

## Tutorial Sistem Keamanan Pintu dengan Sidik Jari

### Menggunakan Arduino UNO R3

Hallo IoT... Pada kesempatan kali ini kita akan belajar tentang Sistem keamanan Pintu menggunakan Sidik Jari Berbasis Arduino UNO R3. Pada sistem keamanan ini menggunakan solenoid door lock sebagai aktuator dan mengendalikannya melalui Sidik Jari atau Fingerprint. Disini kita menggunakan sensor Fingerprint DY50, sebelumnya kita harus mengenal apa itu sensor fingerprint. Fingerprint adalah sebuah alat elektronik yang menerapkan sensor scanning untuk mengetahui sidik jari seseorang guna keperluan verifikasi identitas. Secara sederhana cara kerja dari sebuah sensor Fingerprint adalah dengan merekam data sidik jari untuk pertama kalinya untuk digunakan sebagai acuan. Data sidik jari tersebut akan di simpan dalam database. Ketika ada yang ingin mengakses sebuah alat yang di pasang sensor Fingerprint, maka akan di lakukan scanning ulang, kemudian data sidik jari dari hasil scanning ulang tersebut akan di cocokan apakah sama seperti data sidik jari yang sudah pernah di simpan dalam database atau tidak. Kemudian data tersebut diolah dengan mikrokontroller, kemudian mikrokontroller akan memberikan perintah kepada solenoid untuk membuka kunci pintu.



Gb. Sensor Fingerprint

(Sumber : <https://www.ebay.com/>)

### Spesifikasi Fingerprint :

Tegangan	3.6V - 6V
Arus	120mA
Waktu capturing	1 detik
Dimensi area pembacaan	18 x 14 mm
Match mode	1:1
Search mode	1:N
Signature file	256 bytes
Templates file	512 bytes
Kapasitas penyimpanan	1000
Interface PC	UART (TTL logic) usb 2.0/usb 1.1
Dimensi	56 x 20 x 21.5 mm
Serial komunikasi	3.3V /5V

Baiklah IOTA itu dulu sedikit pengertian tentang komponen yang akan kita gunakan pada tutorial kali ini. Selanjutnya mari kita langsung praktikkan dengan mengikuti tahap demi tahap dibawah ini

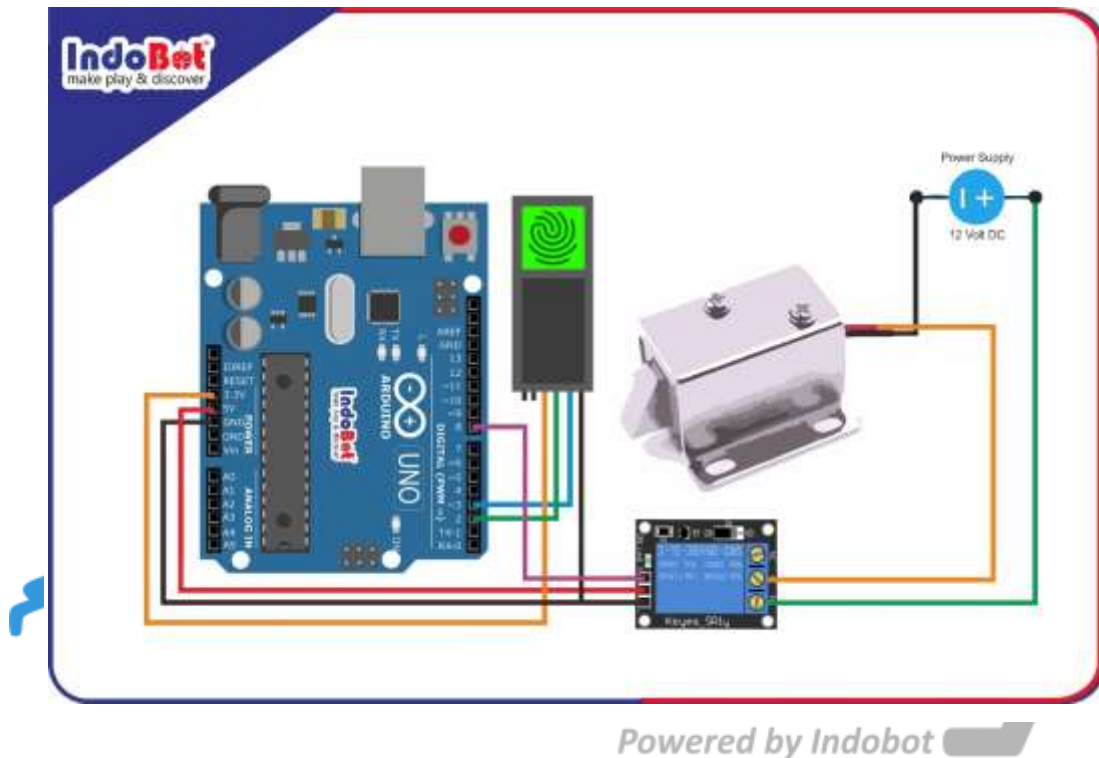
#### 1. Alat dan Bahan

Pertama IOTA harus menyiapkan alat beserta bahan yang dibutuhkan seperti dibawah ini :

- |  |            |
|--|------------|
| 1) Komputer yang terinstal Arduino IDE | 1 unit     |
| 2) Arduino Uno R3 dan USB              | 1 buah     |
| 3) Fingerprint DY50                    | 1 buah     |
| 4) Solenoid Door Lock                  | 1 buah     |
| 5) Relay 1 channel                     | 1 buah     |
| 6) Adaptor 12VDC                       | 1 buah     |
| 7) Project Board                       | 1 buah     |
| 8) Kabel Penghubung                    | Secukupnya |

## 2. Gambar Rangkaian

Selanjutnya buatlah rangkaian yang sama persis dengan gambar dibawah ini :



Keterangan :

- 1) Kaki VCC Relay pada pin 5V Arduino
- 2) Kaki Ground Relay pada pin GND Arduino
- 3) Kaki Input Relay pada pin 8 Arduino
- 4) Kaki COM Relay pada pin VCC Solenoid Door Lock
- 5) Kaki NO Relay pada sumber 12V Adaptor
- 6) Kaki GND Solenoid Door Lock pada GND Adaptor
- 7) Kaki 3V3 Fingerprint pada pin 3V3 Arduino
- 8) Kaki TX Fingerprint pada pin 2 Arduino
- 9) Kaki RX Fingerprint pada pin 3 Arduino
- 10) Kaki GND Fingerprint pada pin GND Arduino

### 3. Programming

Setelah kalian pastikan semua rangkaian sudah oke, saatnya masuk kepada tahap koding nih IoT. Tapi saya ingatkan ya IoT karena ini menggunakan Fingerprint pastikan Arduino IDE pada komputer kalian telah terinstall Librarynya. Jika belum silahkan download dan install terlebih dahulu, jika belum tahu caranya silahkan bisa cari tutorialnya terlebih dahulu. Nah, Kalau Arduino IDE kalian sudah terinstall library Adafruit Fingerprint Sensor, terdapat tiga bagian program, yaitu :

1. Yang pertama program daftar yaitu program untuk mendaftarkan jari kalian ke Fingerprint Sensor

```
#include <Adafruit_Fingerprint.h> //library Adafruit Fingerprint
#include <SoftwareSerial.h> //library komunikasi serial untuk fingerprint
SoftwareSerial mySerial(2, 3); //mengatur pin RX pada pin 2 dan TX pada pin 3
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial); //pemanggilan fungsi Adafruit_Fingerprint
uint8_t id; //membuat variabel id dengan tipe data uint8_t yang tidak bisa bernilai negatif dan beresolusi 8 bit
void setup(){
  Serial.begin(9600); //Membuka komunikasi serial dengan mengatur baudrate 9600
  finger.begin(57600); // atur kecepatan data untuk port serial sensor
  if (finger.verifyPassword()) { //pernyataan bersyarat, jika memenuhi syarat maka akan menjalankan program dibawah. jika tidak langsung masuk ke program else
    Serial.println("FingerPrint Sensor Ditemukan!"); //Menampilkan tulisan "FingerPrint Sensor Ditemukan!" di serial monitor
  } else { //jika tidak memenuhi pernyataan pada program if maka masuk ke program else
    Serial.println("FingerPrint Sensor Tidak Ditemukan! :("); //Menampilkan tulisan "FingerPrint Sensor Tidak Ditemukan!" di serial monitor
    while (1) { delay(1); } //looping delay 1ms
  }
}
```

```

}

void loop(){
    ENROLL(); //Memanggil fungsi Enroll
}

//-----ENROLL-----//

void ENROLL(){
    Serial.println("Ready to enroll a fingerprint!"); //menampilkan tulisan "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(); //variabel id yang sudah dideklarasikan berisi program
    readnumber

    if (id == 0) { // ID # 0 tidak diizinkan, coba lagi!

        return; //keluar dari looping
    }

    Serial.print("Enrolling ID #"); //menampilkan tulisan "Enrolling ID #"
    Serial.println(id); //menampilkan hasil variabel id pada serial monitor

    while (! getFingerprintEnroll() );
}

//-----PROSES ENROLL-----//

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

```

```
//-----RETURN NUM-----//

uint8_t getFingerprintEnroll() {
    int p = -1; //mendeklarasikan p = -1 dengan tipe data integer
    Serial.print("Waiting for valid finger to enroll as #"); Serial.println(id);

    //Program untuk input data sidik jari, cek sidik jari
    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;
        }
    }

    //Program untuk convert gambar menjadi data
    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;
}

//Remove data sidik jari
Serial.println("Remove finger");
delay(2000);
p = 0;
while (p != FINGERPRINT_NOFINGER) {
    p = finger.getImage();
}

//program untuk masukkan sidik jari lagi
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;
  }
}

//Program untuk convert gambar menjadi data
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;
}
// Membuat data untuk sidik jari
Serial.print("Creating model for #"); Serial.println(id);
//Mencocokkan data sidik jari
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;
}
//Data tersimpan
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;
}
}
}
//-----ENROLL END-----//

```

2. Setelah selesai melakukan pendaftaran, selanjutnya kamu bisa coba program Fingerprint, yaitu program untuk cek jari kalian terdaftar atau tidak.

```

#include <Adafruit_Fingerprint.h> //library Adafruit Fingerprint
#include <SoftwareSerial.h> //library komunikasi serial untuk fingerprint
SoftwareSerial mySerial(2, 3); //mengatur pin RX pada pin 2 dan TX pada pin 3
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial); //pemanggilan
fungsi Adafruit_Fingerprint

```

```

#define kunci 8 //mendefinisikan pin 8 sebagai input relay dengan aktuator berupa
solenoid door lock

void setup(){
  Serial.begin(9600); //Membuka komunikasi serial dengan mengatur baudrate
9600

  finger.begin(57600); // mengatur kecepatan data untuk port serial sensor

  pinMode(kunci, OUTPUT); //Mengatur pin kunci sebagai output

  if (finger.verifyPassword()) { //pernyataan bersyarat, jika memenuhi syarat maka
akan menjalankan program dibawah. jika tidak langsung masuk ke program else

    Serial.println("FingerPrint Sensor Ditemukan!"); //Menampilkan tulisan
"FingerPrint Sensor Ditemukan!" di serial monitor

  } else { //jika tidak memenuhi pernyataan pada program if maka masuk ke
program else

    Serial.println("FingerPrint Sensor Tidak Ditemukan! :("); //Menampilkan
tulisan "FingerPrint Sensor Tidak Ditemukan!" di serial monitor

    while (1) { delay(1); } //looping delay 1ms
  }
}

void loop(){
  FINGERPRINT(); //Memanggil fungsi FINGERPRINT
}

//-----FINGERPRINT-----//

void FINGERPRINT(){
  getFingerprintIDez(); //Berfungsi untuk mendapatkan sidik jari
  delay(50); //jeda waktu 50ms
}

//-----PROSES FINGERPRINT-----//

//program untuk input data sidik jari, cek sidik jari
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;
}
// Program untuk convert gambar menjadi data
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_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;
}

//Program untuk mencocokkan data sidik jari
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;
}

//Menemukan kecocokan
Serial.print("Found ID #"); Serial.print(finger.fingerID);
Serial.print(" with confidence of "); Serial.println(finger.confidence);
return finger.fingerID;
}

// mengembalikan -1 jika gagal, jika tidak mengembalikan ID #
int getFingerprintIDez() {

```

```

uint8_t p = finger.getImage();
if (p != FINGERPRINT_OK) return -1;
p = finger.image2Tz();
if (p != FINGERPRINT_OK) return -1;
p = finger.fingerFastSearch();
if (p != FINGERPRINT_OK) return -1;
//Menemukan kecocokan
Serial.print("Found ID #"); Serial.print(finger.fingerID);
Serial.print(" with confidence of "); Serial.println(finger.confidence);
digitalWrite(kunci, HIGH);
delay(5000);
digitalWrite(kunci, LOW);
return finger.fingerID;
}
//-----END FINGERPRINT-----//

```

3. Program terakhir adalah program Delete, ini fungsinya untuk menghapus data Jari yang sudah terdaftar di Fingerprint Sensor

```

#include <Adafruit_Fingerprint.h> //library Adafruit Fingerprint
#include <SoftwareSerial.h> //library komunikasi serial untuk fingerprint
SoftwareSerial mySerial(2, 3); //mengatur pin RX pada pin 2 dan TX pada pin 3
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial); //pemanggilan
fungsi Adafruit_Fingerprint
void setup(){
  Serial.begin(9600); //Membuka komunikasi serial dengan mengatur baudrate
  9600
  finger.begin(57600); // mengatur kecepatan data untuk port serial sensor
  if (finger.verifyPassword()) { //pernyataan bersyarat, jika memenuhi syarat maka
    akan menjalankan program dibawah. jika tidak langsung masuk ke program else

```

```

    Serial.println("FingerPrint Sensor Ditemukan!"); //Menampilkan tulisan
"FingerPrint Sensor Ditemukan!" di serial monitor

    } else { //jika tidak memenuhi pernyataan pada program if maka masuk ke
program else

    Serial.println("FingerPrint Sensor Tidak Ditemukan! :("); //Menampilkan
tulisan "FingerPrint Sensor Tidak Ditemukan!" di serial monitor

    while (1) { delay(1); } //looping delay 1ms

    }

}

void loop(){

    DELETE(); //Memanggil fungsi DELETE

}

//-----DELETE-----//

void DELETE(){

    Serial.println("Silakan ketik ID # (dari 1 hingga 127) yang ingin Anda hapus
..."); //Menampilkan tulisan Silakan ketik ID # (dari 1 hingga 127) yang ingin
Anda hapus ...

    uint8_t id = readnumber(); //Membaca nilai id

    if (id == 0) { // ID # 0 tidak diizinkan, coba lagi!

        return;

    }

    Serial.print("Menghapus ID #"); //Menampilkan tulisan Menghapus ID # pada
serial monitor

    Serial.println(id); //Menampilkan nilai ID pada serial monitor

    deleteFingerprint(id); //Hapus fingerprint dari ID yang dipilih

}

//-----PROSES DELETE-----//

uint8_t readnumber(void) {

    uint8_t num = 0;

    while (num == 0) {

        while (! Serial.available());

        num = Serial.parseInt();

    }

}

```

```

    }
    return num;
}
//-----RETURN NUM-----//
uint8_t deleteFingerprint(uint8_t id) {
    uint8_t p = -1;
    //Hapus data dari ID yang dipilih
    p = finger.deleteModel(id);
    if (p == FINGERPRINT_OK) {
        Serial.println("Deleted!");
    } else if (p == FINGERPRINT_PACKETRECEIVEERR) {
        Serial.println("Communication error");
    }
    return p;
} else if (p == FINGERPRINT_BADLOCATION) {
    Serial.println("Could not delete in that location");
    return p;
} else if (p == FINGERPRINT_FLASHERR) {
    Serial.println("Error writing to flash");
    return p;
} else {
    Serial.print("Unknown error: 0x"); Serial.println(p, HEX);
    return p;
}
}

```

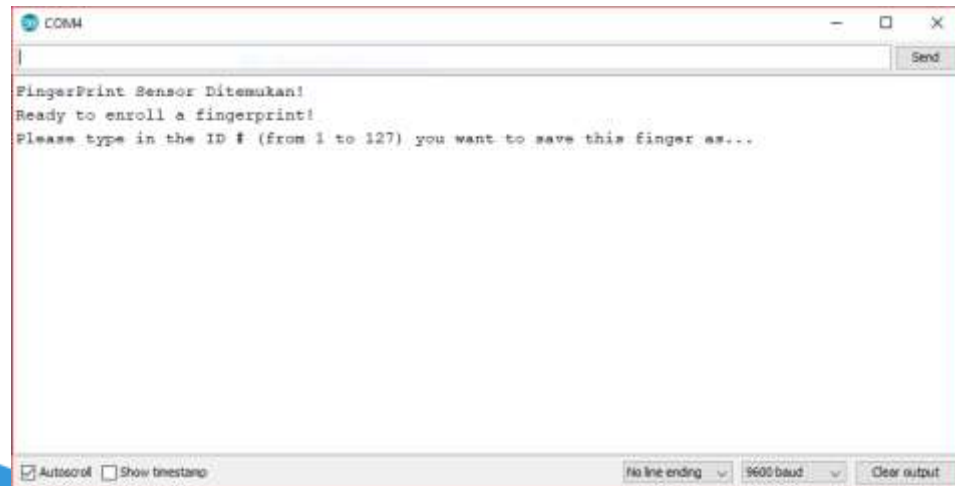
Setelah program diatas selesai diketik, hubungkan Board arduino dengan Komputer menggunakan USB Arduino kemudian upload program ke Board Arduino. Pastikan proses upload selesai dan berhasil.



## 4. Hasil

### 1. Hasil program daftar

Setelah proses upload berhasil, silahkan buka serial monitor untuk mendaftarkan jari kalian. Lalu masukkan angka 1 – 127, nomor ID untuk menyimpan data sidik jari kalian, kemudian klik send atau enter.



Kemudian silahkan letakkan jari kalian ke fingerprint sensor agar data sidik jari kalian tersimpan.



Kemudian lepas lagi jari kalian, lalu letakkan lagi untuk konfirmasi. Jika sudah ada keterangan stored, berarti sidik jari kalian telah berhasil didaftarkan

## 2. Hasil program Fingerprint

Setelah proses upload berhasil, silahkan buka serial monitor, kemudian letakkan jari kalian disensor fingerprint. Jika muncul Found ID seperti pada gambar, berarti sidik jari kalian sudah terdaftar.



## 3. Hasil program Hapus

Setelah proses upload berhasil, silahkan buka serial monitor dan masukkan no ID yang ingin kalian hapus lalu klik send atau tekan enter. Jika sudah ada tulisan Deleted! Berarti data sidik jari yang terdaftar pada ID tersebut sudah terhapus.



Nah jika kalian sudah mendaftarkan data sidik jari kalian, maka upload program kedua, yaitu program fingerprint. Sekarang cobalah masukkan sidik jari orang yang belum terdaftar. Saat memasukkan sidik jari yang belum terdaftar maka inti

besi pada solenoid door lock akan tetap memanjang sehingga pintu tidak akan terbuka. Dan sekarang cobalah memasukkan sidik jari kalian yang sudah terdaftar. Maka pada serial monitor akan menampilkan tulisan Found ID dan solenoid door lock terdapat kawat yang melingkar pada inti besi. Ketika arus listrik mengalir melalui kawat ini, maka terjadi medan magnet untuk menghasilkan energi yang akan menarik inti besi ke dalam, sehingga pintu akan terbuka.

Sampai disini dulu Tutorial Sistem Keamanan Pintu Menggunakan Android ya IOTA. Semoga Berhasil .. !!

