LAMPIRAN

LISTING PROGRAM

```
#include <Adafruit_Fingerprint.h>
#include <HardwareSerial.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <WiFi.h>
#include <WiFiClientSecure.h>
#include <UniversalTelegramBot.h>

Adafruit_Fingerprint finger = Adafruit_Fingerprint(&Serial2);

// Inisialisasi LCD I2C
LiquidCrystal_I2C lcd(0x27, 16, 2);

const int buzzerPin = 13;
const int magneticSensorPin = 27;
const int exitbtn = 26;
const int relayPin = 14;
const int ledwifi  = 5;
const int ledhijau = 18;
const int ledmerah = 19;

const unsigned long botRequestDelay = 3000;
const unsigned long timefinger = 10;
const unsigned long timeexit = 500;
unsigned long lasttimeexit;
unsigned long lasttimefinger;
unsigned long lastTimeBotRan;
int counter = 0;
int failedAttempts = 0;
bool alarmActive = false;
unsigned long alarmStartTime = 0;

// WiFi dan Telegram
const char* ssid = "doorlockiot";
const char* password = "doorlockiot";
const char* botToken = "7466405744:AAEpCfV8UtQDBoLYUJKgM19Jp7gVoPPHuEg";
// const char* chatID = "";
const char* chatID = "1320126788";

WiFiClientSecure client;
UniversalTelegramBot bot(botToken, client);

void setup() {
```

```
// Inisialisasi Serial
Serial2.begin(57600);
Serial.begin(115200);
finger.begin(57600);

delay(10);

// Inisialisasi LCD
lcd.init();
lcd.backlight();

// Inisialisasi fingerprint sensor
if (finger.verifyPassword()) {
  Serial.println("Sensor sidik jari ditemukan.");
  lcd.setCursor(0, 0);
  lcd.print("Sensor sidikjari");
  lcd.setCursor(0, 1);
  lcd.print(" --CONNECTED--  ");
} else {
  Serial.println("Sensor sidik jari tidak ditemukan, pastikan koneksi sudah benar.");
  lcd.setCursor(0, 0);
  lcd.print("Sensor sidikjari");
  lcd.setCursor(0, 1);
  lcd.print("--DISCONNECTED--");
  while (1) { delay(1); }
}
delay(1000);
lcd.clear();

Serial.println(F("Reading sensor parameters"));
finger.getParameters();
Serial.print(F("Status: 0x")); Serial.println(finger.status_reg, HEX);
Serial.print(F("Sys ID: 0x")); Serial.println(finger.system_id, HEX);
Serial.print(F("Capacity: ")); Serial.println(finger.capacity);
Serial.print(F("Security level: ")); Serial.println(finger.security_level);
Serial.print(F("Device address: ")); Serial.println(finger.device_addr, HEX);
Serial.print(F("Packet len: ")); Serial.println(finger.packet_len);
Serial.print(F("Baud rate: ")); Serial.println(finger.baud_rate);

// Inisialisasi pin
pinMode (buzzerPin, OUTPUT);
pinMode (relayPin, OUTPUT);
pinMode (ledmerah, OUTPUT);
pinMode (ledhijau, OUTPUT);
pinMode (ledwifi, OUTPUT);
pinMode (exitbtn, INPUT_PULLUP);
pinMode (magneticSensorPin, INPUT_PULLUP);
```

```
digitalWrite (ledmerah, HIGH);
digitalWrite (ledhijau, HIGH);
digitalWrite (ledwifi, HIGH);
digitalWrite (buzzerPin, LOW);
delay (1500);

digitalWrite (relayPin, HIGH);
digitalWrite (buzzerPin, HIGH);
digitalWrite (ledhijau, LOW);
digitalWrite (ledmerah, LOW);
digitalWrite (ledwifi, LOW);

// Tampilkan pesan awal
lcd.setCursor (0, 0);
lcd.print(F("INISIALISASI I/O"));
Serial.println("INISIALISASI I/O");

delay(1500);

Serial.print("Connecting");
lcd.setCursor (0, 0);
lcd.print(F("Connecting Wi-Fi"));
lcd.setCursor (0, 1);

// Inisialisasi koneksi WiFi
WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED)
{
  delay(200);
  if (++counter > 50)
    ESP.restart();
  Serial.print( "." );
}

Serial.println("Connected to WiFi");
lcd.setCursor (0, 1);
lcd.print(" --CONNECTED-- ");
digitalWrite(ledwifi, HIGH);
delay (2000);
lcd.clear();

client.setInsecure(); // Disable SSL certificate verification

lcd.setCursor (0, 0);
lcd.print(F("Connecting Tele"));
lcd.setCursor (0, 1);
```

```
  lcd.print(" --CONNECTED-- ");
  delay (2000);
  lcd.clear();

  Serial.println();
  Serial.println("Wi-Fi & Telegram");
  Serial.println(" --Connected-- ");

  lcd.setCursor (0, 0);
  lcd.print(F("Wi-Fi & Telegram"));
  lcd.setCursor (0, 1);
  lcd.print(F(" --CONNECTED-- "));

  // Kirim notifikasi ke Telegram saat sistem mulai
  bot.sendMessage(chatID, "System Online", "");
  Serial.println("Pesan Terkirim");
  delay (2000);

  lcd.clear();
}

void loop() {
  if (alarmActive) {
    // Jika alarm aktif, periksa apakah waktu alarm telah habis
    if (millis() - alarmStartTime >= 10000) {
      // Matikan buzzer dan reset alarm
      digitalWrite(buzzerPin, HIGH);
      digitalWrite(ledmerah, LOW);
      alarmActive = false;
      lcd.clear();
      bot.sendMessage(chatID, "Alarm telah dimatikan, System Kembali Normal", "");
      failedAttempts = 0;
      fingerprint();
    }
    lcd.setCursor(0, 0);
    lcd.print("SYSTEM TERKUNCI!");
    lcd.setCursor(0, 1);
    lcd.print("               ");
    return;  // Alarm aktif, hentikan eksekusi loop
  }

  fingerprint();
  // newmessage();
  // delay(10);
  // delay(5);
}
```

```
void triggerAlarm() {
  // Aktifkan buzzer dan setel waktu alarm
  digitalWrite(buzzerPin, LOW);
  digitalWrite(ledmerah, HIGH);
  alarmStartTime = millis();
  alarmActive = true;
  // lcd.setCursor(0, 0);
  // lcd.print("SYSTEM TERKUNCI!");
  // lcd.setCursor(0, 1);
  // lcd.print("              ");
  bot.sendMessage(chatID, "Alarm Aktif!", "");
  // delay(250);
  // return;
}

void resetSystem() {
  // Reset semua variabel dan status sistem
  failedAttempts = 0;
  alarmActive = false;
  lcd.clear();
  bot.sendMessage(chatID, "Sistem di-reset", "");
}

void fingerprint() {
  if (digitalRead(magneticSensorPin) == LOW) {
    if(millis() - lasttimefinger > timefinger) {
    lcd.setCursor(0,0);
    lcd.print(" Silahkan Scan  ");
    lcd.setCursor(0, 1);
    lcd.print("Sidik Jari Anda");
     if (finger.getImage() == FINGERPRINT_OK){
     Serial.println("image taken");
       if (finger.image2Tz() == FINGERPRINT_OK) {
         Serial.println("Image Converted") ;
         if(finger.fingerFastSearch() == FINGERPRINT_OK) {
           Serial.println("Found a print match!");
           digitalWrite(ledhijau, HIGH);
           lcd.clear();
           lcd.setCursor(0, 0);
           lcd.print("Akses Diterima  ");
           lcd.setCursor(0, 1);
           lcd.print("ID: "); lcd.print(finger.fingerID);
           delay(1500);
           lcd.clear();
           lcd.setCursor(0, 0);
           lcd.print("Send Notif Tele");
           bot.sendMessage(chatID, "Akses diterima, ID: " + String(finger.fingerID), "");
```

```
  lcd.clear();
  digitalWrite(relayPin, LOW); // Buka kunci pintu
  lcd.setCursor(0, 0);
  lcd.print("  SYSTEM UNLOCK ");
  lcd.setCursor(0, 1);
  lcd.print("  SILAKAN MASUK ");
  delay(10000); // Biarkan pintu terbuka selama 10 detik
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("  Locking DOOR ");
  delay(2000); // Tunda untuk menampilkan pesan
  digitalWrite(ledhijau, LOW);
  digitalWrite(relayPin, HIGH); // Kunci kembali pintu
  failedAttempts = 0; // Reset hitungan percobaan gagal
  delay(1000);
  lcd.clear();
 }
 else if (finger.fingerFastSearch() == FINGERPRINT_NOTFOUND) {
  Serial.println("Did not find a match");
  lcd.setCursor(0, 0);
  lcd.print("  data Finger   ");
  lcd.setCursor(0,1);
  lcd.print("Tidak Ditemukan ");
  delay(500);
  lcd.clear();

  failedAttempts++;
  if (failedAttempts >= 3) {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Akses Ditolak   ");
    lcd.setCursor(0, 1);
    lcd.print("Alarm Aktif     ");

    delay(1000);
    triggerAlarm();
    // return;  // Alarm aktif, hentikan eksekusi loop
  }

  else {
    digitalWrite(ledmerah, HIGH);
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Akses Ditolak   ");
    lcd.setCursor(0, 1);
    lcd.print("Percobaan: ");
    lcd.print(failedAttempts);
```

```
          bot.sendMessage(chatID, "Akses ditolak, Percobaan: " + String(failedAttempts),
"");
          delay(2000); // Tunda untuk menampilkan pesan
          digitalWrite(ledmerah, LOW);
          lcd.clear();
         }
        }
        // else {
        //   Serial.println("unknown error");
        // }
       }

      // else {
      //   Serial.println("fail take fingerprint!!");
      // }
     }

     lastTimeBotRan = millis();
   }

   if(millis() - lasttimeexit > timeexit) {
    if (digitalRead(exitbtn) == LOW) {
      lcd.clear();
         lcd.setCursor(0, 0);
         lcd.print("  Pintu Dibuka  ");
         lcd.setCursor(0, 1);
         lcd.print("   Dari Dalam   ");
         delay(1500);
         lcd.clear();
         lcd.setCursor(0, 0);
         lcd.print("Send Notif Tele");
         bot.sendMessage(chatID, "Pintu Berhasil dibuka Dari Dalam", "");
         lcd.clear();
         digitalWrite(relayPin, LOW); // Buka kunci pintu
         lcd.setCursor(0, 0);
         lcd.print("  SYSTEM UNLOCK ");
         lcd.setCursor(0, 1);
         lcd.print("  PINTU DIBUKA  ");
         delay(10000); // Biarkan pintu terbuka selama 10 detik
         lcd.clear();
         lcd.setCursor(0,0);
         lcd.print("  Locking DOOR ");
         delay(2000); // Tunda untuk menampilkan pesan
         digitalWrite(ledhijau, LOW);
         digitalWrite(relayPin, HIGH); // Kunci kembali pintu
         delay(1000);
         lcd.clear();
```

```
    }
    lasttimeexit = millis();
   }
  }

  else {
   lcd.clear();
   lcd.setCursor(0, 0);
   lcd.print(" PINTU DIBUKA!! ");
   lcd.setCursor(0, 1);
   lcd.print("LOCKING SYSTEM!!");
   delay(1000);
   triggerAlarm();
   // return;  // Alarm aktif, hentikan eksekusi loop
  }
}

void newmessage() {
 if (millis() - lastTimeBotRan > botRequestDelay)  {
   int numNewMessages = bot.getUpdates(bot.last_message_received + 1);

   while(numNewMessages) {
    Serial.println("got response");
    handleNewMessages(numNewMessages);
    numNewMessages = bot.getUpdates(bot.last_message_received + 1);
   }
   lastTimeBotRan = millis();
 }
}

void handleNewMessages(int numNewMessages) {
 Serial.println("handleNewMessages");
 Serial.println(String(numNewMessages));

 for (int i=0; i<numNewMessages; i++) {
   // Chat id of the requester
   String chat_id = String(bot.messages[i].chat_id);
   if (chat_id != chatID){
    bot.sendMessage(chat_id, "Unauthorized user", "");
    continue;
   }

   // Print the received message
   String text = bot.messages[i].text;
   Serial.println(text);

   String from_name = bot.messages[i].from_name;
```

```
if (text == "/start") {
  String welcome = "Welcome, " + from_name + ".\n";
  welcome += "Silakan Gunakan Perintah Dibawah ini.\n\n";
  welcome += "/Resetsystem Untuk me-reset System \n";
  welcome += "/bukapintu untuk membuka pintu dari jarak jauh \n";
  bot.sendMessage(chatID, welcome, "");
}

else if (text == "/Resetsystem") {
  // bot.sendMessage(chatID, "SYSTEM DIHIDUPKAN", "");
     resetSystem();
}

else if (text == "/bukapintu") {
     digitalWrite(ledhijau, HIGH);
     lcd.clear();
     lcd.setCursor(0, 0);
     lcd.print("  Pintu Dibuka  ");
     lcd.setCursor(0, 1);
     lcd.print("   Jarak Jauh   ");
     delay(500);
     lcd.clear();
     lcd.setCursor(0, 0);
     lcd.print("Send Notif Tele");
     bot.sendMessage(chatID, "Pintu Berhasil Dibuka", "");
     lcd.clear();
     digitalWrite(relayPin, LOW); // Buka kunci pintu
     lcd.setCursor(0, 0);
     lcd.print("  SYSTEM UNLOCK ");
     lcd.setCursor(0, 1);
     lcd.print("  SILAKAN MASUK ");
     delay(10000); // Biarkan pintu terbuka selama 10 detik
     lcd.clear();
     lcd.setCursor(0,0);
     lcd.print("   HARAP PINTU  ");
     lcd.setCursor(0,1);
     lcd.print(" DITUTUP KEMBALI");
     delay(5000); // Tunda untuk menampilkan pesan
     digitalWrite(ledhijau, LOW);
     digitalWrite(relayPin, HIGH); // Kunci kembali pintu
     failedAttempts = 0; // Reset hitungan percobaan gagal
     delay(1500);
     lcd.clear();
}

else {
```

```
        bot.sendMessage(chatID, "Perintah tidak dikenal", "");
    }
  }
}

Adafruit_Fingerprint finger = Adafruit_Fingerprint(&Serial2);

uint8_t id;

void setup()
{
  Serial.begin(9600);
  Serial2.begin(57600);
  // 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); }
  }

  Serial.println(F("Reading sensor parameters"));
  finger.getParameters();
  Serial.print(F("Status: 0x")); Serial.println(finger.status_reg, HEX);
  Serial.print(F("Sys ID: 0x")); Serial.println(finger.system_id, HEX);
  Serial.print(F("Capacity: ")); Serial.println(finger.capacity);
  Serial.print(F("Security level: ")); Serial.println(finger.security_level);
  Serial.print(F("Device address: ")); Serial.println(finger.device_addr, HEX);
  Serial.print(F("Packet len: ")); Serial.println(finger.packet_len);
  Serial.print(F("Baud rate: ")); Serial.println(finger.baud_rate);
}

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);

 while (! getFingerprintEnroll() );
}

uint8_t getFingerprintEnroll() {

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

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

  return true;
}
```

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

LEMBAR BIMBINGAN LAPORAN TUGAS AKHIR

Nama Mahasiswa	:	M. Sami Ramadhan
NIM	:	062130701720
Jurusan/Program Studi	:	Teknik Komputer/ DIII Teknik Komputer
Dosen Pembimbing	:	Adi Sutrisman, S.Kom., M.Kom.
Judul	:	Sistem Keamanan Pintu Otomatis Menggunakan Sidik Jari (*Fingerprint*) Berbasis IOT

NO	TANGGAL	URAIAN	PARAF PEMBIMBING
	24 / 06 2024	Revisi Bab I	
	26 / 06 2024	ACC Bab I	
	27 / 06 2024	Revisi Bab II	
	01 / 07 2024	ACC Bab II	
	02 / 07 2024	Revisi Bab III	
	04 / 07 2024	ACC Bab III	
	09 / 07 2024	Revisi Bab IV & V	
	10 / 07 2024	ACC Bab IV & V	
	11 / 07 2024	Rekomendasi Ujian	

Palembang 2024

Mengetahui,
Ketua jurusan

Azwardi, S.T., M.T.
NIP. 197005232005011004

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

LEMBAR BIMBINGAN LAPORAN TUGAS AKHIR

Nama Mahasiswa	:	M. Sami Ramadhan
NIM	:	062130701720
Jurusan/Program Studi	:	Teknik Komputer/ DIII Teknik Komputer
Dosen Pembimbing	:	Arsia Rini, S.Kom., M.Kom.
Judul	:	Sistem Keamanan Pintu Otomatis Menggunakan Sidik Jari *(Fingerprint)* Berbasis IOT

NO	TANGGAL	URAIAN	PARAF PEMBIMBING
	25/06 2024	Bab I Pendahuluan	
	26/06 2024	Bab II Tinjauan Pustaka	
	28/06 2024	Bab III Rancang Bangun	
	03/07 2024	Bab IV Hasil Pembahasan	
	05/07 2024	Bab IV Pembuatan Alat	
		Pembuatan Alat	
		Testing Alat	
	08/07 2024	Bab IV Pembahasan	
	09/07 2024	Bab V Simpulan Saran	
	09/07 2024	Acc Ujian LA	

Palembang 2024
Mengetahui,
Ketua jurusan

Azwardi, S.T., M.T.
NIP. 197005232005011004

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,
RISET DAN TEKNOLOGI
POLITEKNIK NEGERI SRIWIJAYA
JURUSAN TEKNIK KOMPUTER
Jalan Srijaya Negara, Palembang 30139. Telp. 0711-353414
Website : www.polsri.ac.id E-mail : info@polsri.ac.id

REKOMENDASI UJIAN TUGAS AKHIR

Pembimbing Laporan Tugas Akhir, memberikan rekomendasi ujian laporan tugas akhir kepada,

Nama Mahasiswa	:	M. Sami Ramadhan
NIM	:	062130701720
Jurusan/Program Studi	:	Teknik Komputer/DIII Teknik Komputer
Judul Tugas Akhir	:	Sistem Keamanan Pintu Otomatis Menggunakan Sidik Jari (*Fingerprint*) Berbasis IOT

Mahasiswa tersebut telah memenuhi persyaratan dan dapat mengikuti Ujian Laporan Tugas Akhir, pada Tahun Akademik 2023/2024.

Palembang, Juli 2024

Disetujui oleh,

Pembimbing I	Pembimbing II
Adi Sutrisman, S.Kom., M.Kom.	**Arsia Rin, S.Kom., M.Kom.**
NIP. 197503052001121005	**NIP. 198809222020122014**

No. Dok. :	Tgl. Berlaku :	No. Rev. :
	KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET DAN TEKNOLOGI POLITEKNIK NEGERI SRIWIJAYA Jalan Srijaya Negara, Palembang 30139. Telp. 0711-353414 **Website :** www.polsri.ac.id **E-mail :** info@polsri.ac.id	
	REVISI TUGAS AKHIR (TA)	

Dosen Penguji : Yulian Mirza, S.T., M.Kom
Nama Mahasiswa : M. Sami Ramadhan
NIM : 062130701720
Jurusan /Program Studi : Teknik Komputer/DIII Teknik Komputer

Judul Laporan Akhir : Sistem Keamanan Pintu Otomatis Menggunakan Sidik Jari (*Fingerprint*) Berbasis IOT

No	Uraian Revisi	Paraf
	Pembahasan bandingkan data uji dengan coding	

Palembang,
Dosen Penguji,

(Yulian Mirza, S.T., M.Kom)
NIP : 196607121990031003

No. Dok. :	Tgl. Berlaku :	No. Rev. :
	**KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET DAN TEKNOLOGI** **POLITEKNIK NEGERI SRIWIJAYA** Jalan Srijaya Negara, Palembang 30139. Telp. 0711-353414 **Website :** www.polsri.ac.id **E-mail :** info@polsri.ac.id	
	REVISI TUGAS AKHIR (TA)	

Dosen Penguji : Alan Novi Tumpunu, S.T., M.T
Nama Mahasiswa : M. Sami Ramadhan
NIM : 062130701720
Jurusan /Program Studi : Teknik Komputer/DIII Teknik Komputer

Judul Laporan Akhir : Sistem Keamanan Pintu Otomatis Menggunakan Sidik Jari (*Fingerprint*) Berbasis IOT

No	Uraian Revisi	Paraf
	(Mohon Revisi Laporan.	

Palembang, 18 juli 2024.
Dosen Penguji,

(Alan Novi Tumpunu, S.T., M.T)
NIP : 197611082000031002

No. Dok. :	Tgl. Berlaku :	No. Rev. :
	**KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET DAN TEKNOLOGI** **POLITEKNIK NEGERI SRIWIJAYA** Jalan Srijaya Negara, Palembang 30139. Telp. 0711-353414 **Website :** www.polsri.ac.id **E-mail :** info@polsri.ac.id	
	REVISI TUGAS AKHIR (TA)	

Dosen Penguji : Hartati Deviana, S.T.,M.Kom.
Nama Mahasiswa : M. Sami Ramadhan
NIM : 062130701720
Jurusan /Program Studi : Teknik Komputer/DIII Teknik Komputer

Judul Laporan Akhir : Sistem Keamanan Pintu Otomatis Menggunakan Sidik Jari (*Fingerprint*) Berbasis IOT

No	Uraian Revisi	Paraf
	tambahkan analisa dr tabel.	

Palembang,
Dosen Penguji,

(Hartati Deviana, S.T.,M.Kom.)
NIP : 197405262008122001

No. Dok. :	Tgl. Berlaku :	No. Rev. :
	KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET DAN TEKNOLOGI POLITEKNIK NEGERI SRIWIJAYA Jalan Srijaya Negara, Palembang 30139. Telp. 0711-353414 **Website :** www.polsri.ac.id **E-mail :** info@polsri.ac.id	
	REVISI TUGAS AKHIR (TA)	

Dosen Penguji : Rian Rahmanda Putra, S.Kom., M.Kom
Nama Mahasiswa : M. Sami Ramadhan
NIM : 062130701720
Jurusan /Program Studi : Teknik Komputer/DIII Teknik Komputer

Judul Laporan Akhir : Sistem Keamanan Pintu Otomatis Menggunakan Sidik Jari (*Fingerprint*) Berbasis IOT

No	Uraian Revisi	Paraf
	Sudah dihilis di softcopy yang dikirim via WA	

Palembang,
Dosen Penguji,

(Rian Rahmanda Putra, S.Kom., M.Kom)
NIP : 198901252019031013

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,
RISET DAN TEKNOLOGI
POLITEKNIK NEGERI SRIWIJAYA
JURUSAN TEKNIK KOMPUTER
Jalan Srijaya Negara, Palembang 30139. Telp. 0711-353414
Website : www.polsri.ac.id E-mail : info@polsri.ac.id

PELAKSANAAN REVISI UJIAN TUGAS AKHIR

Nama Mahasiswa	: M. Sami Ramadhan
NIM	: 062130701720
Jurusan/Program Studi	: Teknik Komputer/DIII Teknik Komputer
Judul Tugas Akhir	: Sistem Keamanan Pintu Otomatis Menggunakan Sidik Jari (*Fingerprint*) Berbasis IOT

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

No	Komentar	Nama Dosen Penguji	Tanggal/ bulan	Tanda Tangan
1.	ACC	Yulian Mirza, S.T., M.Kom	31 / 07 2024	
2.	ACC	Ir. Alan Novi Tompunu, S.T., M.T., IPM., ASEAN Eng	9/8 2024	
3.	ACC	Hartati Deviana, S.T., M.Kom	01 / 08 2024	
4.	ACC	Rian Rahmanda Putra, S.Kom., M.Kom	01/08 2024	

Palembang,
Ketua Penguji

Yulian Mirza, S.T., M.Kom
NIP : 196607121990031003