LAMPIRAN

SCRIPT ARDUINO IDE

```
#include <HX711.h>
#include <Arduino.h>
#include <WiFi.h>
#include <FirebaseESP32.h>

#include "addons/TokenHelper.h"
#include "addons/RTDBHelper.h"

#define DEVICE_UID "4X"
#define WIFI_SSID "AF101"
#define WIFI_PASSWORD
"Des@0705"

#define API_KEY
"AIzaSyBOAORj6EwvYwXWWA4
H6zeZ02CmppM0Pt4"
#define DATABASE_URL
"https://tabung-gas-67-default-
rtdb.firebaseio.com/"

short measurements[30];
short measureIndex = 0;
unsigned long lastMeasurement = 0;
unsigned long timeFinishedSetup =
0;

String kapasitas = "Kapasitas Gas
Elpiji";

FirebaseData fbdo;
FirebaseAuth auth;
FirebaseConfig config;

String databasePath = "";
String fuid = "";
unsigned long elapsedMillis = 0;
unsigned long update_interval =
5000;
int count = 0;
bool isAuthenticated = false;

const int LOADCELL_DOUT_PIN
= 4;
```

```
const int LOADCELL_SCK_PIN  =
0;

HX711 scale;

float berat;
float berat_isi;
float kalibrasi = 210;
int n_max = 8000;
int n_min = 5000;

int persen;

FirebaseJson persen_json;

void Wifi_Init() {
  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();
}

void firebase_init() {
    config.api_key = API_KEY;
    config.database_url =
DATABASE_URL;

    Firebase.reconnectWiFi(true);

    Serial.println("----------------------
------------");
    Serial.println("Sign up new
user...");

     if (Firebase.signUp(&config,
&auth, "", ""))
    {
```

```
    Serial.println("Success");
    isAuthenticated = true;
    databasePath = "/" + kapasitas;
    fuid = auth.token.uid.c_str();
  }
  else
  {
    Serial.printf("Failed, %s\n",
config.signer.signupError.message.c
_str());
    isAuthenticated = false;
  }

  config.token_status_callback =
tokenStatusCallback;
  Firebase.begin(&config, &auth);
}

void setup() {
 Serial.begin(115200);

scale.begin(LOADCELL_DOUT_PI
N, LOADCELL_SCK_PIN);
 Wifi_Init();
 firebase_init();
}

void BacaDataSensor(){
 Serial.println("Pengambilan Data
Sensor ...");
 if (scale.is_ready())
 {
   scale.set_scale(kalibrasi);
   berat = (scale.get_units(4));
   berat = constrain(berat, n_min,
n_max);
   Serial.printf("Berat Tabung Gas
Elpiji: %.2f \n", berat);
   berat_isi = berat - n_min;
   Serial.printf("Berat Isi Tabung
Gas Elpiji: %.2f \n", berat_isi);
   persen = (berat_isi / (n_max -
n_min)) * 100;
   persen_json.set("Persen",
persen);
   persen_json.set("Pesan", "1");
```

```
    persen_json.add("Persen",
persen);
    persen_json.add("Pesan", "1");
    String jsonStr1;
    persen_json.toString(jsonStr1,
true);
  }
  else
  {
    Serial.println("HX711 not
found.");
    return;
  }
   delay(1000);}

void KirimDataSensor() {
  if (millis() - elapsedMillis >
update_interval && isAuthenticated
&& Firebase.ready())
   {
    elapsedMillis = millis();
    BacaDataSensor();
    String persen_node =
databasePath;
    if (Firebase.setJSON(fbdo,
persen_node.c_str(), persen_json))
    {
          printResult(fbdo);
        }
    else
    {
      Serial.println("FAILED");
      Serial.println("REASON: " +
fbdo.errorReason());
      Serial.println("----------------
----------------");
      Serial.println();
    }
   }
}

void loop() {
  KirimDataSensor();
}
```

BLOCKS PROGRAMING KODULAR



