajah. Pengambilan sample maupun deteksi untuk membuka pintu tidak berhasil.

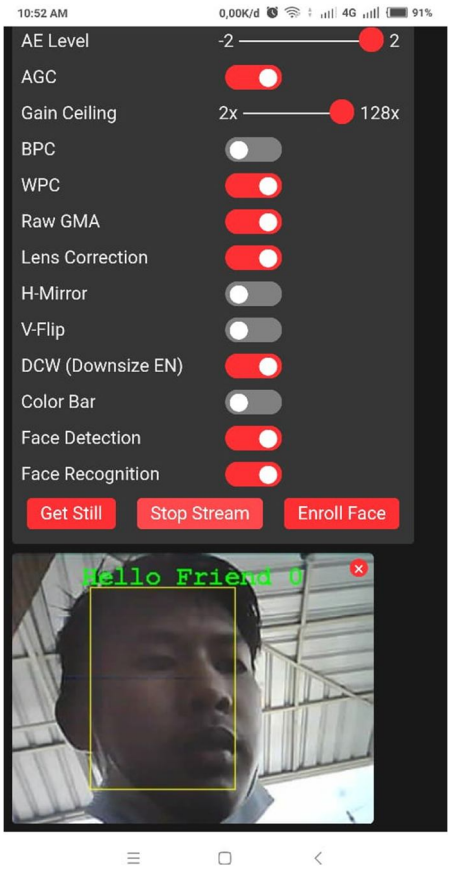
2. Pengujian *Black Box*

Tabel 4.4 Terkoneksi *Software dan Hardware*

Test Factor	**Hasil**	**Keterangan**
Proses Inisialisasi Apabila sukses	✓	Berhasil, karena Aplikasi telah terinisialisasi dan semua Modul telah terkoneksi.
		

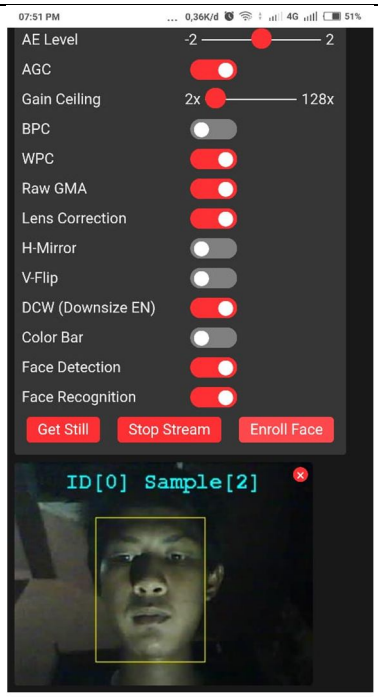
Tabel 4.5 Pendeteksi Wajah Berhasil

Test Factor	**Hasil**	**Keterangan**
Proses Deteksi Wajah Berhasil	✓	Berhasil, kamera dapat mendeteksi wajah ditandai dengan munculnya kotak warna kuning diarea wajah dan muncul pesan *Hello Friend.*

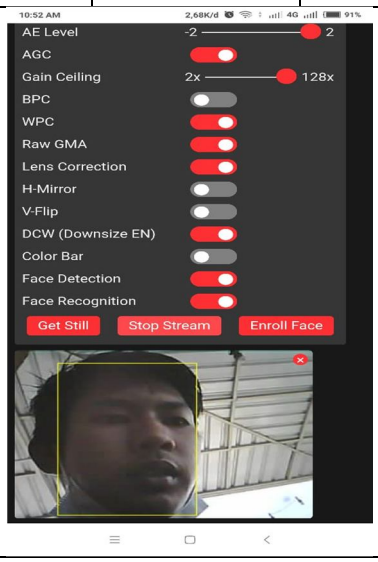


Tabel 4.6 pengambilan *Sample*

Test Factor	**Hasil**	**Keterangan**
Proses pengambilan *Sample* wajah	✓	Berhasil, pengambilan *Sample* berhasil dengan jumlah pengambilan *Sample* sebanyak 5 *Sample*



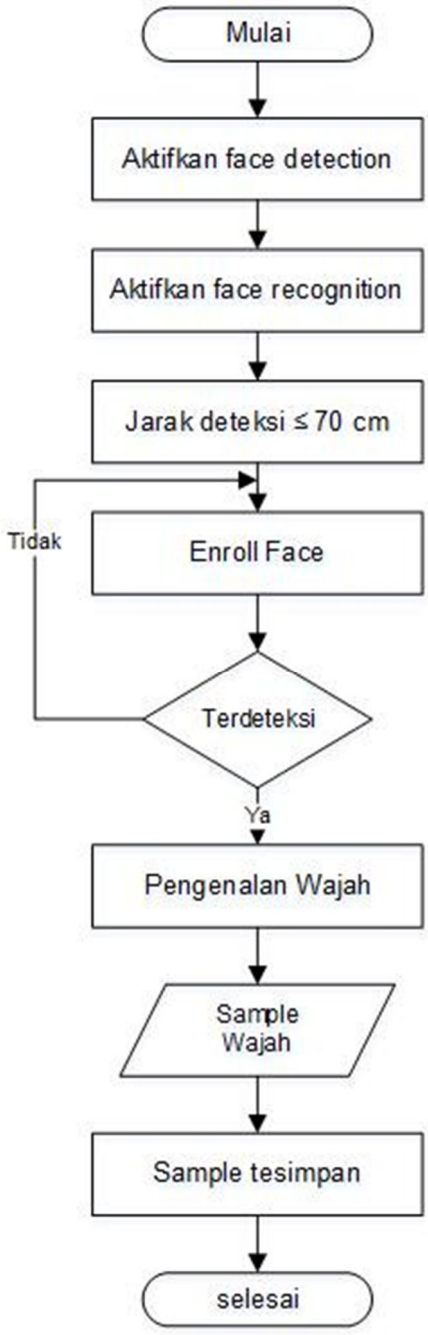
Tabel 4.7 pengenalan wajah

Test Factor	**Hasil**	**Keterangan**
Proses pengenal wajah setelah pengambilan *Sample*	✓	Berhasil, pengenalan wajah hanya ditandai dengan kotak kuning dan tanpa pesan kemudian *Solenoid* akan otomatis terbuka.
		

3.Pengujian *White Box*

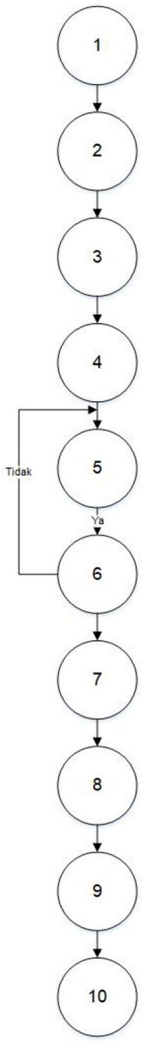
a. *White Box Enroll Face*

1. *Flowchart Enroll Face*



Gambar 4. 14 *Flowchart Enroll* Face

2. *Flowgraph Enroll Face*



Gambar 4. 15 *Flowgraph* Enroll Face

Dari Gambar *FlowGraph* di atas dapat di lakukan proses perhitungan sebagai berikut :

a) Menghitung C*yclomatic Complexcity* V(G) dari *Egde* dan *Node*:

Dengan Rumus : V (G) = E – N + 2

N(*node*) = 10

E(*edge*) = 10

P (*predikat node*) = 1

Penyelesaian : V(G) = E – N + 2

= 10 - 10 + 2

= 2

Predikat *Node* (N) = P + 1

= 1 + 1

= 2

b) Berdasarkan perhitungan C*yclomatic Complexcity* dari *Flowgraph* diatas memiliki *Region* = 2

c) *Independent Path* pada *flowgraph* diatas adalah :

Path 1 = 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10

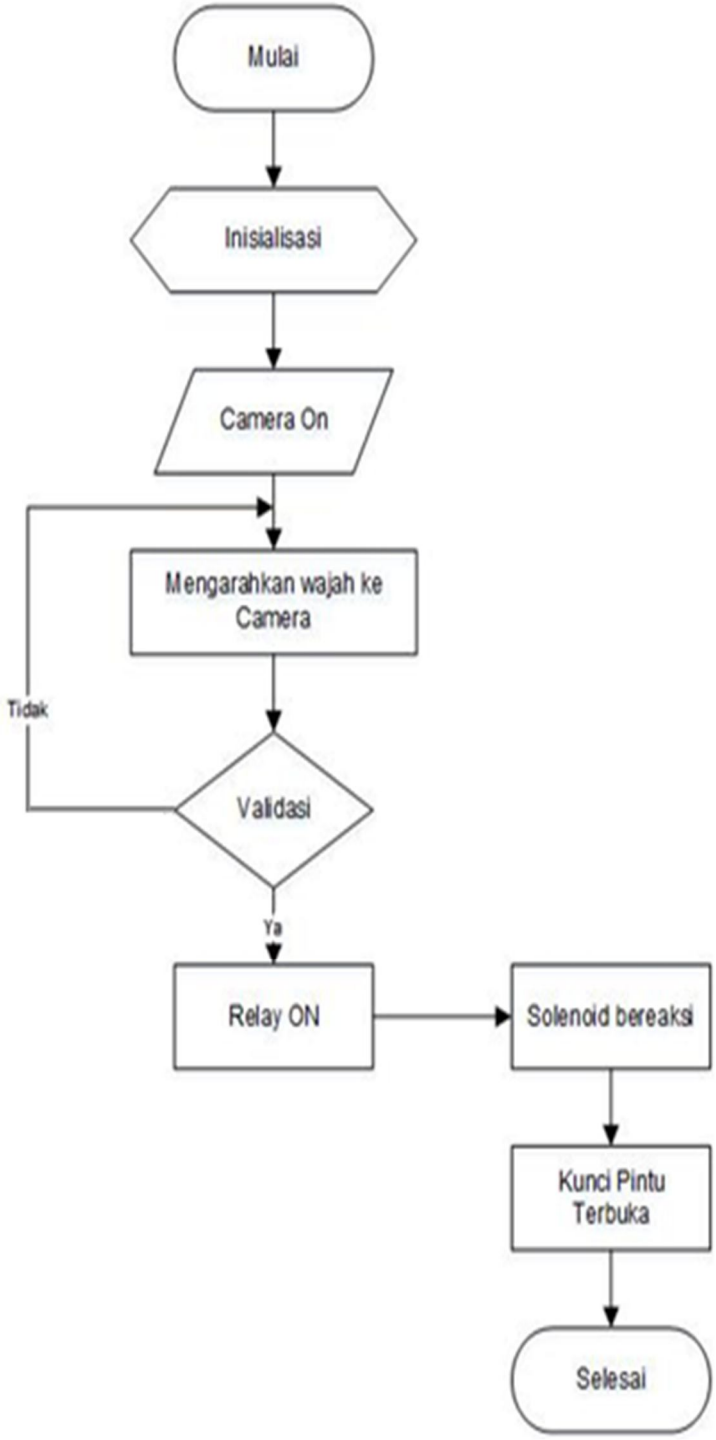
Path 2 = 1 – 2 – 3 – 4 – 5 – 6 – 5 – 6 – 7 – 8 – 9 – 10

d) Grafik *Matriks*

	1	2	3	4	5	6	7	8	9	10	E - 1
1		1									1 – 1 = 0
2			1								1 – 1 = 0
3				1							1 – 1 = 0
4					1						1 – 1 = 0
5					1	1					1 – 1 = 0
6							1				2 – 1 = 1
7								1			1 – 1 = 0
8									1		1 – 1 = 0
9										1	1 – 1 = 0
10											0
Zum (E + 1)											1 + 1 = 2

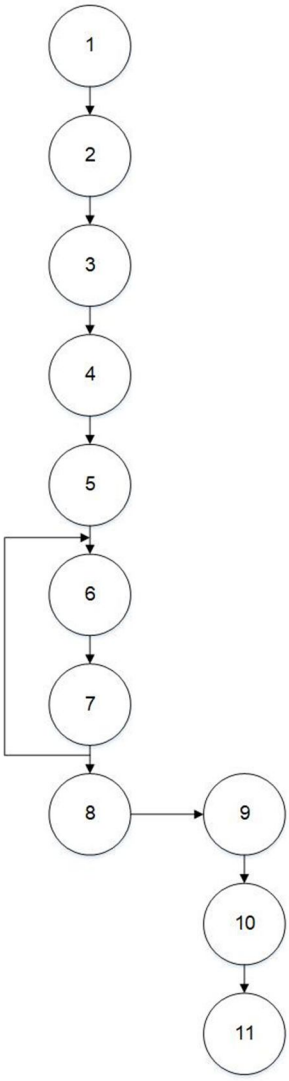
b. *White Box* pengujian alat

1. *Flowchart*pengujian alat



Gambar 4. 16 *Flowchart*pengujian alat

2. *Flowgraph* Pengujian alat



Gambar 4. 17 *Flowgraph* pengujian alat

Dari Gambar *FlowGraph* di atas dapat di lakukan proses perhitungan sebagai berikut :

a) Menghitung C*yclomatic Complexcity* V(G) dari *Egde* dan *Node*:

Dengan Rumus : V (G) = E – N + 2

N(*node*) = 11

E(*edge*) = 11

P (*predikat node*) = 1

Penyelesaian : V(G) = E – N + 2

= 11 - 11 + 2

= 2

Predikat *Node* (N) = P + 1

= 1 + 1

= 2

b) Berdasarkan perhitungan C*yclomatic Complexcity* dari *Flowgraph* diatas memiliki *Region* = 2

c) *Independent Path* pada *flowgraph* diatas adalah :

Path 1 = 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11

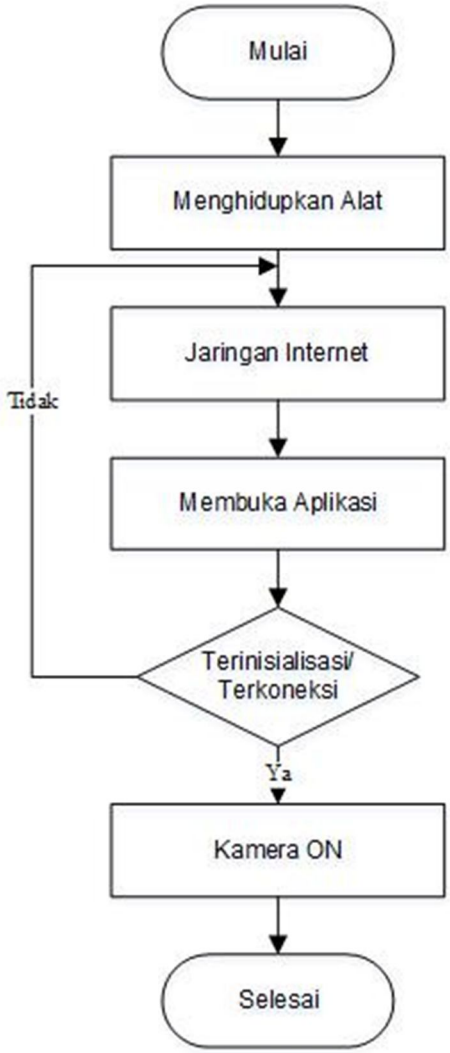
Path 2 = 1 – 2 – 3 – 4 – 5 – 6 – 7 – 6 – 7 – 8 – 9 – 10 – 11

d) Grafik *Matriks*

	1	2	3	4	5	6	7	8	9	10	11	E - 1
1		1										1 – 1 = 0
2			1									1 – 1 = 0
3				1								1 – 1 = 0
4					1							1 – 1 = 0
5						1						1 – 1 = 0
6						1	1					2 – 1 = 1
7								1				1 – 1 = 0
8									1			1 – 1 = 0
9										1		1 – 1 = 0
10											1	1 – 1 = 0
11												0
Zum (E + 1)												1 + 1 = 2

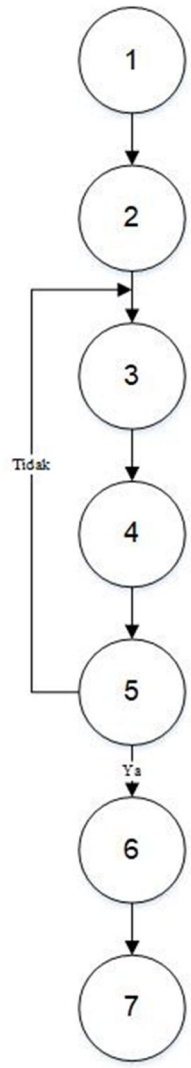
c. *White Box* Koneksi Hardware dan Software

1. *Flowchart* Koneksi Hardware dan Software



Gambar 4. 18 *Flowchart*Koneksi *Hardware* dan *Software*

2. *Flowgraph* Koneksi *Hardware* dan *Software*



Gambar 4. 19 *Flowgraph* Koneksi *Hardware* dan *Software*

Dari Gambar *FlowGraph* di atas dapat di lakukan proses perhitungan sebagai berikut :

a) Menghitung C*yclomatic Complexcity* V(G) dari *Egde* dan *Node*:

Dengan Rumus : V (G) = E – N + 2

N(*node*) = 7

E(*edge*) = 7

P (*predikat node*) = 1

Penyelesaian : V(G) = E – N + 2

= 7 - 7 + 2

= 2

Predikat *Node* (N) = P + 1

= 1 + 1

= 2

b) Berdasarkan perhitungan C*yclomatic Complexcity* dari *Flowgraph* diatas memiliki *Region* = 2

c) *Independent Path* pada *flowgraph* diatas adalah :

Path 1 = 1 – 2 – 3 – 4 – 5 – 6 – 7

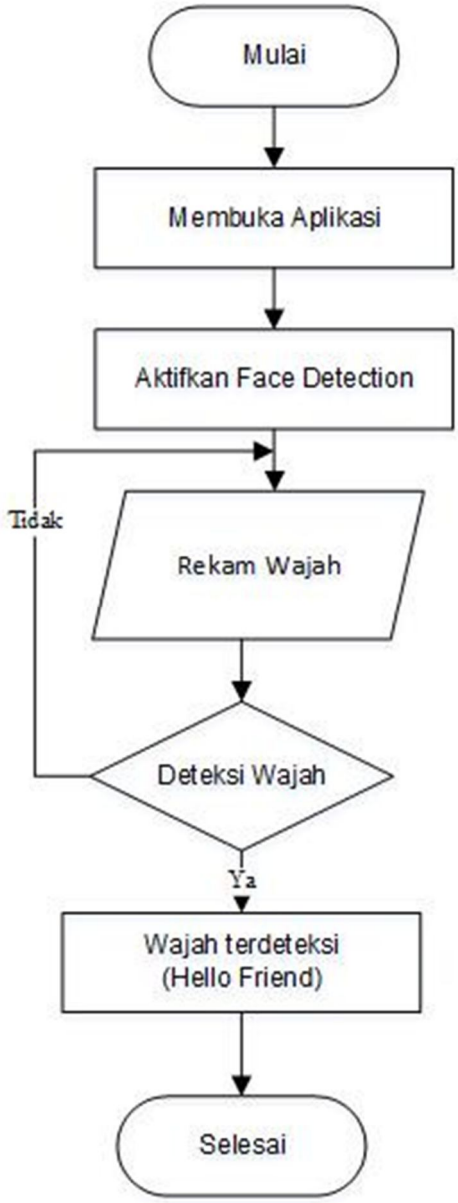
Path 2 = 1 – 2 – 3 – 4 – 5 – 3 – 4 – 5 – 6 – 7

d) Grafik *Matriks*

	1	2	3	4	5	6	7	E - 1
1		1						1 – 1 = 0
2			1					1 – 1 = 0
3				1				1 – 1 = 0
4					1			1 – 1 = 0
5			1			1		2 – 1 = 1
6							1	1 – 1 = 0
7								0
Zum (E + 1)								1 + 1 = 2

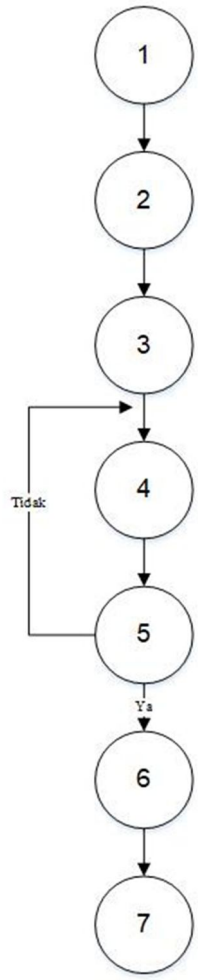
d. *White Box* Deteksi Wajah

1. *Flowchart*Deteksi Wajah



Gambar 4. 20 *Flowchart* Deteksi Wajah

2. *Flowgraph* Deteksi Wajah



Gambar 4. 21 *Flowgraph* Deteksi Wajah

Dari Gambar *FlowGraph* di atas dapat di lakukan proses perhitungan sebagai berikut :

a) Menghitung C*yclomatic Complexcity* V(G) dari *Egde* dan *Node*:

Dengan Rumus : V (G) = E – N + 2

N(*node*) = 7

E(*edge*) = 7

P (*predikat node*) = 1

Penyelesaian : V(G) = E – N + 2

= 7 - 7 + 2

= 2

Predikat *Node* (N) = P + 1

= 1 + 1

= 2

b) Berdasarkan perhitungan C*yclomatic Complexcity* dari *Flowgraph* diatas memiliki *Region* = 2

c) *Independent Path* pada *flowgraph* diatas adalah :

Path 1 = 1 – 2 – 3 – 4 – 5 – 6 – 7

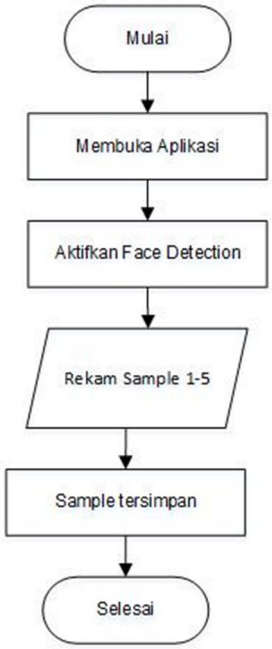
Path 2 = 1 – 2 – 3 – 4 – 5 – 4 – 5 – 6 – 7

d) Grafik Matriks

	1	2	3	4	5	6	7	E - 1
1		1						1 – 1 = 0
2			1					1 – 1 = 0
3				1				1 – 1 = 0
4					1			1 – 1 = 0
5				1		1		2 – 1 = 1
6							1	1 – 1 = 0
7								0
Zum (E + 1)								1 + 1 = 2

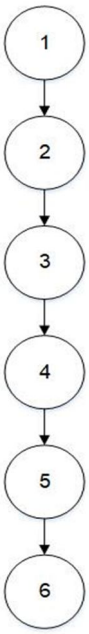
e. *White Box Sample* Wajah

1. *Flowchart Sample* Wajah



Gambar 4. 22 *FlowchartSample* Wajah

2. *Flowgraph Sample* Wajah



Gambar 4. 23 *Flowgraph Sample* Wajah

Dari Gambar *FlowGraph* di atas dapat di lakukan proses perhitungan sebagai berikut :

a) Menghitung C*yclomatic Complexcity* V(G) dari *Egde* dan *Node*:

Dengan Rumus : V (G) = E – N + 2

N(*node*) = 6

E(*edge*) = 5

P (*predikat node*) = 0

Penyelesaian : V(G) = E – N + 2

= 5 - 6 + 2

= 1

Predikat *Node* (N) = P + 1

= 0+ 1

= 1

b) Berdasarkan perhitungan C*yclomatic Complexcity* dari *Flowgraph* diatas memiliki *Region* = 1

c) *Independent Path* pada *flowgraph* diatas adalah :

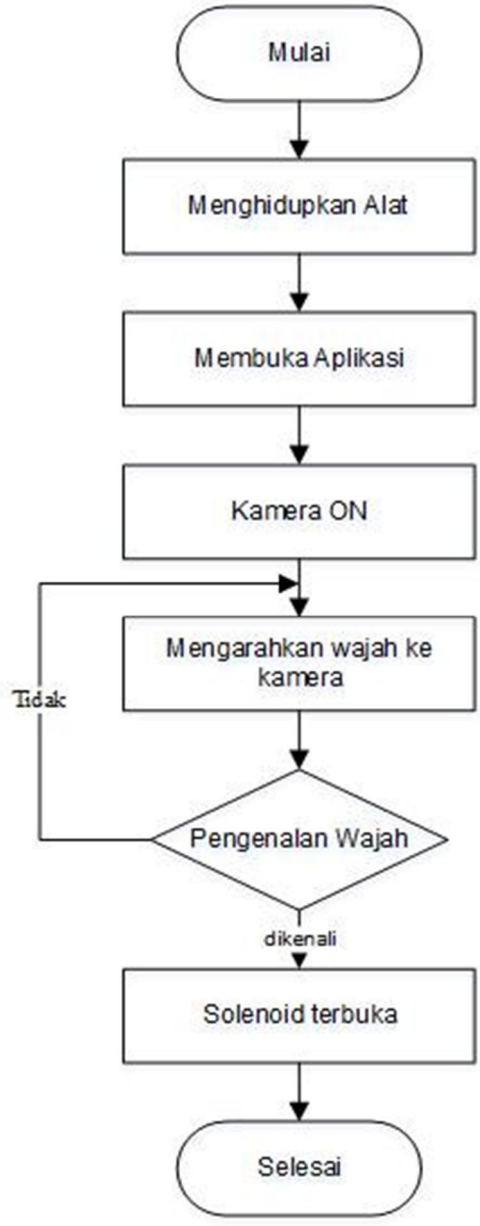
Path 1 = 1 – 2 – 3 – 4 – 5 – 6

d) Grafik *Matriks*

	1	2	3	4	5	6	E - 1
1		1					1 – 1 = 0
2			1				1 – 1 = 0
3				1			1 – 1 = 0
4					1		1 – 1 = 0
5						1	1 – 1 = 0
6							1 – 1 = 0
							0
Zum (E + 1)							0 + 1 = 1

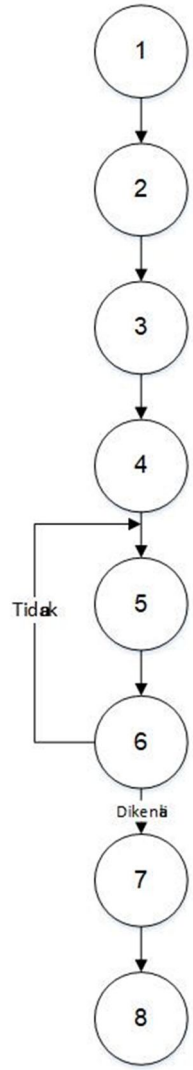
f. *White Box* Pengenalan Wajah

1. *Flowchart*Pengenalan Wajah



Gambar 4. 24 *Flowchart* Pengenalan Wajah

2. *Flowgraph* Pengenalan Wajah



Gambar 4. 25 *Flowgraph* Pengenalan Wajah

Dari Gambar *FlowGraph* di atas dapat di lakukan proses perhitungan sebagai berikut :

a) Menghitung C*yclomatic Complexcity* V(G) dari *Egde* dan *Node*:

Dengan Rumus : V (G) = E – N + 2

N(*node*) = 8

E(*edge*) = 8

P (*predikat node*) = 1

Penyelesaian : V(G) = E – N + 2

= 8 - 8 + 2

= 2

Predikat *Node* (N) = P + 1

= 1 + 1

= 2

b) Berdasarkan perhitungan C*yclomatic Complexcity* dari *Flowgraph* diatas memiliki *Region* = 2

c) *Independent Path* pada *flowgraph* diatas adalah :

Path 1 = 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8

Path 2 = 1 – 2 – 3 – 4 – 5 – 6 – 5 – 6 – 7 – 8

d) Grafik *Matriks*

	1	2	3	4	5	6	7	8	E – 1
1		1							1 – 1 = 0
2			1						1 – 1 = 0
3				1					1 – 1 = 0
4					1				1 – 1 = 0
5						1			1 – 1 = 0
6					1		1		2 – 1 = 1
7								1	1 – 1 = 0
8									0
Zum (E + 1)									1 + 1 = 2

BAB V

PENUTUP

A. Kesimpulan

Berdasarkan hasil penelitian dan pembahasan yang telah dijelaskan pada bab- bab sebelumnya, maka dapat ditarik kesimpulan sebagai berikut:

1. Setelah alat ini jadi maka dapat di ketahui bahwa sistem ini bisadigunakan untuk keamanan akses pintu masuk yang menggunakan wajah sebagai kunci.
2. Sistem ini menggunakan deteksi wajah sebagai sistem keamanan agar dapat maengatasi masalah yang sering terjadi seperti lupa letak menyimpan kunci, kunci tercecer dan penyalahgunaan kunci oleh orang yang tidak bertanggung jawab
3. Untuk membuka pintu menggunakan pendeteksi wajah dibutuhkan ruangan dengan pencahayaan yang cukup agar sistem dapat mengenal wajah dengan baik.
4. Pengujian yang dilakukan dalam sebuah ruangan dengan menggunakan pencahayaan lampu ruangan serta jarak objek dengan kamera minimal kurang lebih 30 cm dan jarak maksimal kurang lebih 70cm berhasil mengaktifkan *solenoid.*
5. Sistem keamanan pintu masuk yang di uji pada ruangan kurang cahaya tidak dapat mendeteksi wajah sehingga pengambilan sample maupun deteksi untuk membuka pintu tidak berhasil.

6. Pengujian yang dilakukan dengan pengambilan sample wajah pengguna yang menggunakan masker maupun kacamatadapat terdeteksi sehingga dapat membuka pintu.
7. Sistem keamanan akses pintu masuk hasil penelitian ini, belum memakai *memory card* sebagai media penyimpanan sehingga pada saat melakukan *enrollface* maka foto wajah yang lama akan terhapus

B. Saran

Diharapkan dalam pengembangan selanjutnya rancang bangun *Prototype* sistem keamanan akses pintu masuk menggunakan pendeteksi wajah berbasis *IOT* ini dapat dikembangkan dengan penambahan fitur notifikasi berupa *capture* wajah orang yang sedang berdiri depan pintu menghadap ke kamera serta penambahan kualitas kamera berserta *flash* untuk ruangan yang kurang cahaya.

DAFTAR PUSTAKA

Alam, H., Parinduri, I., Hutagalung, S. N., Hutagalung, J. E., & Masri, M. (2020). *PEMBELAJARAN & PRAKTIKUM DASAR: Mikrokontroler AT8535, Arduino UNO R-3 BASCOM AVR, Arduino UNO 1.16 dan Fritzing.* Medan: Yayasan Kita Menulis.

Ariandi, A., Yesputra, R., & Risnawati. (2021). *Perancangan Smart Home dengan Sistem Kendali dari Android di CV Rifanta Tanjung Balai. JUTSI (Jurnal Teknologi dan Sistem Informasi)* , 51-60.

Fauzan, M. N., & Adiputri, L. C. (2019). *Tutorial Membuat Prototipe Prediksi Ketinggian Air (PKA) Untuk Pendeteksi Banjir Peringatan Dini Berbasis IOT.* Bandung: Kreatif Industri Nusantara.

Hakiem, I. (2015). TOKOTEKNOLOGI*: Electronic desain & repair.* Malang: PT. Tokoteknologi Mikroelektronik Nusantara.

Kurniawan, T. A. (2020). *Sistem Informasi Akuntansi dengan Pendekatan Simulasi.* Yogyakarta: Deepublish.

Malliwang, Y. (2020). *Tempat Sampah Otomatis Berbasis Mikrokontroller Arduino Uno.* Makassar.

Mifta, M., & Arifin. (2016). *Pengamanan Laptop Menggunakan Pengenalan Wajah Berbasis Triangel Face.* Semarang.

Muhammad, F. (2018). Sistem *Keamanan Akses Pintu Masuk Menggunakan Face Recognition Berbasis Raspaberry PI 3.* Makassar.

Nusyirwan, D., Lingga, S., & Perdana, P. P. (2020). *Rancang Bangun Smart Book Calculator Berbasis Mikrokontroller Arduino Uno Sebagai Inovasi Cerdas Untuk Memudahkan Siswa Menghitung Guna Menuju Revolusi Industri 4.0. Jurnal Teknologi dan Terapan Bisnis (JTTB)* , 13-22.

Piarsa, N., & Wibawa, K. S. (2017). *Prototipe Deteksi dan Pengenalan Wajah pada Sistem Monitoring dan Kontrol Visual Keamanan Rumah.* Badung.

Razor, A. (2020, April 26). *Kabel Jumper Arduino : Pengertian, Fungsi, Jenis, dan Harga*. Dipetik Februari 9, 2021, dari ALDYRAZOR.COM: https://www.aldyrazor.com/2020/04/kabel-*Jumper*-arduino.html