

DAFTAR PUSTAKA

- [1] M. M. Thoyyib, Sistem Keamanan Sepeda Motor Dari Perampasan Menggunakan Sms dan GPS Berbasis Arduino Nano, Proyek Akhir. Yogyakarta, 2017.
- [2] Badan Pusat Statistik, “Perkembangan Jumlah Kendaraan Bermotor Menurut Jenis, 1949-2018,” *www.bps.go.id*, 2018. <https://www.bps.go.id/linkTableDinamis/view/id/1133> (accessed Feb. 24, 2022).
- [3] STATISTIK KRIMINAL 2021. Badan Pusat Statistik, 2021.
- [4] D. V. Azkiya, “Kejahatan Curanmor Paling Sering Terjadi di Sulawesi Tengah pada 2020,” *databoks*, 2021. <https://databoks.katadata.co.id/datapublish/2021/10/28/kejahatan-curanmor-paling-sering-terjadi-di-sulawesi-tengah-pada-2020> (accessed Mar. 02, 2022).
- [5] Radlis Muh, “Maling Motor Ini Bisa Bobol Pengaman Lubang Kunci Motor,” *Tribunnews.com*, 2022. <https://www.tribunnews.com/regional/2016/11/01/waspada-maling-motor-ini-bisa-bobol-pengaman-lubang-kunci-motor>.
- [6] K. Munzilin, Perancangan Sistem Aktivasi Pengaman Sepeda Motor Menggunakan Radio Frequency Identification (RFID) E-KTP Serta GPS, Terintegrasi Telegram Berbasis Arduino. Riau: Universitas Islam Negeri Sultan Syarif Kasim Riau, 2021.
- [7] A. Shofiudin, Sistem Pengamanan Ganda Pada Sepeda Motor Menggunakan Sensor *Fingerprint* Dan Remote Control Rf Berbasis Arduino. Semarang, 2020.
- [8] T. Hidayat, Sistem monitoring dan pengaman motor menggunakan remote cerdas berbasis android dengan gps *Google api*, Skripsi. Jakarta, 2019.
- [9] H. Simanjuntak, “Sistem Keamanan Sepeda Motor Berbasis Arduino Menggunakan GPS Dan Radio Frequency Identification (RFID),” *Inf. Commun. Technol.*, vol. 20, no. 6, pp. 47–53, 2020.
- [10] Dhika Aulia, Keamanan Sepeda Motor Menggunakan Kamera Dan *Fingerprint*. 2021.
- [11] Suprianto, Keamanan Sepeda Motor Dengan Pelacakan Lokasi Secara Live Tracking GPS Terintegrasi, Skripsi. Semarang: Universitas Negeri Semarang, 2019.
- [12] E. F. Y. Aritonang, Perancangan Sistem Keamanan Sepeda

- Motor Dengan Sensor *Fingerprint*, SMS Gateway, Dan GPS Ttracker Berbasis Atmega328, Skripsi., vol. 1, no. 3. Sumatra Utara, 2018.
- [13] R. Akbar, Sistem Kunci Kendaraan Bermotor Menggunakan Radio Frequency Identification (RFID) Dan SIM Berbasis *NodeMCU* ESP32, Skripsi. Riau, 2020.
- [14] V. Cossalter, *Motorcycle Dynamics*. 2006.
- [15] C. Honda, “Apa Saja Sih Kelebihan Honda BeAT eSP,” 2014. <https://www.hondacengkareng.com/kelebihan-honda-beat-esp-dibanding-honda-beat-pendahulunya/> (accessed Jul. 30, 2022).
- [16] L. A. Sandy, R. J. Akbar, and R. R. Hariadi, “Rancang Bangun Aplikasi Chat pada Platform Android dengan Media Input Berupa Canvas dan Shareable Canvas untuk Bekerja dalam Satu Canvas Secara Online,” *J. Tek. ITS*, vol. 6, no. 2, 2017, doi: 10.12962/j23373539.v6i2.23782.
- [17] A. N. Jonar, “Belajar Menggunakan *Database Firebase*.” <https://arthanugraha.com/yuk-belajar-menggunakan-Database-Firebase-1/>
- [18] Erinta, “Perangkat Lunak Arduino IDE,” 2021. <https://www.kmtech.id/post/mengenal-perangkat-lunak-arduino-ide> (accessed Jul. 29, 2022).

LAMPIRAN A

Listing Program Arduino

```
#include <TinyGPS++.h>
#include <SoftwareSerial.h>
#include <WiFi.h>
#include <FirebaseESP32.h>
#include <addons/TokenHelper.h>
#include <addons/RTDBHelper.h>
#include <Adafruit_Fingerprint.h>
#include "FS.h"
#include "SPIFFS.h"
#define WIFI_SSID "iya"
#define WIFI_PASSWORD "12345678"
#define API_KEY
"AIzaSyDriaDw71VjAYf7mITJCzOTWQ2Vqyiu9bw"
#define DATABASE_URL "gpsfingger-default-rtdb.Firebaseio.com"
//<DatabaseName>.Firebaseio.com or
<DatabaseName>.<region>.FirebaseDatabase.app
#define USER_EMAIL "gpsfinggerpnc@gmail.com"
#define USER_PASSWORD "gpsfingger2022."
#define mySerial Serial2
#define R_kontak 5
#define R_Starter 21
#define R_Alarm 19
#define R_ 18
byte en_add, id, en_Alarm;
bool state;
int x, flag_on;
double Latitude , Longitude;
String lat_str = "0", lng_str = "0";
long proveus;
long interval = 5000, prov;
int flagSend;
String kontak_str = "0", Starter_str = "0", Alarm_str = "0", del;
int kontakInt, StarterInt, AlarmInt;
byte en_Fingerprint = 0;
FirebaseData fbdo;
```

```

FirebaseAuth auth;
FirebaseConfig config;
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);
TinyGPSPlus gps;
SoftwareSerial SerialGPS;
void setup()
{
//program untuk firebase data base dan aplikasi//
  Serial.begin(115200);
  SerialGPS.begin(9600, SWSERIAL_8N1, 33, 32, false);
  SPIFFS.begin();
  id = readFile(SPIFFS, "/id.txt").toInt();
  Serial.println("id:" + String(id));
  if (id > 254) {
    writeFile(SPIFFS, "/id.txt", "0");
    delay(1000);
    id = readFile(SPIFFS, "/id.txt").toInt();
  }
  Serial.println("id:" + String(id));
  pinMode(2, OUTPUT);
  digitalWrite(2, LOW);
  setup_Fingerprint();
  add_finger();
  pinMode(R_kontak, OUTPUT);
  pinMode(R_Starter, OUTPUT);
  pinMode(R_Alarm, OUTPUT);
  pinMode(R_, OUTPUT);
  digitalWrite(R_kontak, LOW);
  digitalWrite(R_Starter, LOW);
  digitalWrite(R_Alarm, LOW);
  digitalWrite(R_, LOW);
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("Connecting to Wi-Fi");
  while (WiFi.status() != WL_CONNECTED)
  {
    Serial.print(".");
    delay(300);
  }
  Serial.println();

```

```

Serial.print("Connected with IP: ");
Serial.println(WiFi.localIP());
Serial.println();
Serial.printf("Firebase Client v%s\n\n",
FIREBASE_CLIENT_VERSION);
config.api_key = API_KEY;
auth.user.email = USER_EMAIL;
auth.user.password = USER_PASSWORD;
config.Database_url = DATABASE_URL;
config.token_status_callback = tokenStatusCallback; //see
addons/TokenHelper.h
Firebase.begin(&config, &auth);
Firebase.reconnectWiFi(true);
Firebase.setString(fbdo, F("/kontak"), "0");
Firebase.setString(fbdo, F("/Starter"), "0");
Firebase.setString(fbdo, F("/Alarm"), "0");
digitalWrite(2, HIGH);
}
void loop()
{
read_gps();
delay(10);
getFingerprintID();
delay(50);
if (en_Fingerprint == 1) {
flag_on++;
delay(1000);
}
if (en_Alarm == 1) {
if ((millis() - prov) > 500) {
state = !state;
prov = millis();
}
digitalWrite(R_Alarm, state);
}
if (flag_on > 4) {
flag_on = 4;
}
else if (flag_on < 0) {

```

```

flag_on = 0;
}
Serial.println("flag_on: " + String(flag_on));
if (flagSend == 0) {
kontak_str = Firebase.getString(fbdo, F("/kontak")) ? fbdo.to<const
char *>() : fbdo.errorReason().c_str();
kontakInt = kontak_str.toInt();
flagSend = 1;
}
else if (flagSend == 1) {
Starter_str = Firebase.getString(fbdo, F("/Starter")) ? fbdo.to<const
char *>() : fbdo.errorReason().c_str();
StarterInt = Starter_str.toInt();
flagSend = 2;
}
else if (flagSend == 2) {
Alarm_str = Firebase.getString(fbdo, F("/Alarm")) ? fbdo.to<const
char *>() : fbdo.errorReason().c_str();
AlarmInt = Alarm_str.toInt();
flagSend = 3;
}
else if (flagSend == 3) {
flagSend = 4;
}
else if (flagSend == 4) {
del = Firebase.getString(fbdo, F("/delete")) ? fbdo.to<const char
*>() : fbdo.errorReason().c_str();
flagSend = 0;
}
Serial.println("flag_send: " + String(flagSend));
if (del == "1") {
for (int x = 0; x < id; x++) {
deleteFingerprint(x);
digitalWrite(2, HIGH);
delay(150);
digitalWrite(2, LOW);
delay(150);
}
writeFile(SPIFFS, "/id.txt", "0");

```

```

Firestore.setString(fbdo, F("/delete"), "0");
Firestore.setString(fbdo, F("/id"), String(0));
}
if (kontakInt == 1 or flag_on == 1) {
digitalWrite(R_kontak, HIGH);
}
else if (kontakInt == 0 and flag_on == 0) {
digitalWrite(R_kontak, LOW);
}
if (StarterInt == 1 or flag_on == 2) {
x++;
if (x > 5)
{
digitalWrite(R_Starter, LOW);
digitalWrite(R_kontak, HIGH);
flag_on = 3;
Firestore.setString(fbdo, F("/Starter"), "0");
}
else {
digitalWrite(R_kontak, HIGH);
digitalWrite(R_Starter, HIGH);
}
}
Else
{
digitalWrite(R_Starter, LOW);
x = 0;
}
if (flag_on == 4) {
flag_on = 0;
digitalWrite(R_kontak, LOW);
digitalWrite(R_Starter, LOW);
}
if (AlarmInt == 1) {
digitalWrite(R_Alarm, HIGH);
}
else if (AlarmInt == 0) {
state = 0;
en_Alarm = 0;
}

```

```

    digitalWrite(R_Alarm, LOW);
}
Serial.println("del=" + del);
Serial.println("kontak=" + String(kontakInt));
Serial.println("Starter=" + String(StarterInt));
Serial.println("Alarm=" + String(AlarmInt));
Serial.println("timer=" + String(x));
}
//program untuk penampilan latitude longitude pada aplikasi//
void read_gps() {
while (SerialGPS.available()) {
if (gps.encode(SerialGPS.read()))
{
if (gps.location.isValid())
{
Latitude = gps.location.lat();
lat_str = String(Latitude , 6);
Longitude = gps.location.lng();
lng_str = String(Longitude , 6);
Firebase.setString(fbdo, F("/Latitude"), String(lat_str));
Firebase.setString(fbdo, F("/Longitude"), String(lng_str));
Serial.print("Latitude = ");
Serial.println(lat_str);
Serial.print("Longitude = ");
Serial.println(lng_str);
}
}
Serial.println();
}
}
}
//program pembacaan sensor sidikjari//
void setup_Fingerprint() {
finger.begin(57600);
delay(5);
if (finger.verifyPassword()) {
Serial.println("Found Fingerprint sensor!");
}
else {
Serial.println("Did not find Fingerprint sensor :(");
}
}
}

```



```

}
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);
finger.getTemplateCount();
if (finger.templateCount == 0) {
Serial.print("Sensor doesn't contain any Fingerprint data. Please run
the 'enroll' example.");
}
else {
Serial.println("Waiting for valid finger...");
Serial.print("Sensor contains "); Serial.print(finger.templateCount);
Serial.println(" templates");
}
}
uint8_t getFingerprintID() {
uint8_t p = finger.getImage();
switch (p) {
case FINGERPRINT_OK:
digitalWrite(2, HIGH);
en_add = 1;
Serial.println("en_add: " + String(en_add));
Serial.println("Image taken");
break;
case FINGERPRINT_NOFINGER:
en_add = 0;
en_Fingerprint = 0;
Serial.println("No finger detected");
return p;
case FINGERPRINT_PACKETRECEIVEERR:
en_add = 0;

```

```

Serial.println("Communication error");
return p;
case FINGERPRINT_IMAGEFAIL:
en_add = 0;
Serial.println("Imaging error");
return p;
default:
en_add = 0;
Serial.println("Unknown error");
return p;
}
// OK success!
p = finger.image2Tz();
switch (p) {
case FINGERPRINT_OK:
Serial.println("Image converted");
break;
case FINGERPRINT_IMAGEMESS:
Serial.println("Image too messy");
return p;
case FINGERPRINT_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!
p = finger.fingerSearch();
if (p == FINGERPRINT_OK) {
en_Fingerprint = 1;
en_Alarm = 0;
Firestore.setString(fbdo, F("/id"), String(finger.fingerID));

```

```

Serial.println("Found a print match!");
} else if (p == FINGERPRINT_PACKETRECEIVED) {
Serial.println("Communication error");
return p;
} else if (p == FINGERPRINT_NOTFOUND) {
Firestore.setString(fbdo, F("/Alarm"), String(1));
en_Fingerprint = 0;
en_Alarm = 1;
Serial.println("Did not find a match");
return p;
}
else
{
Serial.println("Unknown error");
return p;
}
// found a match!
Serial.print("Found ID #"); Serial.print(finger.fingerID);
Serial.print(" with confidence of "); Serial.println(finger.confidence);
return finger.fingerID;
}
// returns -1 if failed, otherwise returns 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;
// found a match!
Serial.print("Found ID #"); Serial.print(finger.fingerID);
Serial.print(" with confidence of "); Serial.println(finger.confidence);
return finger.fingerID;
}
String readFile(fs::FS &fs, const char * path) {
File file = fs.open(path, "r");
if (!file || file.isDirectory())
{
return String();
}
}

```

```

}
String fileContent;
while (file.available())
{
fileContent += String((char)file.read());
}
return fileContent;
}
void writeFile(fs::FS &fs, const char * path, const char * message)
{
File file = fs.open(path, "w");
if (!file)
{
return;
}
file.print(message);
}
//program hapus sidik jari//
// deleteFingerprint(id);
uint8_t deleteFingerprint(uint8_t id) {
uint8_t p = -1;
p = finger.deleteModel(id);
if (p == FINGERPRINT_OK) {
Serial.println("Deleted!");
} else if (p == FINGERPRINT_PACKETRECEIVEERR) {
Serial.println("Communication error");
} else if (p == FINGERPRINT_BADLOCATION) {
Serial.println("Could not delete in that location");
} else if (p == FINGERPRINT_FLASHERR) {
Serial.println("Error writing to flash");
} else {
Serial.print("Unknown error: 0x"); Serial.println(p, HEX);
}
return p;
}

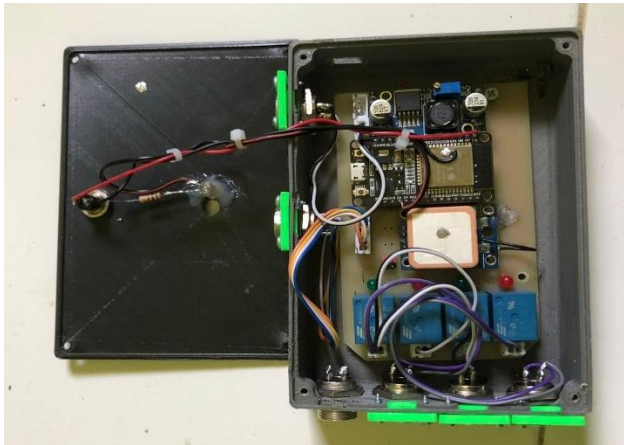
```

LAMPIRAN B

Dokumentasi Hasil Alat



Gambar (a) Alat tampak bagian luar



Gambar (b) Alat tampak bagian dalam



Gambar (c) Pemasangan alat pada sepeda motor



Gambar (d) Pemasangan sensor sidik jari pada sepeda motor

BIODATA PENULIS



- Nama : Bachtiar Rizki Andriyanto
- Tempat/Tanggal Lahir : Cilacap, 22 Januari 2001
- Alamat : Jl. Kyai safari RT 01 / RW 02, Desa Padangsari, Kecamatan Majenang, Kab.Cilacap, Jawa Tengah.
- Email : bachtiarrizkiandriyanto@gmail.com
- Telephone/Hp. : 085708539959
- Hobi : Meratapi Nasib.
- Motto : Tetap bersyukur.
- Riwayat Pendidikan :
1. MI Elbayan Padangsari Tahun 2007 – 2013
 2. SMP Negeri 3 Majenang Tahun 2013 – 2016
 3. MAN 2 Cilacap Tahun 2016 – 2019
 4. Politeknik Negeri Cilacap Tahun 2019 – 2022
Prodi D3 – Teknik Elektronika

Penulis telah mengikuti seminar Tugas Akhir pada tanggal 23 Agustus 2022 sebagai salah satu persyaratan untuk memperoleh gelar Ahli Madya (A.Md).