

LAMPIRAN

Program Arduino IDE

```

#include "FirebaseESP8266.h" // Install Firebase ESP8266 library

#include <ESP8266WiFi.h>

#include <Servo.h>

#include "HX711.h"

#include <Wire.h> // Comes with Arduino IDE

#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);

#define DOUT 14

#define CLK 12

#define FIREBASE_HOST "https://pakankucingotomatis-f577c-default-
rtbd.firebaseio.com/" //Without http:// or https:// schemes

#define FIREBASE_AUTH "FZLGxt86TVcWieNvj7kUPVQzQLVUCXsjKoP1eIx8"

#define WIFI_SSID "mau bgt lo"

#define WIFI_PASSWORD "galuhcangtip"

//-----

HX711 scale;

float calibration_factor = -449.80; //Hasil Kalibrasi

int units;

// motor servo

Servo mekanik;

//-----Ultrasonik;

const int echoPin = 13;

```

```
const int trigPin = 15;

//float jarak;

int duration, distance;

//Define FirebaseESP8266 data object

FirebaseData firebaseData;

FirebaseData ledData;

FirebaseData led1Data;

FirebaseJson json;

void setup () {

  Serial.begin(9600);

  lcd.begin(16,2);

  lcd.init();

  // Nyalakan backlight

  lcd.backlight();

  pinMode(trigPin, OUTPUT);

  pinMode(echoPin, INPUT);

  scale.begin(DOUT, CLK);

  scale.set_scale();

  scale.tare(); //Reset the scale to 0

  long zero_factor = scale.read_average(); //Get a baseline reading

  Serial.print("Zero factor: "); //This can be used to remove the need to tare the
scale. Useful in permanent scale projects.

  Serial.println(zero_factor);

  mekanik.attach(2);

  mekanik.write(50);
```

```

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();

Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
Firebase.reconnectWiFi(true);
}
void loop () {
  if (Firebase.setFloat(firebaseData, "/FirebaseIOT/stokpakan", distance))
  {
    Serial.println("PASSED");
    Serial.println("PATH: " + firebaseData.dataPath());
    Serial.println("TYPE: " + firebaseData.dataType());
    Serial.println("ETag: " + firebaseData.ETag());
    Serial.println("-----");
    Serial.println();
  }
}

```

```
}  
else  
{  
  Serial.println("FAILED");  
  Serial.println("REASON: " + firebaseData.errorReason());  
  Serial.println("-----");  
  Serial.println();  
}  
if (Firebase.setFloat(firebaseData, "/FirebaseIOT/kondisipakan", units ))  
{  
  Serial.println("PASSED");  
  Serial.println("PATH: " + firebaseData.dataPath());  
  Serial.println("TYPE: " + firebaseData.dataType());  
  Serial.println("ETag: " + firebaseData.ETag());  
  Serial.println("-----");  
  Serial.println();  
}  
else  
{  
  Serial.println("FAILED");  
  Serial.println("REASON: " + firebaseData.errorReason());  
  Serial.println("-----");  
  Serial.println();  
}
```

```
digitalWrite(trigPin, LOW);
delayMicroseconds(2);
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
duration = pulseIn(echoPin, HIGH);
distance = duration/29.0/2.0;
Serial.println(distance);
lcd.setCursor(0,0);
lcd.print("K.Tandon:");
lcd.setCursor(10,0);
lcd.print(distance);
lcd.setCursor(14,0);
lcd.print("CM");

if (distance >=20){

lcd.setCursor(0,1);
lcd.print("TANDON HABIS ");
}

else if (distance >=5 && distance <=20 ){

lcd.setCursor(0,1);
lcd.print("TANDON TERSEDIA");
```

```

}

scale.set_scale(calibration_factor); //Adjust to this calibration factor

Serial.print("Reading: ");
units = scale.get_units(), 1;
if (units < 0)
{
  units = 0.00;
}
Serial.print("Berat: ");
Serial.print(units);
Serial.print(" Gram");
Serial.println();
if (Firebase.getString(led1Data, "/FirebaseIOT/led1")) {
  Serial.println(led1Data.stringData());
  if (led1Data.stringData() == "1"){
    if (units < 15 ){
      mekanik.write (0);
    }
    else if (units > 200 ){
      mekanik.write (50);
    }
  }

}

else if (led1Data.stringData() == "0") {

```

```
if (Firebase.getString(ledData, "/FirebaseIOT/led")) {  
  Serial.println(ledData.stringData());  
  if (ledData.stringData() == "1"){  
    mekanik.write (0);  
  }  
  else if (ledData.stringData() == "0") {  
    mekanik.write (50);  
  }  
}  
}  
}  
}  
delay(100);  
}
```