Lampiran 3 Source Code

```
#define BLYNK_TEMPLATE_ID "TMPL6t-j8CWUH"

#define BLYNK_TEMPLATE_NAME "kotak amal"

#define BLYNK_AUTH_TOKEN
"nXkMRwZ9kc2iVbHFPKPC17Q0YBq9gmyT"

#define BLYNK_PRINT Serial

#include <WiFi.h>

#include <WiFiClient.h>

#include <BlynkSimpleEsp32.h>

#include <WiFiClientSecure.h>

#include <UniversalTelegramBot.h>

#include <ArduinoJson.h>




const char* ssid = "Gemoy";

const char* password = "23456789";

#define BOTtoken "6749096303:AAEc2RwXnKPiFK9Ze1xoilfyKuB9XSg_trU"
// your Bot Token (Get from Botfather)

#define CHAT_ID "1390010676"
```

```
WiFiClientSecure client;

UniversalTelegramBot bot(BOTtoken, client);



#include <HardwareSerial.h>

HardwareSerial SerialPort(1); // use UART2

#include <Wire.h>

#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);

#include <Adafruit_Fingerprint.h>

Adafruit_Fingerprint sidikJari =

  Adafruit_Fingerprint(&SerialPort);



#include <Ultrasonic.h>

Ultrasonic ultrasonic1(12, 14);



int jarak1;
```

```
const int Tdaftar = 25;

const int Magnetic = 13;


const int selenoid = 26;


const int buzzer = 4;

int daftarfp, Mode = 0, hitdaftar = 0;


int magnetic;

int stlock = 0;

int idSidikJari;


boolean tele = false, telepenuh = false, bpenuh = false;;

String stt = "Kunci ON", stp = "Tersedia", nama = "-";


boolean tek1 = true, tek2 = true, bdaftar = true;

int sthit = 0, hit = 0;

uint8_t id;
```

```
BlynkTimer timer;

uint8_t readnumber(void) {
  uint8_t num = 0;


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


void sendSensor()
{
  jarak1 = 14 - ultrasonic1.read();


  daftarfp = digitalRead(Tdaftar);
  magnetic = digitalRead(Magnetic);
```

```
int jhasil = (jarak1 / 11.0) * 100 ;


if (jarak1 <= 0) {
  jarak1 = 0;
}
if (jhasil <= 0) {
  jhasil = 0;
}
if (jhasil >= 100) {
  jhasil = 100;
}


if (jhasil >= 81 && magnetic == LOW) {
  stp = "Full";
  if (bpenuh == false) {
    bot.sendMessage(CHAT_ID, "Penuh!!!!!!", "");
```

```
    bpenuh = true;
  }
}
else if (jhasil >= 54) {
  stp = "3/4";
}
else if (jhasil >= 27) {
  stp = "setengah";
}
else if (jhasil >= 9) {
  stp = "1/4";
}
else if (jhasil <= 9) {
  stp = "kosong";
  bpenuh = false;
}

if (daftarfp == LOW) {
```

```
  if (bdaftar == true) {
    Mode++;
    bdaftar = false;
  }
  if (Mode >= 2) {
    Mode = 0;
  }
  Serial.println("mode = " + String(Mode));
} else {
  bdaftar = true;
}


if (Mode == 1) {
  lcd.setCursor(0, 0);
  lcd.print("Ready to enroll");
  lcd.print("      ");
  lcd.setCursor(0, 1);
```

```
    lcd.print("fingerprint!");
    lcd.print("       ");
    Serial.println("Ready to enroll a fingerprint!");
    Serial.println("Please type in the ID # (from 1 to 127) you want to save this
finger as...");
    id = readnumber();
    if (id == 0) {// ID #0 not allowed, try again!
      return;
    }
    Serial.print("Enrolling ID #");
    Serial.println(id);


    getFingerprintEnroll();
  } else {
    idSidikJari = perolehID();


    if (idSidikJari == 1) {
      nama = "Ronaldo";
```

```
  Blynk.virtualWrite(V4, nama);
  Blynk.virtualWrite(V3, idSidikJari);
  stlock = 1;
  Serial.println(idSidikJari);
  lcd.setCursor(0, 0);
  lcd.print("Akses Di Terima");
  buzz();
  digitalWrite(selenoid, LOW);
  sthit = 1;
  bot.sendMessage(CHAT_ID, "AKSES DITERIMA An :" + String(nama), "");
} else    if (idSidikJari == 2) {
  nama = "Messi";
  Blynk.virtualWrite(V4, nama);
  Blynk.virtualWrite(V3, idSidikJari);
  stlock = 1;
  Serial.println(idSidikJari);
  lcd.setCursor(0, 0);
  lcd.print("Akses Di Terima");
```

```
  buzz();
  digitalWrite(selenoid, LOW);
  sthit = 1;
  bot.sendMessage(CHAT_ID, "AKSES DITERIMA An :" + String(nama), "");
}
else    if (idSidikJari == 3) {
  nama = "Neymar";
  Blynk.virtualWrite(V4, nama);
  Blynk.virtualWrite(V3, idSidikJari);
  stlock = 1;
  Serial.println(idSidikJari);
  lcd.setCursor(0, 0);
  lcd.print("Akses Di Terima");
  buzz();
  digitalWrite(selenoid, LOW);
  sthit = 1;
  bot.sendMessage(CHAT_ID, "AKSES DITERIMA An :" + String(nama), "");
}
```

```
  else if (sidikJari.confidence >= 2 && sidikJari.confidence <= 50) {
    nama = "Akses Ditolak";
    Blynk.virtualWrite(V4, nama);
    digitalWrite(buzzer, HIGH);
    lcd.setCursor(0, 0);
    lcd.print("AKASES DITOLAK");
    bot.sendMessage(CHAT_ID, "AKSES DITOLAK!!!!!", "");
    delay(2000);
    digitalWrite(buzzer, LOW);
    sidikJari.confidence = 0;
  }
}


if (magnetic == HIGH && stlock == 0) {
  digitalWrite(buzzer, HIGH);
  Serial.println("dibuka paksa");
  if (tele == false) {
    bot.sendMessage(CHAT_ID, "DIBUKA PAKSA!!!!!!", "");
```

```
    tele = true;
  }
  stt = "DIBUKA PAKSA!!!!!!";
}
else if (magnetic == LOW && stlock == 1) {
  stlock = 2;
  digitalWrite(buzzer, LOW);
  stt = "Kunci Off";
}
else if (magnetic == HIGH && stlock == 2) {
  stlock = 3;
  digitalWrite(buzzer, LOW);
  stt = "Kunci Off";
}
else if (magnetic == LOW && stlock == 3) {
  stlock = 0;
  digitalWrite(buzzer, LOW);
  stt = "Kunci Off";
```

```
}

else if (magnetic == LOW && stlock == 0) {

  digitalWrite(buzzer, LOW);

  stt = "Kunci ON";

  tele = false;

}



lcd.setCursor(0, 0);

lcd.print(jhasil);

lcd.print(" %                ");



lcd.setCursor(0, 1);

lcd.print(stt);

lcd.print("                    ");




// untuk waktu mengunci selenoid

if (runEvery(1000) && sthit  == 1) {
```

```
  hit++;
  if (hit >= 10) {
    digitalWrite(selenoid, HIGH);
    hit = 0;
    sthit = 0;
  }
  Serial.println("hitung = " + String(hit));
}
//lcd.print(" %                ");
Serial.println(stlock);
Serial.println(magnetic);
Serial.println("jarak1 = " + String(jarak1));
//  Serial.println("jarak2 = " + String(jarak2));
Serial.println("hasil = " + String(jhasil));
Blynk.virtualWrite(V0, jhasil);
Blynk.virtualWrite(V1, stt);
Blynk.virtualWrite(V2, stp);
```

```
}


void setup()
{
  // Debug console
  Serial.begin(115200);

  Blynk.begin(BLYNK_AUTH_TOKEN, ssid, password);
  Serial.println(ssid);

  WiFi.begin(ssid, password);
  client.setCACert(TELEGRAM_CERTIFICATE_ROOT); // Add root certificate
for api.telegram.org

  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
  }
```

```
timer.setInterval(1000L, sendSensor);

lcd.begin();

lcd.backlight();

lcd.print("Hello....");

lcd.setCursor(0, 0);



SerialPort.begin(57600, SERIAL_8N1, 16, 17);

pinMode(Tdaftar, INPUT_PULLUP);

pinMode(Magnetic, INPUT_PULLUP);

pinMode(buzzer, OUTPUT);

pinMode(selenoid, OUTPUT);

digitalWrite(selenoid, HIGH);



if (!sidikJari.verifyPassword())

{

  Serial.println("Sensor sidik jari tidak dikenal");

  while (1);
```

```
  }


  Serial.println("Siap untuk membaca sidik jari!");
  buzz();
  bot.sendMessage(CHAT_ID, "Hellowww", "");
}


void loop()
{
  Blynk.run();
  timer.run();
}


int perolehID()
{
  uint8_t hasil = sidikJari.getImage();
  if (hasil != FINGERPRINT_OK)
    return -1;
```

```
  hasil = sidikJari.image2Tz();
  if (hasil != FINGERPRINT_OK)
    return -1;

  hasil = sidikJari.fingerFastSearch();
  if (hasil != FINGERPRINT_OK)
    return -1;

  Serial.print("Found ID #"); Serial.print(sidikJari.fingerID);
  Serial.print(" with confidence of "); Serial.println(sidikJari .confidence);
  return sidikJari.fingerID;
}

void buzz() {
  digitalWrite(buzzer, HIGH);
```

```
  delay(200);
  digitalWrite(buzzer, LOW);
  delay(200);
  digitalWrite(buzzer, HIGH);
  delay(200);
  digitalWrite(buzzer, LOW);
}
void buz() {
  digitalWrite(buzzer, HIGH);
  delay(1000);
  digitalWrite(buzzer, LOW);
}

uint8_t getFingerprintEnroll() {

  int p = -1;
  Serial.print("Waiting for valid finger to enroll as #"); Serial.println(id);
  while (p != FINGERPRINT_OK) {
```

```
p = sidikJari.getImage();
switch (p) {
  case FINGERPRINT_OK:
    Serial.println("Image taken");
    break;
  case FINGERPRINT_NOFINGER:
    Serial.println(".");
    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 = sidikJari.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 = sidikJari.getImage();
}
Serial.print("ID "); Serial.println(id);
p = -1;
Serial.println("Place same finger again");
while (p != FINGERPRINT_OK) {
```

```
p = sidikJari.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 = sidikJari.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 = sidikJari.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 = sidikJari.storeModel(id);
if (p == FINGERPRINT_OK) {
  Serial.println("Stored!");
  Mode = 0;
} 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;
  //  Mode = 0;
}

boolean runEvery(unsigned long interval)
{
  static unsigned long previousMillis = 0;
  unsigned long currentMillis = millis();
  if (currentMillis - previousMillis >= interval)
  {
```

```
  previousMillis = currentMillis;

  return true;

 }

 //untuk mengetahui nilai millis

 //Serial.println("currentMillis === " + String(currentMillis));

 //Serial.println("previousMillis === " + String(previousMillis));

 // Serial.println("interval === " + String(interval));

 return false;

}
```

Lampiran 4 Tempat Penelitian