

DAFTAR PUSTAKA

- Algoritma data science school (2022) BAGAIMANA MELAKUKAN DATA COLLECTION UNTUK ANALISIS?. Available at: <https://algorit.ma/blog/data-collection-2022/> (Accessed: 16 Feb 2025).
- Anggara, H. (2019), Penerapan Rfid (Radio Frequency Identification) Pada Prototype Pintu Otomatis Dan Terintegrasi Dengan Saklar Pengendali Kipas Angin Berbasis Arduino Uno, Skripsi, Universitas Sriwijaya.
- Pradana Bisma, "Perancangan Sistem Ticketing Pariwisata Berbasis RFID di Objek Wisata Suban Air Panas Rejang Lebong" Skripsi Program Studi Ilmu Komputer, Universtas Pat Petulai
- Ferdiansyah, Prayogi, A. Rahman Sujatmika, and Izzatul Ummami. "Rancang Bangun Absensi Mahasiswa Menggunakan RFID dan Sensor DS18B20 Berbasis NodeMCU Di Universitas Darul Ulum." Jurnal Jurnal Sains Dan Teknologi (JSIT) 3.1 (2023): 158-164.
- Hartanto, Sri, and Andre Dwi Prabowo. "Rancang Bangun Sistem Absensi Dengan Pemeriksaan Suhu Tubuh Berbasis Arduino ATmega2560." Jurnal Elektrokrisna 9.3 (2021): 27-40.
- Huda, Qomarul, Hairul Fahmi, and Ahmad S. Pardiansyah. "Prototype Sistem Absensi Menggunakan Rfid Berbasis Iot." Jurnal Publikasi Ilmu Komputer dan Multimedia 1.1 (2022): 87-90.
- Kumala, Ardelia Ruciragati. "Prototipe Sistem Rekam Data Alat-alat Ukur, Takar, Timbang dan Perlengkapannya Berbasis RFID." Telekontran: Jurnal Ilmiah Telekomunikasi, Kendali dan Elektronika Terapan 12.1 (2024): 38-49.
- Muliadi, M., Imran, A., & Rasul, M. (2020). Pengembangan tempat sampah pintar menggunakan ESP32. Jurnal Media Elektrik, 17(2), 73-79.
- Nizam, M. N., Yuana, H., & Wulansari, Z. (2022). Mikrokontroler Esp 32 Sebagai Alat Monitoring Pintu Berbasis Web. JATI (Jurnal Mahasiswa Teknik Informatika), 6(2), 767-772.
- Nugroho, A. (2021). Rancang Bangun Sistem Keamanan Mobil Rental Menggunakan Sensor Sidik Jari Berbasis Arduino Dengan Kendali Aplikasi Telegram (Doctoral dissertation, Universitas Komputer Indonesia).
- Pradana, V., & Wiharto, H. L. (2020). Rancang Bangun Smart Locker Menggunakan Rfid Berbasis Arduino Uno. Jurnal EL Sains P-ISSN, 2527, 6336.

- Ryan, Hidayat, Limpraptono F Yudi, and Ardita Michael. "Rancang Bangun Alat Absensi Karyawan Menggunakan RFID dan ESP32Cam Berbasis Internet of Things." (2022): 137-145.
- Samsugi, S., Mardiyansyah, Z., & Nurkholis, A. (2020). Sistem Pengontrol Irigasi Otomatis Menggunakan Mikrokontroler Arduino UNO. Jurnal Teknologi Dan Sistem Tertanam, 1(1), 17-22.
- Taufiq, Imam (2018). "Deep *Learning* Untuk Deteksi Tanda Nomor Kendaraan Bermotor Menggunakan Algoritma *Convolutional Neural Network* Dengan Python Dan Tensorflow". Skripsi. Program Studi Sistem Informasi Sekolah Tinggi Manajemen Informatika dan Komputer AKAKOM.
- Tyas, U. M., & Buckhari, A. A. (2023). IMPLEMENTASI APLIKASI ARDUINO IDE PADA MATA KULIAH SISTEM DIGITAL. TEKNOS: Jurnal Pendidikan dan Teknologi, 1(1), 1-9.

LAMPIRAN

Program di ESP 32

```
#include <SPI.h>
#include <MFRC522.h>
#include <UrlEncode.h>
#include <LiquidCrystal_I2C.h>
#include <ArduinoJson.h>
#include "Arduino.h"
#include "DFRobotDFPlayerMini.h"
#include <WiFi.h>
#include <HTTPClient.h>

#define SS_PIN 5
#define RST_PIN 27
int lcdColumns = 16;
int lcdRows = 2;
String TAG;
String MobileNumber = "";
String APIKey = "";
bool absensiAktif = false;
LiquidCrystal_I2C lcd(0x27, lcdColumns, lcdRows);
MFRC522 mfrc522(SS_PIN, RST_PIN);
DFRobotDFPlayerMini mp3;

const char* ssid = ".";
const char* password = "00000000";

void setup() {
  Serial.begin(115200);
  Serial2.begin(9600);
  lcd.begin();
  lcd.backlight();
  SPI.begin();
  mfrc522.PCD_Init();
  WiFi.begin(ssid, password);

  Serial.println();
  Serial.println(F("DFRobot DFPlayer Mini Demo"));
  Serial.println(F("Initializing DFPlayer ... (May take 3~5 seconds)"));

  if (!mp3.begin(Serial2)) {
    Serial.println(F("Unable to begin:"));
  }
}
```

```

        Serial.println(F("1.Please recheck the connection!"));
        Serial.println(F("2.Please insert the SD card!"));
        while (true);
    }
    Serial.println(F("DFPlayer Mini online."));
    mp3.setTimeout(500);
    mp3.volume(50);
    delay(500);

    while (WiFi.status() != WL_CONNECTED) {
        mp3.play (1);
        delay(1000);
        Serial.println("Menghubungkan..");
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("menghubungkan");
        lcd.setCursor(0, 1);
        lcd.print("ke jaringan");
    }
    delay(1000);
        mp3.play (2);
        Serial.print("Use this URL to connect: ");
        Serial.print("http://");
        Serial.print(WiFi.localIP());
        Serial.println("");
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("Terhubung");
        lcd.setCursor(0, 1);
        lcd.print("ke jaringan");
        delay(1000);
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print(" SISTEM ABSENSI");
        lcd.setCursor(0, 1);
        lcd.print("UMB BELUM DIBUKA");
        delay(100);
    }
    String getTAG() {
        if (!mfr522.PICC_IsNewCardPresent()) return "";
        if (!mfr522.PICC_ReadCardSerial()) return "";

        String tag = "";
        for (byte i = 0; i < mfr522.uid.size; i++) {
            tag += String(mfr522.uid.uidByte[i], HEX);

```

```

    }

    mfrc522.PICC_HaltA();
    mfrc522.PCD_StopCrypto1();
    tag.toUpperCase();
    return tag;
}

void loop() {
    if (WiFi.status() == WL_CONNECTED) {
        TAG = getTAG();
        if (TAG != "") {

            if (TAG == "EB156251" || TAG == "F325ELF")
                if (!absensiAktif) {
                    mp3.play (4);
                    Serial.println("Absensi dimulai! Silakan scan kartu.");
                    lcd.clear();
                    lcd.setCursor(0, 0);
                    lcd.print(" SISTEM ABSENSI");
                    lcd.setCursor(0, 1);
                    lcd.print(" TELAH DIBUKA");
                    absensiAktif = true;
                } else {
                    mp3.play (3);
                    Serial.println("Absensi selesai! Sistem ditutup.");
                    lcd.clear();
                    lcd.setCursor(0, 0);
                    lcd.print(" SISTEM ABSENSI");
                    lcd.setCursor(0, 1);
                    lcd.print(" TELAH DITUTUP");
                    absensiAktif = false;
                    delay(700);
                    lcd.clear();
                    lcd.setCursor(0, 0);
                    lcd.print(" SISTEM ABSENSI");
                    lcd.setCursor(0, 1);
                    lcd.print("UMB BELUM DIBUKA");
                    delay(100);
                }

            }

            else if (absensiAktif) {
                Serial.println("-----");
                Serial.print("Absensi dicatat untuk ID: ");
            }
        }
    }
}

```

```

        Serial.println(TAG);
        parseJson();
    }
    else {
        Serial.println("Sistem belum dibuka. Scan TAG A untuk
memulai.");
        delay(1000);
        lcd.clear();
        delay(100);
        lcd.setCursor(0, 0);
        lcd.print(" SISTEM ABSENSI");
        lcd.setCursor(0, 1);
        lcd.print("UMB BELUM DIBUKA");
        delay(100);
    }

    TAG = "";
}
delay(1000);
}
else {
    Serial.println("Error in WiFi connection");
}
}

void parseJson() {
    HTTPClient http;
    http.begin("http://192.168.1.18/absensi/proses.php");
    http.addHeader("Content-Type", "application/x-www-form-
urlencoded");
    int httpResponseCode = http.POST("uid=" + TAG);
    delay(100);
    if (httpResponseCode > 0) {
        String response = http.getString();
        char json[500];
        response.toCharArray(json, 500);
        StaticJsonDocument<200>doc;
        deserializeJson(doc, json);

        const char* status = doc["Detail"]["Status"];
        const char* nama = doc["Detail"]["Data User"]["Nama"];
        const char* jr = doc["Detail"]["Data User"]["Jurusan"];
        const char* npm = doc["Detail"]["Data User"]["NPM"];
        APIKey = doc["Detail"]["Data User"]["APIkey"].as<String>();
    }
}

```

```
MobileNumber = doc["Detail"]["Data User"]["no  
hp"].as<String>();
```

```
Serial.print("status = ");  
Serial.println(status);  
Serial.print("Nama = ");  
Serial.println(nama);  
Serial.print("NPM = ");  
Serial.println(npm);  
Serial.print("Jurusan = ");  
Serial.println(jr);  
Serial.print("no hp = ");  
Serial.print(MobileNumber);  
Serial.print("  APIkey = ");  
Serial.println(APIKey);  
lcd.clear();  
lcd.setCursor(0, 0);  
lcd.print(status);  
delay(1000);  
lcd.clear();  
lcd.setCursor(0, 0);  
lcd.print(nama);  
lcd.setCursor(0, 1);  
lcd.print(npm);  
    if(String(status) == "absensi Sukses") {  
        mp3.play (5);  
    }  
    delay(1000);  
    if(String(status) == "ID Tidak Valid") {  
        mp3.play (6);  
    }  
    delay(1000);  
    sendMessage("absensi Sukses");  
  
} else {  
    Serial.print("Error on sending POST: ");  
    Serial.println(httpResponseCode);  
    mp3.play (6);  
    delay(1000);  
}  
delay(2000);  
http.end();  
}  
  
void sendMessage(String message) {
```

```

    String url = "https://api.callmebot.com/whatsapp.php?phone=" +
MobileNumber + "&apikey=" + APIKey + "&text=" +
urlencode(message);
    HTTPClient http;
    http.begin(url);

    http.addHeader("Content-Type", "application/x-www-form-
urlencoded");

    int httpResponseCode = http.POST(url);
    if (httpResponseCode == 200) {
        Serial.println("Message sent successfully");
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("pesan WA terkirim");
        delay(1000);
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print(" SISTEM ABSENSI");
        lcd.setCursor(0, 1);
        lcd.print(" TELAH DIBUKA");
    }
    else {
        Serial.println("-----");
        Serial.print("no hp =");
        Serial.print(MobileNumber);
        Serial.print(" APIkey = ");
        Serial.println(APIKey);
        Serial.println("Error sending the message");
        Serial.print("HTTP response code: ");
        Serial.println(httpResponseCode);
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("pesan WA gagal");
        delay(1000);
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print(" SISTEM ABSENSI");
        lcd.setCursor(0, 1);
        lcd.print(" TELAH DIBUKA");
    }
    http.end();
}

```

Program connection.php

```
<?php
$dbhost = 'localhost';
$dbuser = 'root';
$password = '';
$dbname = 'absensi';

$dbconnect = new mysqli($dbhost, $dbuser, $password, $dbname);

if ($dbconnect->connect_error) {
    die('Server Error');
}
```

Program proses.php

```
<?php

if (isset($_POST['uid'])) {
    include 'connection.php';
    $id = $_POST['uid'];
    $query = mysqli_query($dbconnect, "SELECT * FROM tb_user WHERE id='$id'");
    $cek = mysqli_num_rows($query);
    if ($cek > 0) {
        mysqli_query($dbconnect, "INSERT INTO tb_masuk (id, Nama, NPM, Jurusan)
        SELECT id, Nama, NPM, Jurusan
        FROM tb_user
        WHERE id='$id'");
        $status = "Absensi Sukses";
    } else {
        $status = "Absensi gagal";
        $hasil_saldo = $data['saldo'];
    }
}

else {
    $status = "ID Belum Terdaftar";
    $data = "-";
}

$output =
```

```
[
    "Detail" =>
    [
        "Status" => $status,
        "Data User" => $data,
    ]
];
$json = json_encode($output);
echo $json;
}
```