

LAMPIRAN

1. *Source Code* Arduino
2. *Source Code* Python
3. Hasil pengecekan *Turnitin*
4. Lembar Revisi



SOURCE CODE RANCANG BANGUN SMART DOOR LOCK
MENGGUNAKAN KOMBINASI SUARA DAN FINGERPRINT (STUDI
KASUS: PT TYOTECH MANDIRI JAYA)

Source Code Arduino

```
#include <WiFi.h>                #define  BLYNK_AUTH_TOKEN
#include <ArduinoJson.h>          "a54M3HVTAnRmi0EVuVx0b1m4
#include                          tqQfkM14"
<WiFiClientSecure.h>
#include                          unsigned          long
<UniversalTelegramBot.h>        lastAccessTime = 0;
#include                          unsigned          long
<Adafruit_Fingerprint.h>        doorOpenTime = 0;
#include                          int doorOpenCount = 0;
<HardwareSerial.h>              int data_terbuka = 1;
#include <ESP32Servo.h>           int data_tertutup = 0;
#include                          String pintu1_terbuka =
<BlynkSimpleEsp32.h>            "Pintu 1 terbuka";
                                  String pintu1_tertutup =
#define BLYNK_TEMPLATE_ID        "Pintu 1 tertutup";
"TMPL6ZNNnTUHT"                 String pintu2_terbuka =
#define                          "Pintu 2 terbuka";
BLYNK_TEMPLATE_NAME              String pintu2_tertutup =
"Smart Door Lock"                "Pintu 2 tertutup";
```

```

String kata_kunci = "buka           SoftwareSerial
pintu";           mySerial(22, 23);

String  notif_bahaya  =  #else
"Waspada!  Ada  upaya  #define mySerial Serial1
pendobrakan";        #endif

const int servo1Pin = 19;  Adafruit_Fingerprint
const int servo2Pin = 18;  finger  =
const int buzzerPin = 5;  Adafruit_Fingerprint(&Ser
const int voiceSensorPin  ial2);
= 27;

const int infraredPin1 =  uint8_t id = 0;
26;

const int infraredPin2 =  char ssid[] = "A52s"; //
14;  Menggunakan hotspot dari
const int infraredPin3 =  hp
25;  char  password[]  =
int batas_akses = 0;  "Betari1234"; // Password
hotspot

Servo myServo1, myServo2;  String  BOTtoken  =
"6156893507:AAGQL0IC0JHqA
#if  (defined(AVR)  ||  vNWSv15reeD7zRXR1bm-DQ";
defined(ESP8266))  &&  String  chatid  =
!defined(AVR_ATmega2560)  "1393545724";
WiFiClientSecure client;

```

```

UniversalTelegramBot      pinMode (infraredPin2,
bot (BOTtoken, client);   INPUT);

                           pinMode (infraredPin3,
void setup()              INPUT);
{
  Serial.begin(9600);      WiFi.begin(ssid,
                           while (!Serial); // For password);
Yun/Leo/Micro/Zero/...   while (WiFi.status() !=
                           WL_CONNECTED) {
Serial.println("\n\nAdafr   delay(500);
uit finger detect test");
                           Serial.println("Connectin
myServo1.attach(servo1Pin  g to WiFi...");
);                           }
myServo2.attach(servo2Pin  Serial.println("Connected
);                           to WiFi!");
  myServo1.write(0);
  myServo2.write(0);      client.setInsecure();
  pinMode (buzzerPin,     if
OUTPUT);                   (bot.sendMessage(chatid,
  pinMode (voiceSensorPin, "Memulai Telegram")) {
INPUT);
  pinMode (infraredPin1,  Serial.println("Telegram
INPUT);

```

```

Bot          connected          Serial.println("Waiting
successfully!");                for          valid          voice
} else {                          command...");

Serial.println("Telegram          Blynk.begin(BLYNK_AUTH_TO
Bot connection failed!");        KEN, ssid, password);
}

// set the data rate for          void loop(){
the sensor serial port          Blynk.run();
finger.begin(57600);              unsigned          long
if                                  currentTime = millis();
(finger.verifyPassword())          if          (currentTime -
{                                  lastAccessTime          >=
    Serial.println("Found          86400000) { // 86400000
fingerprint sensor!");            milidetik = 24 jam
}                                  //          Reset          hitungan
else                                pintu terbuka
{                                  doorOpenCount = 0;
    Serial.println("Did          lastAccessTime          =
not          find          fingerprint          currentTime;
sensor :(");                        }
    while (1);                      if
}                                  (detectVoiceCommand("buka
                                  pintu"))

```

```

{
    delay(5000); //sendTelegramMessage("Pintu 1 terbuka");
    if (verifyFingerprint()) lastAccessTime =
    {
        millis();
        myServo1.write(90); doorOpenCount++;
        delay(25000); //
digitalWrite(buzzerPin, HIGH); Delay untuk menjaga pintu
        delay(800); myServo1.write(0); terbuka selama 8 detik
digitalWrite(buzzerPin, LOW); Blynk.virtualWrite(V1,
pintul_tertutup);
Blynk.virtualWrite(V1, pintul_terbuka); Blynk.virtualWrite(V2,
data_tertutup);
Blynk.virtualWrite(V2, data_terbuka); Serial.println("Pintu 1
tertutup!");
Blynk.virtualWrite(V4, doorOpenCount); //sendTelegramMessage("Pintu 1
tertutup");
        delay(5000); //
Serial.println("Pintu 1 Delay sebelum mencari
terbuka!"); sidik jari berikutnya

```

```

    }
}

int infraredValue2 =
digitalRead(infraredPin2)

;

//Membatasi akses masuk if (infraredValue2 ==
int infraredValue1 = LOW) {
digitalRead(infraredPin1)

; Serial.println("\nStaff
if (infraredValue1 == terdeteksi ingin keluar
LOW) { ruangan");
if (batas_akses < 5) myServo2.write(90);
{ // Menggerakkan servo2
batas_akses++; untuk membuka pintu
Blynk.virtualWrite(V7, Blynk.virtualWrite(V3,
batas_akses); pintu2_terbuka);

Serial.println("Staff Blynk.virtualWrite(V2,
masuk. Jumlah orang: " + data_terbuka);
String(batas_akses)); Serial.println("Pintu
} 2 terbuka!");
}

//sendTelegramMessage("Pi
// Membuka dan menutup
pintu dari dalam dengan
sensor infrared

```

```

        delay(25000); //
Delay untuk memungkinkan //Blynk.virtualWrite(V8,
pintu terbuka           batas_akses);

        myServo2.write(0); //
Menggerakkan servo2 untuk Serial.println("Staff
menutup pintu setelah 8 keluar. Jumlah akses
detik                    sekarang: " +
                          String(batas_akses));

Blynk.virtualWrite(V3,   }
pintu2_tertutup);      } delay(10000);
                          //int infraredValue3 =
Blynk.virtualWrite(V2, digitalRead(infraredPin3)
data_tertutup);        ;
        Serial.println("Pintu //if (infraredValue3 ==
2 tertutup!");        LOW) {

//sendTelegramMessage("Pi //Serial.println("\nWaspa
ntu 2 tertutup");    da upaya pendobrakan
                          pintu 2");

        //pengurangan jumlah
orang                    //sendTelegramMessage("Wa
                          spada! Ada upaya
        if (batas_akses > 0)   pendobrakan pintu 2!");
{
        batas_akses--;        //}

```



```

    delay(5000); // Don't need to run this at full speed.
}

bool detectVoiceCommand(const char* command)
{
    int sensorValue = digitalRead(voiceSensorPin);

    if (sensorValue == HIGH)
    {
        Serial.write(command);
        {
            Serial.println("\nVoice command detected: " + String(command));
            return true;
        }
        delay(5000);
    }
}

bool verifyFingerprint()
{
    Serial.println("Place your finger on the fingerprint sensor...");
    delay(5000);
    int fingerprintID = getFingerprintID();
    if (fingerprintID != -1)
    {
        Serial.print("Found fingerprint ID: ");
        Serial.println(fingerprintID);
        return true;
    }
    else
    {

```

```

uint8_t p =
Blynk.virtualWrite(V6, finger.getImage());
notif_bahaya); if (p !=
FINGERPRINT_OK)
Serial.println("Fingerpri return -1;
nt verification failed!
Waspada pendobrakan!"); p = finger.image2Tz();
if (p !=
sendTelegramMessage("Wasp FINGERPRINT_OK)
ada! Ada upaya return -1;
pendobrakan!");
p =
digitalWrite(buzzerPin, finger.fingerFastSearch()
HIGH); ;
delay(1000); if (p !=
FINGERPRINT_OK)
digitalWrite(buzzerPin, return -1;
LOW);
return false; return finger.fingerID;
}
}
}
void
int getFingerprintID() sendTelegramMessage(Strin
{ g message) {

```

```

        if (WiFi.status() == WL_CONNECTED) {
            bot.sendMessage(chatid, message, "");
        }
    }
}

```

Source Code Python

- Website IoT

```

from tkinter import *
import streamlit as st
import requests
import pandas as pd
import json
#import matplotlib.pyplot as plt
import time
import streamlit_authenticator as stauth
import pickle
from pathlib import Path
from PIL import Image

st.markdown(
    """
<style>
        footer {visibility: hidden;}
        [data-testid="column"] {
            border: 1px solid #CCCCCC;
            padding: 5% 5% 5% 3%;
            border-radius: 5px;
            border-left: 0.5rem solid #9AD8E1 !important;
            box-shadow: 0 0.15rem 1.75rem 0 rgba(58, 59, 69, 0.15) !important;
        }
        [data-testid="stMarkdownContainer"]{
    """
)

```

```

font-weight: 700          names = ["Betari",
!important;              "Indri"]

text-transform:          usernames = ["betari",
uppercase !important;    "indri"]
}

# load hashed
</style>                passwords
"""                      file_path =
                        Path(__file__).parent /
st.markdown(hide_st_style "hashed_pw.pkl"
, unsafe_allow_html=True) with
                        file_path.open("rb") as
def load(data):          file:
    res = hashed_passwords
requests.get(f'https://bl = pickle.load(file)
ynk.cloud/external/api/ge credentials = {
t?token=a54M3HVTAAnRMi0EVu "usernames":{
Vx0blm4tqQfkM14&{data}') usernames[0]:{
    return res.text      "name":na
                        mes[0],

counter = 1              "password
def app():               ":hashed_passwords[0]
                        # --- USER },
AUTHENTICATION ---      usernames[1]:{

```

```

        "name":name,
        "password":password,
        ":hashed_passwords[1]":hashed_passwords[1]
    }
}

authenticator = stauth.Authenticate(credentials, "app_home", "auth", cookie_expiry_days=30)
authenticator.logout("Logout", "sidebar")
st.header("Dashboard")
st.subheader("Pinjam Kantor PT Tyotech Mandiri Jaya")
authenticator.login("Login", "main")

if authentication_status == False:
    st.error("Username/password is incorrect")

image = Image.open('pintu1terbuka.png')
image2 = Image.open('pintu2terbuka.png')

```

```

        image3 = Image.open('pintutertutup.png')
        for seconds in range(5):
            with placeholder.container():
                data = {
                    'Timestamp': st.columns(2)
                },
                'Keterangan Pintu 1': [],
                'Keterangan Pintu 2': [],
                'Akses Ruangan': [],
                'Jumlah Orang': []
            }
            col1.metr
            placeholder = ic("Keterangan Pintu 1",
            st.empty()
                if
            st.button("Reload"):
                st.experiment
            al_rerun()

```

```

data['Timestamp'].append(pd.Timestamp.now().strftime('%Y-%m-%dT%H:%M:%S.%fZ'))
df3 = pd.concat([df3, df], ignore_index=True)
df3.to_csv('iot.csv', index=False)

data['Keterangan Pintu 1'].append(str(ket_pintu_1))
st.warning("Waspadada! Ada upaya pendobrakan")

data['Keterangan Pintu 2'].append(str(ket_pintu_2))
# Mengecek apakah Keterangan Pintu 1 adalah "Buka" dan memberikan data['Akses notifikasi jika iya Ruangan'].append((akses_ruangan))
if str(ket_pintu_1) == "Pintu 1 terbuka" and str(ket_pintu_2) == "Pintu 2 tertutup":
    st.image(image, caption='Pintu 1 terbuka')

df = pd.DataFrame(data)
if str(ket_pintu_2) == "Pintu 2 terbuka" and
df3=pd.read_csv('iot.csv')

```

```

str(ket_pintu_1) == def Laporan():
"Pintu 1 tertutup": st.header("📄 Laporan
                    st.image(imag Histori 📄")
e2, caption='Pintu 2 st.subheader("Masukka
terbuka') n Rentang Waktu")
                    df2 =
                    pd.read_csv('iot.csv')
                    startdate, enddate =
                    st.columns(2)
                    with startdate:
                    start_date =
                    st.date_input(
                    "masukkan
                    # if tanggal awal",
                    pd.to_datetim
                    e(df2['Timestamp']).min()
                    st.markdown("##
                    ,
                    min_value =
                    pd.to_datetime(df2['Times
                    df2 = tamp']).min(),
                    pd.read_csv('iot.csv') max_value =
                    st.dataframe(df2, pd.to_datetime(df2['Times
                    height=300) tamp']).max().date()-
                    pd.DateOffset(days=1)

```



```

)
df2['Timestamp'] =
st.write("Tanggal pd.to_datetime(df2['Times
Awal", start_date) tamp']).dt.date
with enddate: #Timestamp =
end_date = pd.to_datetime(df2['Times
st.date_input( tamp'], format='%Y-%m-
"masukkan %dT%H:%M:%S.%fZ',
tanggal akhir", utc=True)
pd.to_datetim filtered_df =
e(df2['Timestamp']).min() df2.loc[(df2['Timestamp']
.date()+pd.DateOffset(day >= start_date) &
s=1), (df2['Timestamp'] <=
min_value = end_date)]
pd.to_datetime(df2['Times st.dataframe(filtered
tamp']).min().date()+pd.D _df, height=300)
ateOffset(days=1),
max_value = app_mode =
pd.to_datetime(df2['Times st.sidebar.selectbox('MEN
tamp']).max().date()- U', ['Dashboard',
pd.DateOffset(days=1) 'Tentang Sistem',
) 'Laporan'])
st.write("Tanggal
Akhir", end_date) if app_mode ==
'Dashboard':
# while True:

```

```

app()
# time.sleep(1)
elif app_mode == 'Tentang
Sistem':
    st.header('Rancang
Bangun Sistem Pembukaan
Pintu Hemat Daya
Menggunakan Kombinasi
Suara dan Fingerprint
Serta Kontrol Bot
Telegram Untuk Kantor PT
Tyotech Mandiri Jaya')
    st.subheader('Tentang
Sistem')
long_paragraph = """
Sistem dibuat dengan
berbasis Internet of
Things yang artinya
sistem berjalan dengan
adanya koneksi internet
dan dapat dimonitor
menggunakan aplikasi
"""
st.write(long_paragra
ph)
elif app_mode ==
'Laporan':
    Laporan()

```

- **Autentikasi Login**

```
import pickle

from pathlib import Path

hashed_passwords =
stauth.Hasher(passwords).
generate()

import streamlit_authenticator
as stauth

file_path =
Path(__file__).parent /

names = ["Betari", "Indri"]
"hashed_pw.pkl"
with file_path.open("wb")
as file:
usernames = ["betari", "indri"]
passwords = ["1234", "5678"]
stauth.pickle.dump(hashed_passwords, file)
```

BETARI INDRIANING SUGIARTO_RANCANG BANGUN SMART DOOR LOCK MENGGUNAKAN KOMBINASI SUARA DAN FINGERPRINT (STUDI KASUS: KANTOR PT TYOTECH MANDIRI JAYA)

ORIGINALITY REPORT

16%

SIMILARITY INDEX

16%

INTERNET SOURCES

8%

PUBLICATIONS

2%

STUDENT PAPERS

PRIMARY SOURCES

1	jurusan.tik.pnj.ac.id Internet Source	2%
2	docplayer.info Internet Source	1%
3	digilib.unila.ac.id Internet Source	1%
4	jemit.fmipa.unila.ac.id Internet Source	1%
5	123dok.com Internet Source	<1%
6	ijcs.stmikindonesia.ac.id Internet Source	<1%
7	fr.slideshare.net Internet Source	<1%
8	repository.uin-suska.ac.id Internet Source	<1%



9	id.123dok.com Internet Source	<1 %
10	repo.darmajaya.ac.id Internet Source	<1 %
11	eprints.unisnu.ac.id Internet Source	<1 %
12	murni_rk.staff.gunadarma.ac.id Internet Source	<1 %
13	adoc.pub Internet Source	<1 %
14	journal.uta45jakarta.ac.id Internet Source	<1 %
15	repository.atmaluhur.ac.id Internet Source	<1 %
16	eprints.polsri.ac.id Internet Source	<1 %
17	eprints.radenfatah.ac.id Internet Source	<1 %
18	id.scribd.com Internet Source	<1 %
19	repository.umsu.ac.id Internet Source	<1 %
20	mafiadoc.com Internet Source	<1 %

21	lib.unnes.ac.id Internet Source	<1 %
22	repository.uisu.ac.id Internet Source	<1 %
23	repository.ubharajaya.ac.id Internet Source	<1 %
24	uge-one.com Internet Source	<1 %
25	zh.scribd.com Internet Source	<1 %
26	core.ac.uk Internet Source	<1 %
27	repository.unwira.ac.id Internet Source	<1 %
28	pt.scribd.com Internet Source	<1 %
29	www.slideshare.net Internet Source	<1 %
30	Hani Atun Mumtahana, Sekreningsih Nita, Adzinta Winerawan Tito. "Pemanfaatan Web E-Commerce untuk Meningkatkan Strategi Pemasaran", Khazanah Informatika : Jurnal Ilmu Komputer dan Informatika, 2017 Publication	<1 %

lapangantekno.blogspot.com

31

Internet Source

<1 %

32

library.palcomtech.com

Internet Source

<1 %

33

repo.unand.ac.id

Internet Source

<1 %

34

Hari Aspriyono. "Implementasi Metode Waterfall Dalam Pembuatan E-Learning Pada SMK Teknik PAL Surabaya Menggunakan Codeigniter Dan MySQL", SIMKOM, 2021

Publication

<1 %

35

Saladin Wirawan, Ita Ita. "Implementasi IoT dalam Manajemen Rantai Pasok Distribusi Elpiji pada Agen Gas 3 Kg di Kota Palembang", remik, 2023

Publication

<1 %

36

elib.pnc.ac.id

Internet Source

<1 %

37

garuda.kemdikbud.go.id

Internet Source

<1 %

38

norisahrunedukasi.wordpress.com

Internet Source

<1 %

39

repository.bsi.ac.id

Internet Source

<1 %

40

repository.dinamika.ac.id

Internet Source

<1 %

41

docobook.com

Internet Source

<1 %

42

text-id.123dok.com

Internet Source

<1 %

43

widuri.raharja.info

Internet Source

<1 %

44

Darmanta Sukrianto. "PEMANFAATAN TEKNOLOGI SMS GATEWAY SISTEM ADMINISTRASI KENDARAAN BERMOTOR PADA PT. RIAU ARGO PERKASA", JURNAL FASILKOM, 2020

Publication

<1 %

45

community.amd.com

Internet Source

<1 %

46

jurnal.fkip.uns.ac.id

Internet Source

<1 %

47

kc.umn.ac.id

Internet Source

<1 %

48

ojs.stmikpringsewu.ac.id

Internet Source

<1 %

49

pt.slideshare.net

Internet Source

<1 %

repository.radenfatah.ac.id

50

Internet Source

<1 %

51

repository.upbatam.ac.id

Internet Source

<1 %

52

repository.usd.ac.id

Internet Source

<1 %

53

www.smart-telecom.co.id

Internet Source

<1 %

54

www.warungbelajar.com

Internet Source

<1 %

55

Soni Ayi Purnama. "PERENCANAAN ARSITEKTUR SISTEM INFORMASI PT. MA'SOEM ARIAS DENGAN MENGGUNAKAN ORACLE ENTERPRISE ARCHITECTURE FRAMEWORK", AIMS: Jurnal Accounting Information System, 2018

Publication

<1 %

56

edoc.site

Internet Source

<1 %

57

eprints.poltektegal.ac.id

Internet Source

<1 %

58

jefryrifkiefahrezy.blogspot.com

Internet Source

<1 %

59

karyailmiah.unisba.ac.id

Internet Source

<1 %

60 repository.ibs.ac.id <1 %
Internet Source

61 repository.its.ac.id <1 %
Internet Source

62 repository.uinjkt.ac.id <1 %
Internet Source

63 teknikelektronika12.wordpress.com <1 %
Internet Source

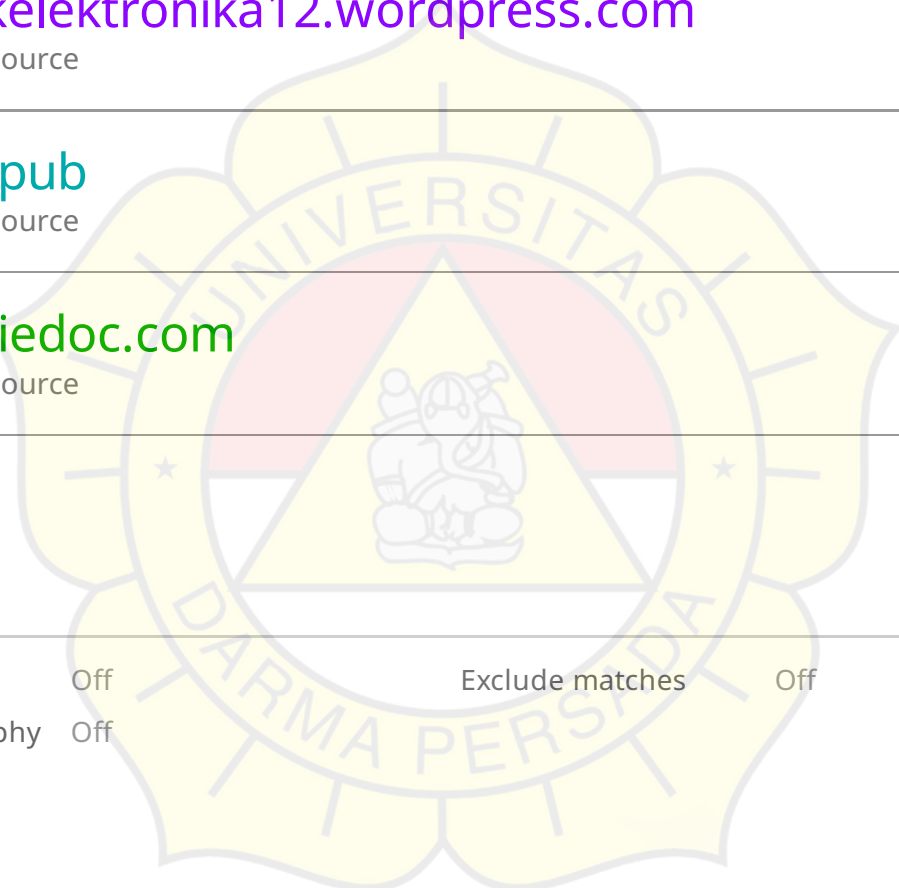
64 doku.pub <1 %
Internet Source

65 zombiedoc.com <1 %
Internet Source

Exclude quotes Off

Exclude matches Off

Exclude bibliography Off





UNIVERSITAS DARMA PERSADA

Jl. Taman Malaka Selatan, Pondok Kelapa, Jakarta Timur, Indonesia 13450

Telp. (021) 8649051, 8649053, 8649057 Fax. (021) 8649052

E-mail : humas@unsada.ac.id Home page : <http://www.unsada.ac.id>


LEMBAR REVISI - SIDANG SKRIPSI

NIM>Nama : 2019230015 Betari Indrianing Sugiarto
Fakultas/Prodi : Teknik / Teknologi Informasi

No.	Keterangan Revisi	Dosen
1.	<p>Laporan Deployment glet dibuat pada laporan</p> <p>- tambahkan ID pengguna pada web yg belum - masuk ruangan.</p>	<p>P. Si</p> <p>#</p> <p>20/2/24.</p>
2.	<p>Laporan diperbaiki: penulisan; Gambar diberi list dan kalimat untuk menjadi ke gambar tersebut.</p> <p>Rangkaian alat di beri nomor, kemudian diberi keterangannya.</p>	<p>Sruti</p> <p>20/2/24</p>

Mengetahui,

Ka Prodi Teknologi Informasi


Herianto, S.Pd., MT.