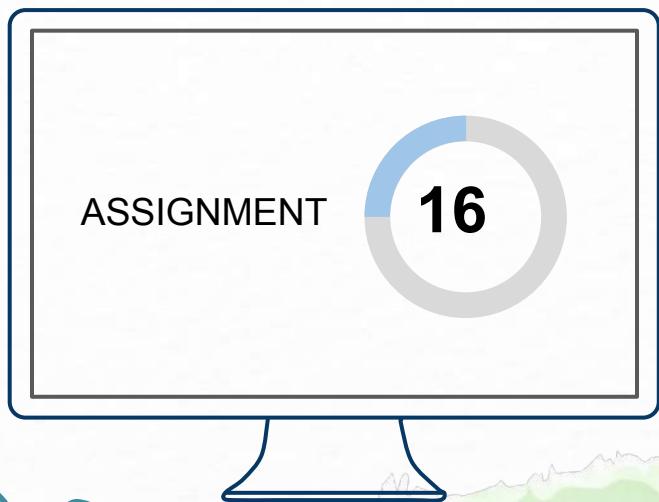


# A39 - AKBAR FADILA



## Tugas

Kirimkan semua bukti penyelesaian Mini  
Task 1-4



# Mini Task 1

## Motor Servo & Potensiometer



MySkill

MSIB

Kampus  
Merdeka

### Objektif

Atur sudut Servo berbasis Input Potensiometer

### Pra-Syarat

Memiliki akun Wokwi

#RancanganInpirasi

<https://wokwi.com/projects/360648739108411393>

wokwi.com/projects/360648739108411393

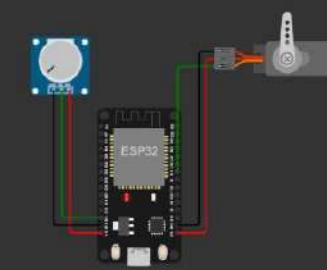
WOKWI SAVE SHARE Minitask 1.16

sketchino diagramjson libraries.txt Library Manager

Simulation 00:23.112 17%

```
1 //Atur sudut servo berbasis input potensiometer
2 #include <ESP32Servo.h> //Library Servo digunakan untuk mengontrol servo motor pada mikrokontroler ESP32.
3 Servo myservo; //Membuat objek Servo dengan nama myservo.
4 #define pinPot 13 //Mendefinisikan pin input dari potensiometer sebagai konstanta dengan nilai 13.
5
6 void setup() { //Merupakan fungsi setup() yang dieksekusi hanya sekali saat pertama kali program dijalankan.
7   serial.begin(115200); //Menginisialisasi komunikasi serial dengan baud rate 115200.
8   Serial.println("Atur sudut servo berbasis input potensiometer!"); //Menampilkan pesan pada Serial Monitor.
9   pinMode(pinPot, OUTPUT); //Menetapkan pin input potensiometer sebagai OUTPUT.
10  myservo.attach(13); //Menetapkan pin output servo motor sebagai pin 13 dan menetapkan koneksi servo motor ke pin 13.
11 }
12
13 void loop() { //Merupakan fungsi loop() yang dieksekusi berulang-ulang selama program berjalan.
14   int value = analogRead(pinPot); //Membaca nilai analog dari pin input potensiometer dan menyimpannya ke dalam variabel value.
15   value = map(value, 0, 1023, 0, 180); //Mengonversi nilai value dari rentang 0-1023 ke rentang 0-180 derajat.
16   myservo.write(value); //Mengirimkan singel PWM ke servo motor dengan nilai sudut yang diatur oleh variabel value.
17   Serial.print("Sudut: "); //Menampilkan teks "Sudut: " pada Serial Monitor.
18   Serial.println(value); //Menampilkan nilai sudut yang dikonversi pada proses sebelumnya ke serial monitor dan menunda program selama 15 miliditik sebelum kembali ke awal loop().
19
20 }
21
```

Sudut: 0  
Sudut: 0  
Sudut: 0  
Sudut: 0  
Sudut: 0  
Sudut: 0  
Sudut: 0





MySkill

MSIB

Kampus  
Merdeka

## Mini Task 2

### Motor Servo & IR Remote

#### Objektif

Menggerakan sudut servo (+ - 1°) dari tombol + - IR Remote

#### Pra-Syarat

Memiliki akun Wokwi

#RiaCkreImpian

<https://wokwi.com/projects/360693957968683009>

wokwi.com/projects/360693957968683009

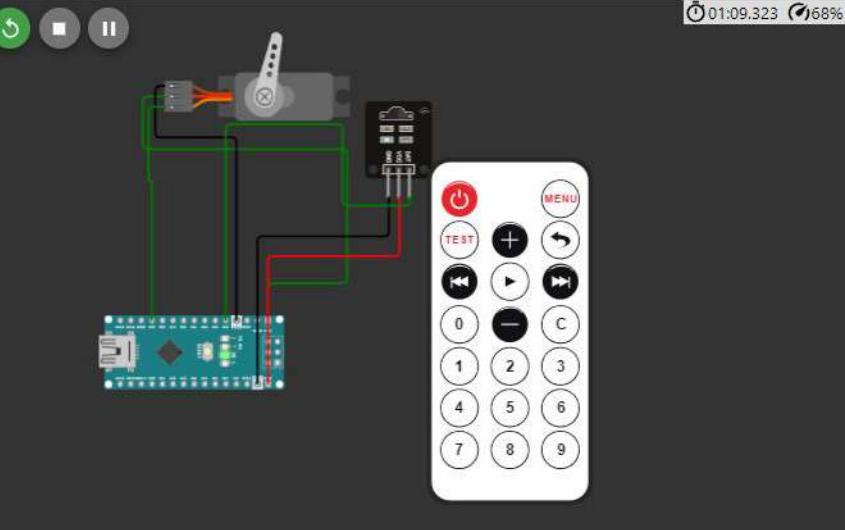
WOKWI    tutor 16 Mini task 2  

sketch.ino diagram.json libraries.txt Library Manager

```
1 //dilakukan pemanggilan library IRremote.h dan Servo.h.
2 #include <IRremote.h>
3 #include <Servo.h>
4
5 // receiver didefinisikan dengan nilai 2, yang menunjukkan bahwa input IR akan diterima melalui pin
6 int receiver = 2;
7 //objek IRrecv dininisialisasi dengan nilai receiver. Objek ini digunakan untuk menerima sinyal IR.
8 IRrecv irrecv(receiver);
9 //variabel results dininisialisasi sebagai decode_results, yang nantinya akan digunakan untuk menyimpan
10 decode_results results;
11
12 //objek Servo bernama myservo diinisialisasi.
13 Servo myservo;
14 // variabel sudut diinisialisasi dengan nilai 0.
15 int sudut = 0;
16 //yang pertama kali dijalankan ketika program dijalankan, dilakukan konfigurasi serial monitor dengan
17 void setup()
18 {
19 Serial.begin(9600);
20 irrecv.enableIRIn();
21 myservo.attach(9);
22 }
23 //fungsi loop(), program akan terus diulang-ulang selama board Arduino masih dinyalakan. Fungsi irrecv
24 void loop(){
25 if (irrecv.decode()) {
26 remote();
27 irrecv.resume();
28 Serial.println(sudut);
29 myservo.write(sudut);
30 }
31 }
32 //fungsi remote(), dilakukan pengecekan terhadap nilai IRinput yang didapat dari objek irrecv.decode
33 void remote(){
34 int IRinput = irrecv.decodedIRData.command;
35
36 if (IRinput == 2) {
```

Simulation

01:09:323 68%



1  
2  
3  
4  
5  
6  
7

# Mini Task 3

## Stepper Motor & Potensiometer



MySkill

MSIB

Kampus  
Merdeka

### Objektif

*Atur Stepper Motor berbasis Input Potensiometer*

### Pra-Syarat

*Memiliki akun Wokwi*

#PontianakImpian

<https://wokwi.com/projects/360697721205520385>

wokwi.com/projects/360697721205520385

WOKWI SAVE SHARE Docs A

sketch.ino • diagram.json • libraries.txt • Library Manager

```
1 #include <Stepper.h> //Library yang dibutuhkan untuk mengontrol stepper motor.
2 #define pinpotensio 12 //mendefinisikan pin yang digunakan untuk menghubungkan potensiometer pada pin
3
4 const int stepsPerRevolution = 200; // Mendefinisikan jumlah step per putaran pada stepper motor.
5 Stepper myStepper(stepsPerRevolution, 14, 27, 26, 29); //Mendefinisikan objek myStepper sebagai objek
6 int previous = 0; // Mendefinisikan variabel previous dengan nilai awal 0.
7 int val,hasil; //Mendefinisikan variabel val dan hasil dengan tipe data integer.
8 void setup() {
9
10    //fungsi yang dijalankan hanya sekali ketika program pertama kali dijalankan. Mendefinisikan pin pada
11    Serial.begin(115200);
12    Serial.println("Hello, ESP32!");
13    myStepper.setSpeed(60);
14    pinMode(pinpotensio, INPUT);
15 }
16
17 //Fungsi yang dijalankan berulang-ulang selama program dijalankan. Membaca nilai analog dari pin pin
18 void loop() {
19    val = analogRead(pinpotensio);
20    hasil = map(val, 0, 1023, 0, 180);
21    myStepper.step(hasil - previous);
22    previous = hasil;
23 }
```

Simulation

00:11.266 14%

Hello, ESP32!

# Mini Task 4

## Stepper Motor & IR Remote



MySkill

MSIA



### Objektif

Mengerakan stepper motor (+ - 1°) dari tombol + - IR Remote

### Pra-Syarat

Memiliki akun Wokwi

#RantikanImpian

wokwi.com/projects/360696448172293121

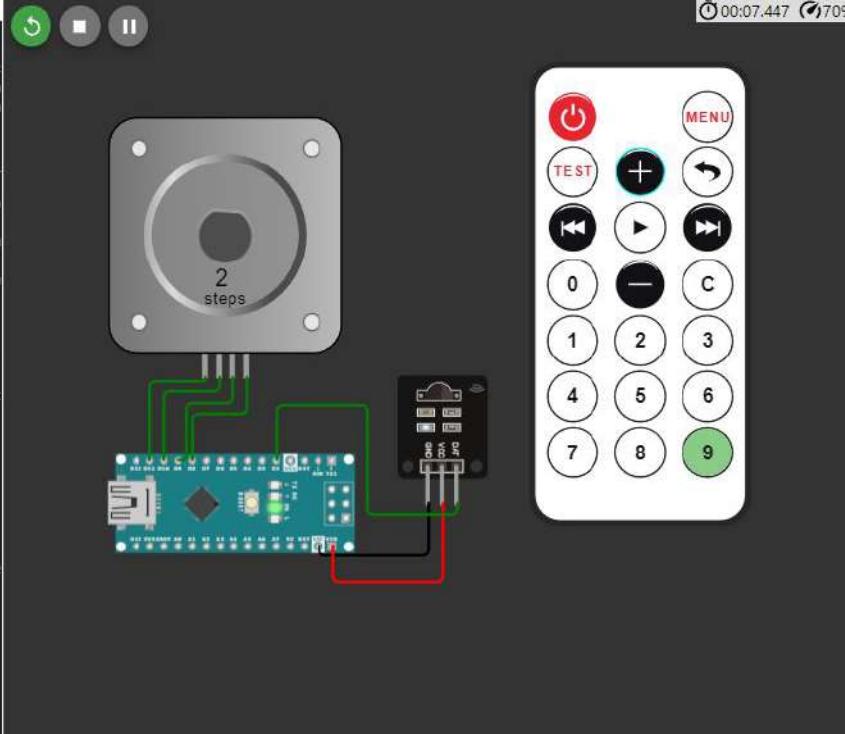
WOKWI SAVE SHARE tutor 16 , mini task 4 Docs A

sketch.ino diagram.json libraries.txt Library Manager

Simulation 00:07.447 70%

```
1 #include <IRremote.h>
2 #include <Stepper.h>
3 //printah untuk mengimpor dua pustaka (library) yang dibutuhkan untuk program ini. IRremote digunakan
4 const int stepsPerRevolution = 200; //adalah konstanta yang menyatakan jumlah langkah pada satu putaran
5 int receiver = 2; //variabel yang menyimpan nomor pin yang digunakan untuk menerima sinyal inframerah.
6
7 //objek-objek yang digunakan untuk menerima dan memproses sinyal inframerah.
8 IRrecv irrecv(receiver);
9 decode_results();
10 //objek motor stepper yang digunakan. Argumen-argumennya adalah jumlah langkah per putaran, serta nomor pin
11 Stepper myStepper(stepsPerRevolution, 8, 9, 10, 11);
12 int previous = 0; //variabel-variabel yang digunakan untuk menyimpan jumlah langkah sebelumnya dan jumlah
13 int stepperSteps = 0;
14 void setup(){//fungsi yang dieksekusi saat program pertama kali dijalankan. Fungsi ini mengatur kecepatan
15 {
16   myStepper.setSpeed(30);
17   Serial.begin(9600);
18   irrecv.enableIRIn();
19 }
20 void loop(){ //fungsi yang dieksekusi secara terus-menerus selama program berjalan. Fungsi ini mengeksekusi
21 if (irrecv.decode()) {
22   atur();
23   irrecv.resume();
24   myStepper.step(stepperSteps - previous);
25   previous = stepperSteps;
26 }
27 }
28 }
29 void atur(){//fungsi yang dipanggil jika ada sinyal inframerah yang diterima. Fungsi ini mengecek je
30 int IRinput = irrecv.decodedIRData.command;
31
32 if (IRinput == 152) { // -
33   stepperSteps -= 1;
34   if (stepperSteps < 0) {
35     stepperSteps = 0;
36   }
37 }
```

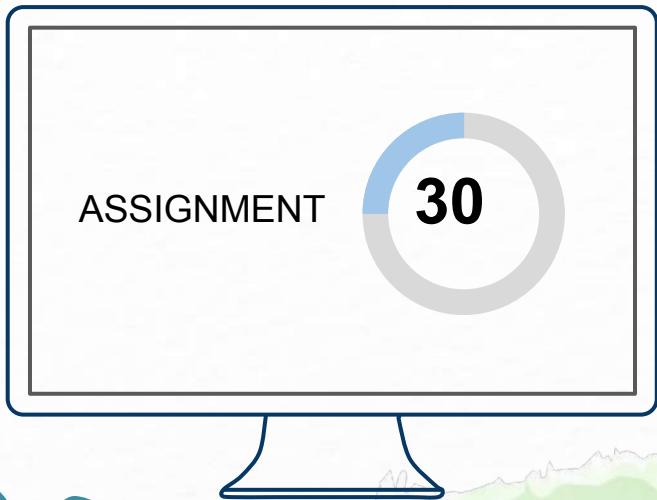
The simulation interface shows a breadboard setup with a microcontroller, a 2-step motor, and a remote control. The breadboard has various components like resistors, capacitors, and jumper wires. The 2-step motor is connected to pins 8, 9, 10, and 11. A digital potentiometer is also connected. On the right, a virtual remote control is used to send commands. The digital potentiometer slider is set to the value '2'.



# THANKS



## A39 - AKBAR FADILA



Kampus  
Merdeka  
INDONESIA JAYA

# Tugas



- - Install Postman
- Buatlah Group, customer, devices, dan dashboard baru pada ThingsBoard

#RintikKarinImpian

# Install Postman

The screenshot shows the Postman web application interface. At the top, there's a navigation bar with links for Home, Workspaces, API Network, and Explore. A search bar is also present. On the left side, there's a sidebar with sections for 'Postman works best with teams' (featuring an illustration of two people), 'Workspaces', 'Private API Network', and 'API Governance'. Below these are sections for 'Learn about Postman Live Collections' and a 'Learn more' button. The main content area is titled 'Explore popular APIs' and lists three collections: 'PayPal APIs' (by PayPal, updated on 20 Apr, 2023), 'spoonacular API' (by spoonacular, updated a month ago), and 'E/PLACE API' (by Super-Eplace-Team-JS-Top, updated on 12 May, 2023). A user profile dropdown menu is open on the right, showing the user 'Akbar Fadila' (akbarrfadila@gmail.com) with options to 'View Profile', 'Settings', 'Sign Out', and 'Switch Accounts'. The bottom of the screen shows a taskbar with various icons and system status indicators.

Home Workspaces API Network Explore

Search Postman

Postman works best with teams

Collaborate in real-time and establish a single source of truth for all API workflows.

Create Team

Workspaces

Private API Network

API Governance

Learn about Postman Live Collections

Postman Collections make it easy to document, share, and test APIs, ultimately streamlining a developer's...

Learn more

PayPal APIs

Official PayPal API Postman Collection including capabilities like Checkout Payments, Payouts, Subscriptions, Webhooks, and more.

By PayPal Updated on 20 Apr, 2023

spoonacular API

By spoonacular Updated a month ago

E/PLACE API

By Super-Eplace-Team-JS-Top Updated on 12 May, 2023

Akbar Fadila  
akbarrfadila@gmail.com

View Profile

Settings

Sign Out

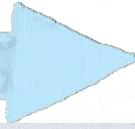
Switch Accounts

Add Account

Fork | 50+ Watch | 100

15° ENG 10:14 PM 15-May-23

# Customer Groups



A screenshot of a web browser showing the ThingsBoard Cloud Platform. The URL in the address bar is `thingsboard.cloud/customerGroups`. The page title is "Customer groups". In the top right corner, it shows "Current subscription: ThingsBoard Cloud Maker" and "Status: Trial ends on the Jun 9, 2023". On the far right, there is a user profile for "Akbar Fadila Tenant administrator" with a "Perbarui" button. The main content area displays a table titled "Customer groups" with two entries:

Created time	Name	Description	Actions
2023-05-15 22:37:17	Customer A39 AKBAR FADILA		
2023-05-10 17:24:09	All		

The left sidebar contains a navigation menu with the following items:

- User groups
- Customer groups (selected)
- All
- Customer A39 AKBAR ...
- Asset groups
- Device groups
- Profiles
- Entity view groups
- Edge groups
- Edge management
- Widgets Library
- Dashboard groups
- OTA updates
- Scheduler

The "Customer groups" section is currently selected. The main content area shows a table titled "Customer groups" with the following data:

Created time	Name	Description	Actions
2023-05-15 22:37:17	Customer A39 AKBAR FADILA		
2023-05-10 17:24:09	All		

At the bottom of the page, there are pagination controls: "Items per page: 10", "1 - 2 of 2", and navigation arrows. The bottom taskbar shows various system icons and the date/time "10:48 PM 15-May-23".

# Device Groups

thingsboard.cloud/deviceGroups

## Device groups

Current subscription: ThingsBoard Cloud Maker  
Status: Trial ends on the Jun 9, 2023

Akbar Fadila Tenant administrator

	Created time	Name	Description	Public					
<input type="checkbox"/>	2023-05-15 22:42:24	Devices-A39 AKBAR FADILA		<input type="checkbox"/>					
<input type="checkbox"/>	2023-05-10 17:39:38	TUGAS 28 -A39 AKBAR FADILA		<input type="checkbox"/>					
	2023-05-10 17:24:09	All		<input type="checkbox"/>					

Items per page: 10 | 1 – 3 of 3 | < < > >>

<https://thingsboard.cloud/deviceGroups>



10:50 PM  
15-May-23

# Dashboard Groups

thingsboard.cloud/dashboardGroups

## Dashboard groups

Current subscription: ThingsBoard Cloud Maker  
Status: Trial ends on the Jun 9, 2023

Akbar Fadila Tenant administrator

	Created time	Name	Description	Public	
<input type="checkbox"/>	2023-05-15 22:39:48	Dashboard-A39 AKBAR FADILA		<input type="checkbox"/>	
	2023-05-10 17:24:09	All		<input type="checkbox"/>	

Items per page: 10 | 1 - 2 of 2 | < > >>

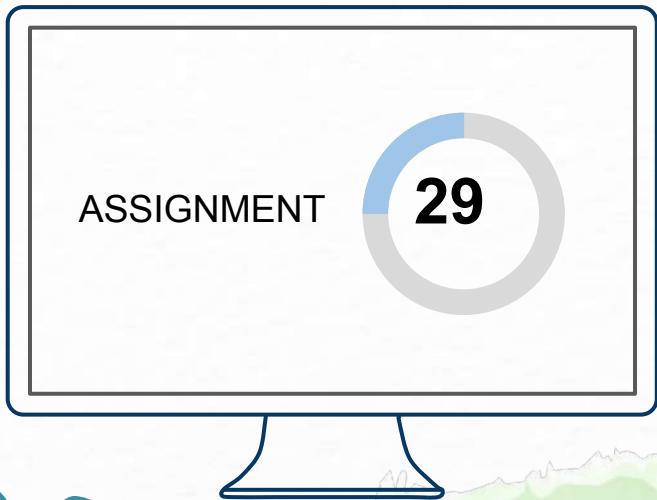
https://thingsboard.cloud/dashboardGroups

16° 10:51 PM 15-May-23

# THANKS



## A39 - AKBAR FADILA



Kampus  
Merdeka  
INDONESIA JAYA

# Tugas



- Buatlah code untuk provisioning device pada wokwi esp32

#RintikKarinImpian



# Codingan Wokwi

<https://wokwi.com/projects/364794724986294273>



# Melakukan Provisioning Pendaftaran serta Konfigurasi pada Blynk

## Firmware configuration di copy lalu di taruh di Wokwi

The screenshot shows the Blynk Device Info page for a device named "A39 akbar fadila". The device is currently Offline. It was last updated "Not updated yet". The device was activated at 8:35 PM Today by "akbarrfadila@gmail.com". The organization is "My organization - 2727OM". The auth token is "gXLhj5ht\_MLcebrDK6tXu6AMpjUoQQEE". The template name is "A39 akbar fadila". The manufacturer is "My organization 2727OM". The SSL setting is "No SSL". The page also displays the firmware configuration code:

```
#define BLYNK_TEMPLATE_ID "TMPL6vjX7-EMK"  
#define BLYNK_TEMPLATE_NAME "A39 akbar fadila"  
#define BLYNK_AUTH_TOKEN "gXLhj5ht_MLcebrDK6tXu6AMpjUoQQEE"
```

A note below the code states: "Template ID, Device Name, and AuthToken should be declared at the very top of the firmware code."

Region: sgp1 Privacy Policy

# Blynk

blynk.cloud/dashboard/292907/product/700170/dashboard

A39 akbar fadila

Metadata Datastreams Events Automations Web Dashboard Mobile Dashboard

This is how the device page will look like for actual devices.

Device name Online  
Device Owner Company Name

Last Hour 6 Hours 1 Day 1 Week 1 Month 3 Months Custom

Kontrol Lampu 1 (V0) Kontrol Lampu 2 (V1)  
Kontrol lampu 3 (V2) Kontrol lampu 4 (V3)  
Kontrol lampu 5 (V4)

Temperature (V6) -2 °C  
Humidity (V5) 22 %

Battery status: 16% available (plugged in)

12° 9:10 PM 15-May-23

# Blynk

blynk.cloud/dashboard/292907/global/filter/2048171/organization/292907/devices/1509463/dashboard

Perbarui

**B**

My organization - 2727OM

← Back

Search

1 Device

A39 akbar fadila

Add Tag

Dashboard Timeline Device Info Metadata Actions Log

Latest Last Hour 6 Hours 1 Day 1 Week 1 Month 3 Months Custom

Kontrol Lampu 1 (Green)

Kontrol Lampu 2 (Purple)

Kontrol lampu 3 (Red)

Kontrol lampu 4 (Blue)

Kontrol lampu 5 (Yellow)

Temperature (11°C)

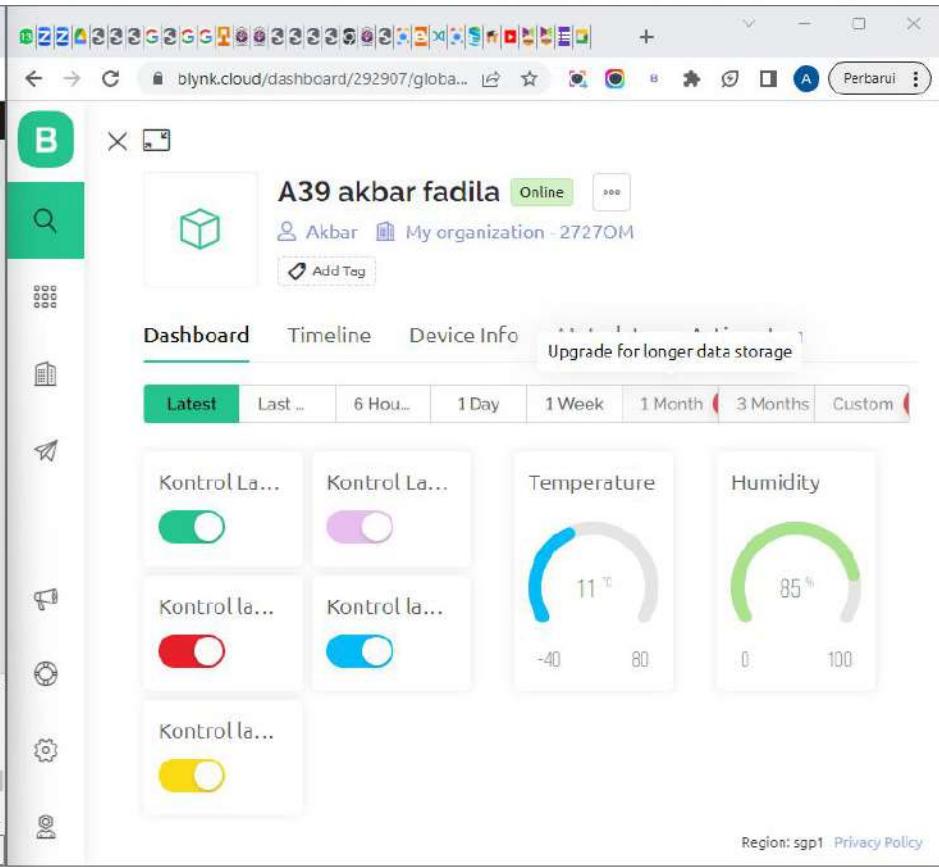
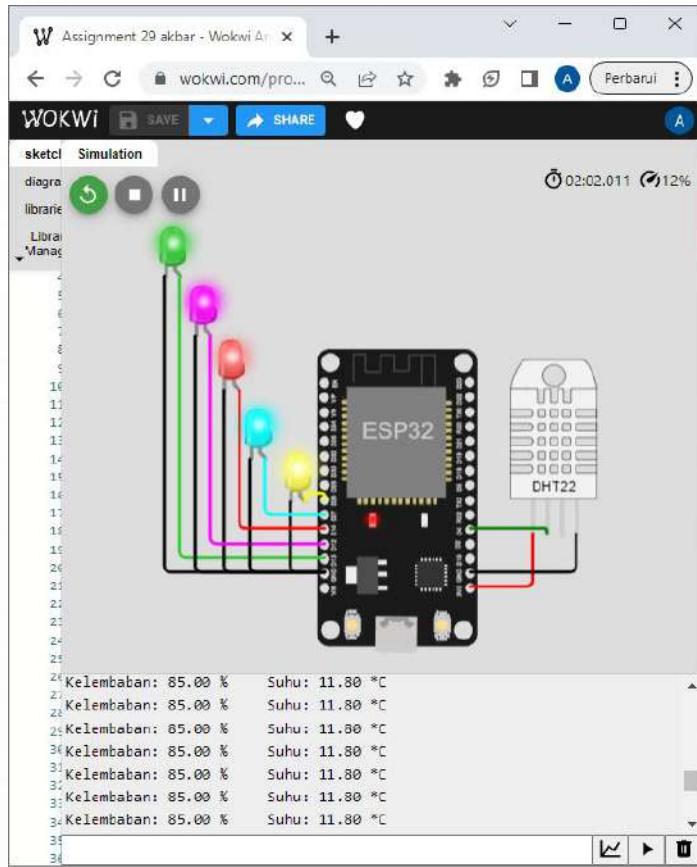
Humidity (85%)

Region: spot Privacy Policy

9:12 PM 15-May-23

The dashboard displays five control buttons for lamps (1-5) and two circular gauges for Temperature (11°C) and Humidity (85%). The left sidebar shows one device named 'A39 akbar fadila'.

# Hasil



# Melakukan PROVISIONING konfigurasi Untuk Menghubungkan ESP32 dengan Blynk

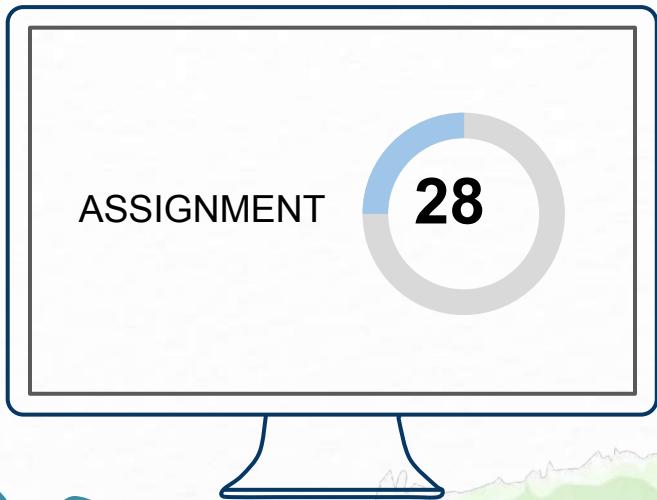
sketch.ino • diagram.json libraries.txt Library Manager

```
1 //A39- AKBAR FADILA
2
3 //Library yang digunakan termasuk WiFi, WiFiClient, BlynkSimpleEsp32, dan DHT (untuk sensor suhu dan kelembaban).
4 #include <WiFi.h>
5 #include <WiFiClient.h>
6 #include <BlynkSimpleEsp32.h>
7 #include <DHT.h>
8
9 //Dilakukan definisi BLYNK_TEMPLATE_ID, BLYNK_TEMPLATE_NAME, dan BLYNK_AUTH_TOKEN untuk menghubungkan ke server Blynk.
10 #define BLYNK_TEMPLATE_ID "TMPL6vjX7-EMk"
11 #define BLYNK_TEMPLATE_NAME "A39 akbar fadila"
12 #define BLYNK_AUTH_TOKEN "gXLhj5ht_MLCebrDK6txU6AMpjUoQQEE"
13
14 //Mengatur parameter SSID dan password WiFi yang digunakan.
15 char ssid[] = "Wokwi-GUEST";
16 char pass[] = "";
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85 void setup()//Dilakukan inisialisasi serial untuk output debugging.
86 {
87     // Debug console
88     Serial.begin(9600);
89     dht.begin(); // mulai sensor DHT
90     Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass); //Memulai koneksi ke server Blynk menggunakan token otorisasi dan informasi WiFi.
91     timer.setInterval(1000L, myTimerEvent); //Mengatur interval waktu untuk memanggil fungsi myTimerEvent().
92     //Mengatur pin-pin yang digunakan sebagai output.
93     pinMode(13, OUTPUT);
94     pinMode(12, OUTPUT);
95     pinMode(14, OUTPUT);
96     pinMode(27, OUTPUT);
97     pinMode(26, OUTPUT);
98
99 }
100
101 void loop()//Loop utama program.
```

# THANKS



## A39 - AKBAR FADILA



Kampus  
Merdeka  
INDONESIA JAYA

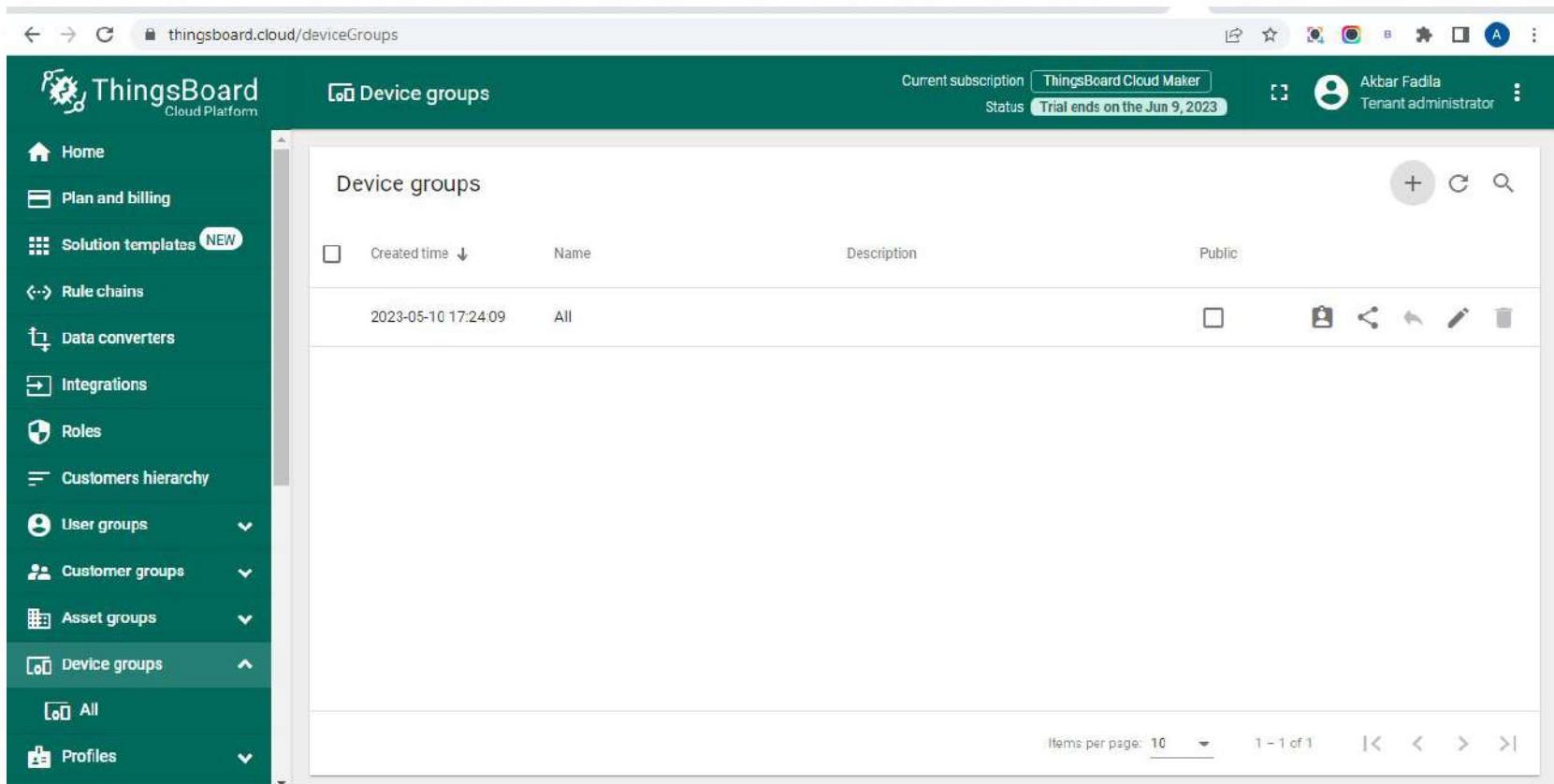
# Tugas



- Buat akun thingsboard tambah satu data dari wokwi esp32

#RintikKarinImpian

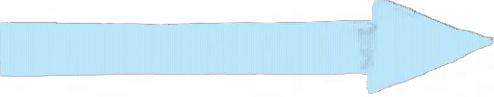
 Setelah membuat akun ThingsBoard, Lalu masuk Ke ThingsBoard dan pilih Devices Groups Lalu klik tanda +



The screenshot shows the ThingsBoard Cloud Platform interface. The left sidebar has a dark green background with white icons and text. The 'Device groups' option is selected, indicated by an upward arrow icon next to it. Other options include Home, Plan and billing, Solution templates (NEW), Rule chains, Data converters, Integrations, Roles, Customers hierarchy, User groups, Customer groups, Asset groups, Device groups (with an upward arrow), All, and Profiles. The main content area has a light gray header with the title 'Device groups'. Below the header is a table with the following columns: Created time (sorted by descending time), Name, Description, and Public. A single row is listed: '2023-05-10 17:24:09' under 'Created time', 'All' under 'Name', and 'Public' under 'Public'. To the right of the table are several icons for actions like edit, delete, and share. At the bottom of the table are pagination controls: 'Items per page: 10', '1 - 1 of 1', and navigation arrows. The top right corner of the interface shows the user's information: 'Akbar Fadila' (Tenant administrator) and a trial status: 'Trial ends on the Jun 9, 2023'.

Created time	Name	Description	Public
2023-05-10 17:24:09	All		   

# Buat Name Entity Group Details



The screenshot shows the ThingsBoard Cloud Platform interface. The left sidebar has a dark green background with white icons and text. The main menu items include Home, Plan and billing, Solution templates (NEW), Rule chains, Data converters, Integrations, Roles, Customers hierarchy, User groups, Customer groups, Asset groups, Device groups, All, and Profiles. The 'Device groups' item is currently selected, indicated by a grey background.

The top navigation bar shows the URL thingsboard.cloud/deviceGroups, the current subscription status (ThingsBoard Cloud Maker), and the user information (Akbar Fadila, Tenant administrator).

A modal window titled "Add entity group" is open in the center. It consists of two steps:

- 1 Entity group details**: A "Name" field containing "TUGAS 28 -A39 AKBAR FADILA".
- 2 Share entity group**: An optional section labeled "Optional".

Below the name input, there is a "Description" input field which is currently empty. At the bottom of the modal, there are "Cancel" and "Add" buttons, along with a "Next: Share entity group" link.

The background shows a list of device groups on the right side of the page, with one entry visible: "TUGAS 28 -A39 AKBAR FADILA" created on "2023-05-10 17:24:0".

# KLIK Tanda +

The screenshot shows the ThingsBoard Cloud Platform interface. At the top, there is a navigation bar with a back arrow, forward arrow, refresh icon, and a search bar containing the URL `thingsboard.cloud/deviceGroups/f6a43f50-ef1e-11ed-8899-9ba30be05e76`. To the right of the search bar are various browser icons and a user profile section.

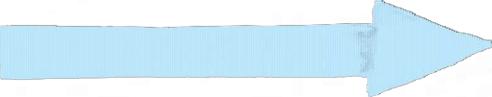
The main header includes the ThingsBoard logo, the text "Device groups > TUGAS 28 -A39 AKBAR FADILA", and information about the current subscription ("ThingsBoard Cloud Maker") and status ("Trial ends on the Jun 9, 2023").

The left sidebar contains a navigation menu with the following items:

- Home
- Plan and billing
- Solution templates (NEW)
- Rule chains
- Data converters
- Integrations
- Roles
- Customers hierarchy
- User groups
- Customer groups
- Asset groups
- Device groups
- All
- TUGAS 28 -A39 AKBAR FADILA

The main content area is titled "TUGAS 28 -A39 AKBAR FADILA: Devices". It features a table with columns: "Name", "Device profile", and "Label". The table header includes a checkbox, sorting by "Created time" (with an arrow pointing down), and a search icon. Below the table, a message says "No devices found". At the bottom of the page, there are pagination controls for "Items per page: 10" and navigation arrows.

# Buat devices details dengan nama IOT



thingsboard.cloud/deviceGroups/f6a43f50-ef1e-11ed-8899-9ba30be05e76

Home Plan and billing Solution templates NEW Rule chains Data converters Integrations Roles Customers hierarchy User groups Customer groups Asset groups Device groups All TUGAS 28 -A39 AKBA...

Add new device

Current subscription | ThingsBoard Cloud Maker

Akbar Fadila Tenant administrator

**1 Device details**

Name \* IOT

Label

Device profile \* default

Select existing device profile  Create new device profile

Is gateway

Description

Next: Credentials

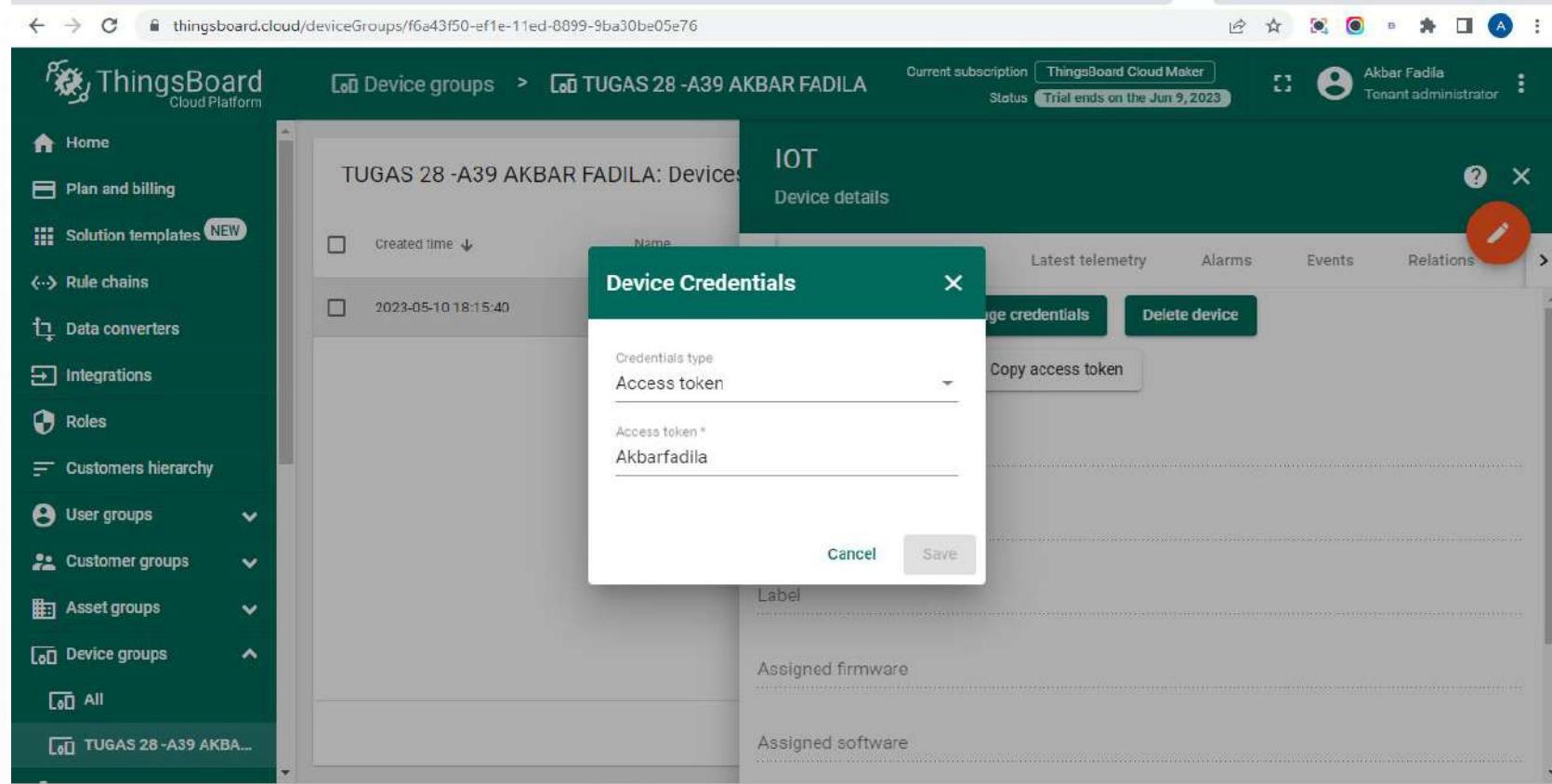
Cancel Add

+  

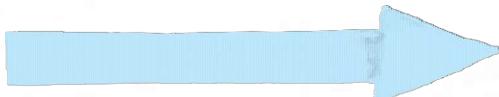
Klik IOT yang telah dibuat lalu klik manage credentials

The screenshot shows the ThingsBoard Cloud Platform interface. On the left, a sidebar lists various management options: Home, Plan and billing, Solution templates (NEW), Rule chains, Data converters, Integrations, Roles, Customers hierarchy, User groups, Customer groups, Asset groups, Device groups (selected), and All. The main content area displays a device group named "TUGAS 28 -A39 AKBAR FADILA". Inside this group, a specific device is shown with the name "IOT". The device details page includes fields for Name (IOT), Device profile (default), Label, Assigned firmware, and Assigned software. At the top right of the device details card, there is a red circle containing a white pencil icon, indicating that the device can be edited. Below the card, there are buttons for "Open details page", "Manage credentials" (which is highlighted in green), and "Delete device".

# Saya menggunakan Access Token > Akbarfadila



The screenshot shows the ThingsBoard Cloud Platform interface. On the left, there's a sidebar with various navigation options like Home, Plan and billing, Solution templates (NEW), Rule chains, Data converters, Integrations, Roles, Customers hierarchy, User groups, Customer groups, Asset groups, Device groups, All, and TUGAS 28 -A39 AKBAR FADILA. The main area shows a 'Device groups' section with 'TUGAS 28 -A39 AKBAR FADILA' selected. Below it, there's a 'IOT Device details' section with tabs for Latest telemetry, Alarms, Events, and Relations. A large blue arrow points from the top left towards the main content. In the center, a modal window titled 'Device Credentials' is open. It has a dropdown for 'Credentials type' set to 'Access token', and an input field below it containing 'Access token\*' and 'Akbarfadila'. At the bottom of the modal are 'Cancel' and 'Save' buttons. To the right of the modal, a context menu is open with options 'Change credentials', 'Delete device', and 'Copy access token', with 'Copy access token' being highlighted by a red circle.



# Data dari Wokwi telah terbaca oleh ThingsBoard



thingsboard.cloud/deviceGroups/f6a43f50-ef1e-11ed-8899-9ba30be05e76

ThingsBoard Cloud Platform

Device groups > TUGAS 28 -A39 AKBAR FADILA

Current subscription: ThingsBoard Cloud Maker  
Status: Trial ends on the Jun 9, 2023

Akbar Fadila Tenant administrator

IOT Device details

TUGAS 28 -A39 AKBAR FADILA: Device

Created time Name

2023-05-10 18:15:40 IOT

Details Attributes Latest telemetry Alarms Events Relations

Latest telemetry

Last update time	Key	Value
2023-05-10 18:18:40	Temperature	-7
2023-05-10 18:18:39	Humidity	37

Items per page: 10 1 - 2 of 2

Home Plan and billing Solution templates NEW Rule chains Data converters Integrations Roles Customers hierarchy User groups Customer groups Asset groups Device groups All TUGAS 28 -A39 AKBABA...

# Pilih show on widget

The screenshot shows the ThingsBoard Cloud Platform interface. On the left, a sidebar lists various management options: Home, Plan and billing, Solution templates (NEW), Rule chains, Data converters, Integrations, Roles, Customers hierarchy, User groups, Customer groups, Asset groups, Device groups, All, and TUGAS 28 -A39 AKBAR FADILA. The 'Device groups' section is currently selected.

The main content area displays the 'Device groups' page for the group 'TUGAS 28 -A39 AKBAR FADILA'. It includes a navigation bar with 'Device groups' and 'TUGAS 28 -A39 AKBAR FADILA', and a status bar indicating 'Current subscription: ThingsBoard Cloud Maker' and 'Status: Trial ends on the Jun 9, 2023'. A user profile for 'Akbar Fadila Tenant administrator' is also shown.

A modal window titled 'IOT' is open, showing 'Device details'. It lists two entries: 'Created time' (2023-05-10 18:15:40) and 'IOT'. Below this, a tab bar includes 'Details', 'Attributes', 'Latest telemetry' (which is selected), 'Alarms', 'Events', and 'Relations'. A red circle highlights the 'Show on widget' button located at the bottom right of this tab bar.

The 'Latest telemetry' section displays '2 telemetry units selected' and a table with three rows:

Last update time	Key	Value
2023-05-10 18:18:40	Temperature	-7
2023-05-10 18:18:39	Humidity	37

At the bottom of the table, there are buttons for 'Show on widget' (orange) and 'Edit' (pencil icon).

Klik Add to dashboard disini saya memilih Charts

thingsboard.cloud/deviceGroups/f6a43f50-ef1e-11ed-8899-9ba30be05e76

ThingsBoard Cloud Platform

Device groups > TUGAS 28 -A39 AKBAR FADILA

Current subscription: ThingsBoard Cloud Maker  
Status: Trial ends on Jun 9, 2023

Akbar Fadila Tenant administrator

TUGAS 28 -A39 AKBAR FADILA: Devices

Device details

IOT

Details Attributes Latest telemetry Alarms Events Relations

Add to dashboard

Timeseries Bar Chart

Temperature (avg 13.16), Humidity (avg 95.4)

Time	Temperature (avg 13.16)	Humidity (avg 95.4)
18:38:00	~45	~120
18:38:10	~10	~100
18:38:20	~10	~110
18:38:30	~10	~105
18:38:40	~10	~108
18:38:50	~10	~112

# ThingsBoard IOT DASHBOARD dengan Wokwi

The image illustrates the integration of Wokwi and ThingsBoard for IoT dashboarding. On the left, the Wokwi IDE is open, showing an ESP32 sketch with a DHT22 sensor connected. The sketch code includes the following print statements:

```
Temperature : 8.00 Humidity : 92.00
Sending data...
Temperature : 8.00 Humidity : 92.00
Sending data...
Temperature : 8.00 Humidity : 92.00
Sending data...
Temperature : 8.00 Humidity : 92.00
```

The Wokwi interface also shows a simulation window displaying the current values: Temperature: 8.0°C and Humidity: 92.0%.

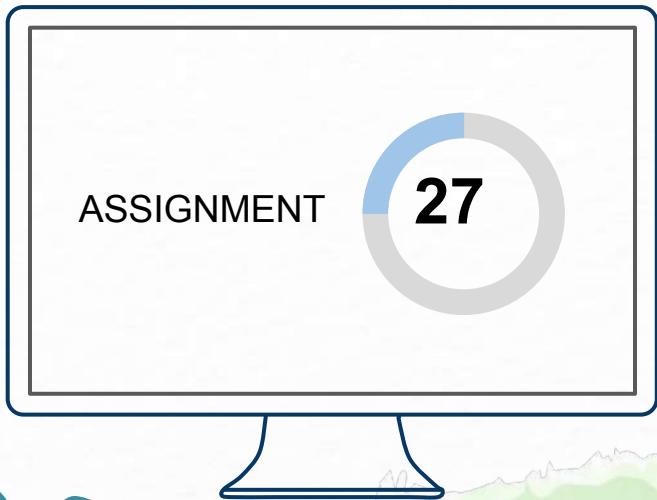
On the right, the ThingsBoard IoT Dashboard is displayed. It features a "Timeseries Bar Chart" for the "IOT Dashboard". The chart shows data points for Humidity (blue bars) and Temperature (green bars) over time intervals from 18:44:50 to 18:45:40. The chart indicates a constant value of 92 for Humidity and 8 for Temperature.

<https://wokwi.com/projects/364333841054218241>

# THANKS



## A39 - AKBAR FADILA



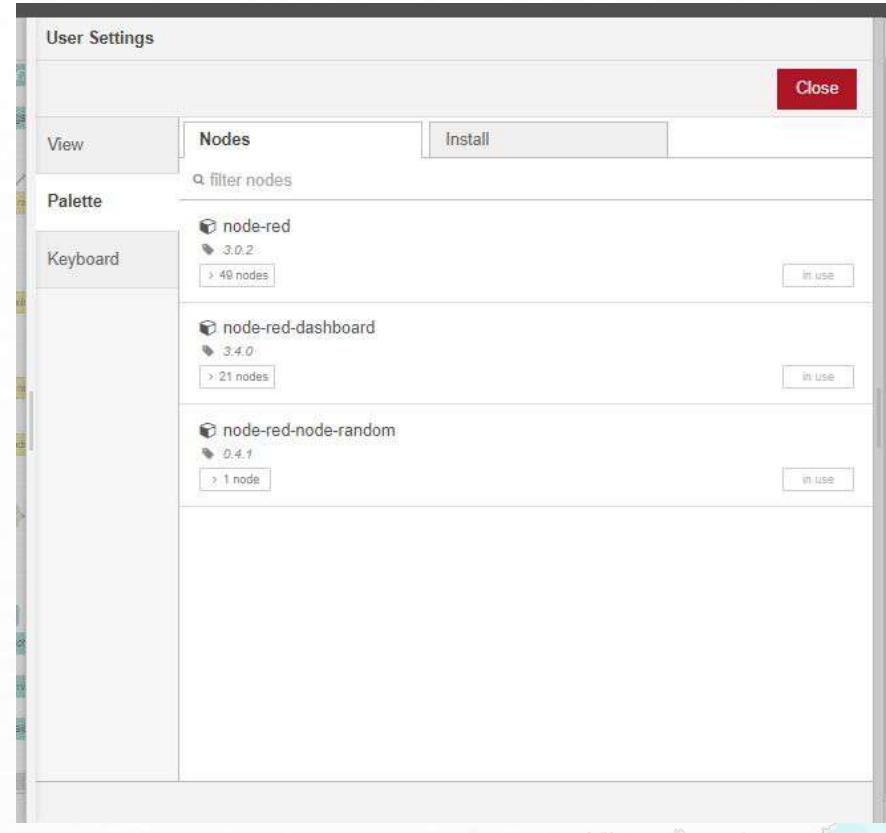
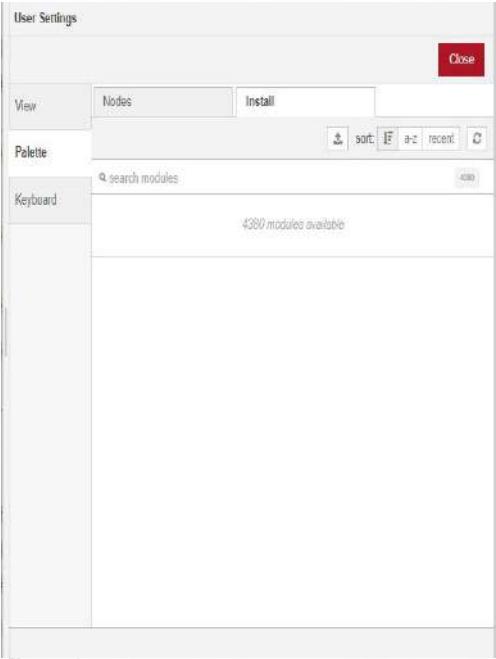
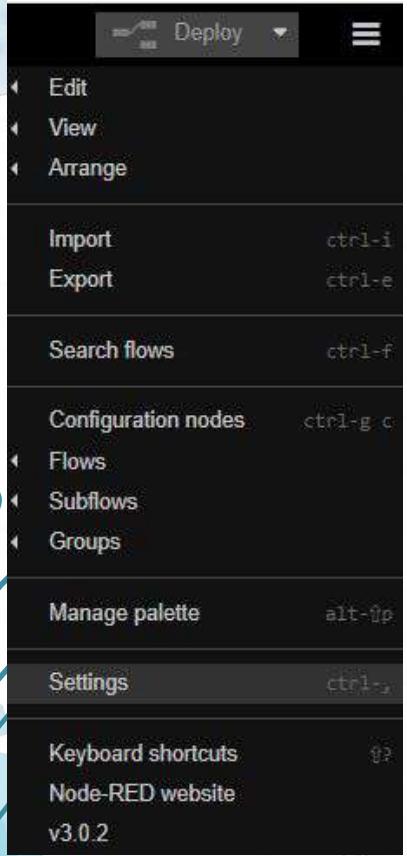
Kampus  
Merdeka  
INDONESIA JAYA

# Tugas

- Visualisasikan data yang dapat dihasilkan oleh solusi iot dengan tema "Pertambangan"



# Install node-red-node-random



# Node-RED (Visualisasi Data Monitoring Pertambangan)



127.0.0.1:1880/#flow/c81882c921d8c2f2

Node-RED

Flow 3

The screenshot shows a complex Node-RED flow titled "Flow 3". The flow starts with a timestamp node, which branches into multiple parallel paths. Each path contains a "random" node followed by a "Data Monitoring logam" node. These nodes then connect to various output nodes representing different environmental parameters: TEMPERATURE, HUMIDITY, CO2, CO, WATER QUALITY, and Kecepatan Angin. The "Kecepatan Angin" path also includes a "data kecepatan angin" node. The "Gelarim" path includes a "data monitoring gelarim" node. The "Kedalaman Tambang" path includes a "Kedalaman Tambang" node. The "data Kedalaman Tambang" path includes a "data Kedalaman Tambang" node. The "Monitoring Logam" path includes a "Monitoring Logam" node.

Deploy

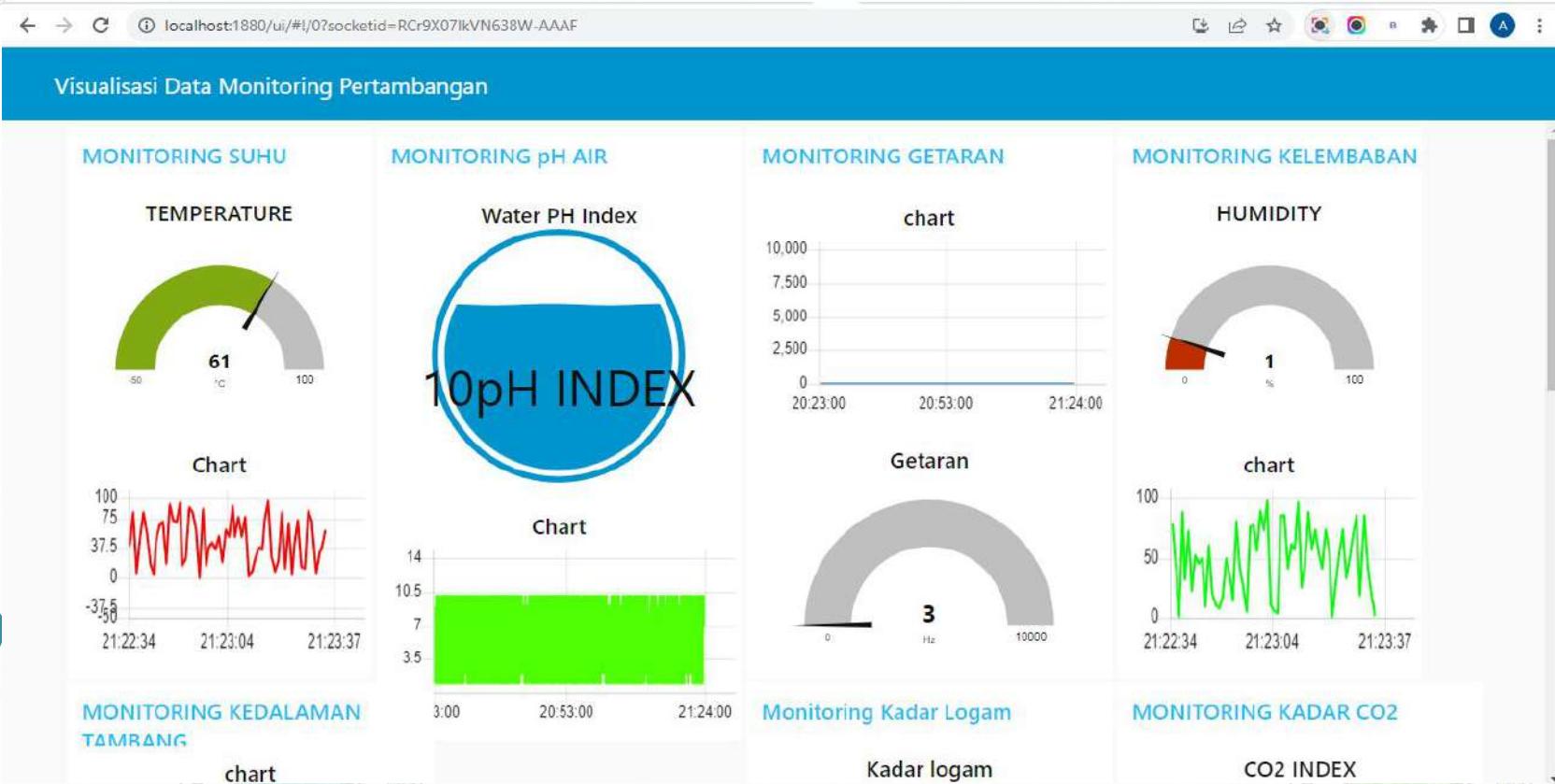
dashboard

Layout Site Theme

Tabs & Links

- Home
  - Visualisasi Data Monitoring Pertambangan
    - MONITORING pH AIR
    - MONITORING SUHU
    - MONITORING GETARAN
    - MONITORING KEDALAMAN TAMBANG
    - MONITORING KELEMBABAN
    - Monitoring Kadar Logam
    - MONITORING KADAR CO2
    - MONITORING KADAR CO
    - MONITORING KECEPATAN ANGIN

# Hasil Visualisasi data



# Visualisasi data tampilan penuh



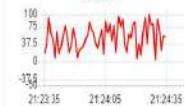
## Visualisasi Data Monitoring Pertambangan

### MONITORING SUHU

#### TEMPERATURE



#### Chart



### MONITORING pH AIR

#### Water PH Index



#### Chart

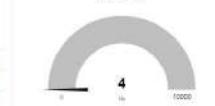


### MONITORING GETARAN

#### Getaran



#### chart

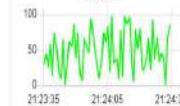


### MONITORING KELEMBABAN

#### HUMIDITY



#### chart



### MONITORING KEDALAMAN TAMBANG

#### chart



#### Kedalaman Tambang



### Monitoring Kadar Logam

#### Kadar logam

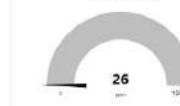


#### chart

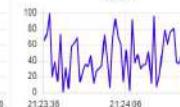


### MONITORING KADAR CO2

#### CO2 INDEX



#### chart

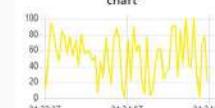


### MONITORING KADAR CO

#### CO INDEX



#### chart



### MONITORING KECEPATAN ANGIN

#### Kecepatan Angin



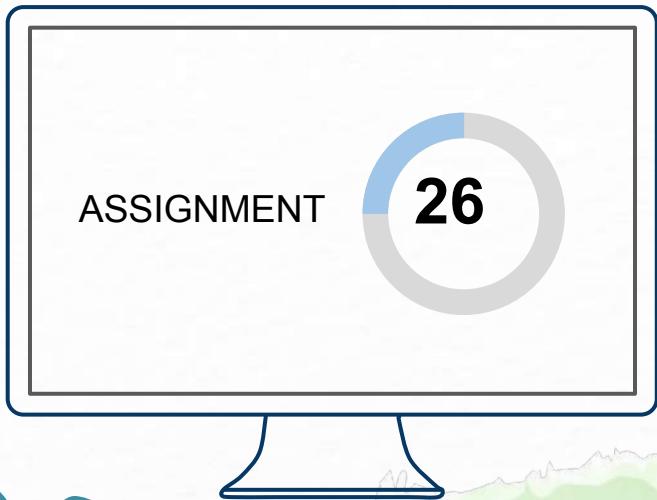
#### chart



# THANKS



## A39 - AKBAR FADILA





MySkill

MSIB

Kampus  
Merdeka  
BERBELAJAR JAUH

## Tugas

- Instalasi Nodred-dashboard
- Dengan Nodered-dashboard tampilkan jumlah Perangkat IoT yang terkoneksi pada topik

#RintikKarinImpian

## INSTALASI Node-red-dashboard

```
node-red      X + ▾
```

Microsoft Windows [Version 10.0.22621.1635]  
(c) Microsoft Corporation. All rights reserved.

```
C:\Users\akbar>node --version  
v18.14.2
```

```
C:\Users\akbar>npm --version  
9.5.0
```

```
C:\Users\akbar>npm install node-red-dashboard
```

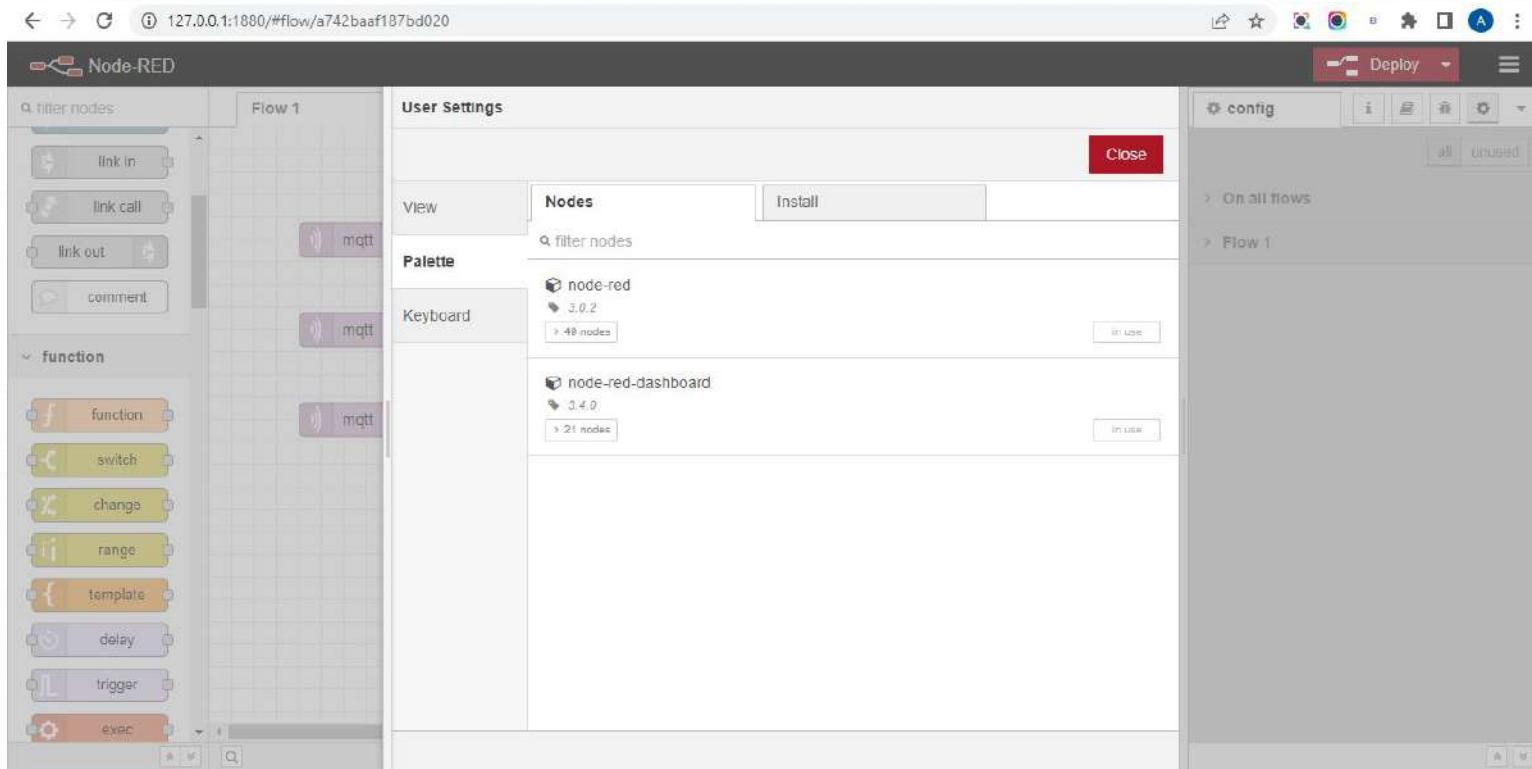
```
added 54 packages in 29s
```

```
C:\Users\akbar>node-red  
9 May 14:34:32 - [info]
```

```
Welcome to Node-RED  
=====
```

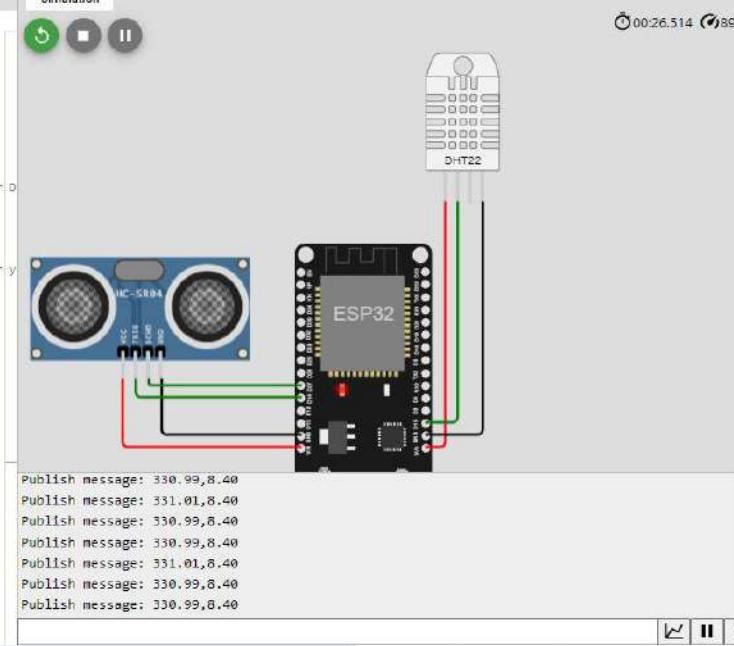
```
9 May 14:34:32 - [info] Node-RED version: v3.0.2  
9 May 14:34:32 - [info] Node.js version: v18.14.2  
9 May 14:34:32 - [info] Windows_NT 10.0.22621 x64 LE  
9 May 14:34:34 - [info] Loading palette nodes  
9 May 14:34:36 - [info] Dashboard version 3.4.0 started at /ui  
9 May 14:34:36 - [info] Settings file : C:\Users\akbar\.node-red\settings.js  
9 May 14:34:36 - [info] Context store : 'default' [module=memory]  
9 May 14:34:36 - [info] User directory : \Users\akbar\.node-red  
9 May 14:34:36 - [warn] Projects disabled : editorTheme.projects.enabled=false
```

## HASIL Instalasi Node-red-dashboard



```

1 //AAB-AKBAR FADILA
2 #include <WiFi.h>
3 #Include <PubSubClient.h>
4 #Include "DHTesp.h"
5 //Pada baris di atas, dilakukan pengimporan library WiFi, PubSubClient, dan DHTesp.
6
7 #define dhtpin 15
8 #define ECHO_PIN 27
9 #define TRIG_PIN 34
10 //Baris ini mendefinisikan tiga konstanta yang akan digunakan sebagai pin yang terhubung ke sensor DHT22.
11
12 const char* ssid = "Wokwi-GUEST";
13 const char* password = "";
14 const char* mqtt_server = "broker.hivemo.com";
15 //Pada baris di atas, disediakan tiga variabel yang harus diatur sesuai dengan jaringan dan broker yang digunakan.
16
17 WiFiClient espClient;
18 PubSubClient client(espClient);
19 //Pada baris di atas, dibuat objek untuk klien WiFi dan objek klien MQTT.
20
21 unsigned long lastMsg = 0;
22 #define MSG_BUFFER_SIZE (50)
23 char msg[MSG_BUFFER_SIZE];
24 DHTesp dht;
25 //Pada baris ini, didefinisikan beberapa variabel yang akan digunakan nanti.
26
27 void setup_wifi() { //perintah koneksi wifi
28   delay(10);
29   Serial.println();
30   Serial.print("Connecting to ");
31   Serial.println(ssid);
32
33   WiFi.mode(WIFI_STA); //setting wifi chip sebagai station/client
34   WiFi.begin(ssid, password); //koneksi ke jaringan wifi
35
36   while (WiFi.status() != WL_CONNECTED) { //perintah tunggu esp32 sampai terkoneksi ke wifi
37     delay(500);
38   }
39 }
```



<https://wokwi.com/projects/364235370957399041>

# Codingan Wokwi

WOKWI SAVE SHARE AKBAR T26 Docs

sketch.ino diagram.json libraries.txt Library Manager

```
37 delay(500);
38 Serial.print(".");
39 }
40 randomSeed(micros());
41 serial.println("");
42 Serial.println("wifi connected");
43 Serial.println("IP address: ");
44 Serial.println(WiFi.localIP());
45 }
46 //Pada baris di atas, dibuat fungsi setup_wifi yang akan terkoneksi ke jaringan WiFi.
47 void setup() {
48   pinMode(TRIG_PIN, OUTPUT);
49   pinMode(ECHO_PIN, INPUT);
50   Serial.begin(115200);
51   setup_wifi(); //memanggil void setup_wifi untuk dieksekusi
52   String clientId = "wskill";
53   client.setServer(mqtt_server, 1883); //perintah connecting / koneksi awal ke broker
54   client.connect(clientId.c_str());
55   dht.setup(dhtpin, DHTesp::DHT22);
56   //Pada fungsi setup(), pin sensor didefinisikan sebagai input/output dan dilakukan koneksi ke jar
57 }
58 float readDistanceCM() {
59   digitalWrite(TRIG_PIN, LOW);
60   delayMicroseconds(2);
61   digitalWrite(TRIG_PIN, HIGH);
62   delayMicroseconds(10);
63   digitalWrite(TRIG_PIN, LOW);
64   int duration = pulseIn(ECHO_PIN, HIGH);
65   return duration * 0.034 / 2;
66 }
67 //Pada fungsi readDistanceCM(), terdapat kode untuk membaca jarak dari sensor ultrasonik HC-SR04.
68 void loop() {
```

Simulation

01:07.927 1026

Publish message: 330.99,8.40  
Publish message: 330.97,8.40  
Publish message: 330.99,8.40  
Publish message: 330.99,8.40  
Publish message: 330.97,8.40  
Publish message: 330.99,8.40

<https://wokwi.com/projects/364235370957399041>

# Codingan Wokwi

The screenshot shows the Wokwi simulation environment for a project titled "WOKWI". The code in the editor is as follows:

```
73 void loop() {  
74     client.loop();  
75     TempAndHumidity data = dht.getTempAndHumidity();  
76     float distance = readDistanceCM();  
77     String jarak = String(distance);  
78     String suhu = String(data.temperature);  
79     String payload = jarak + "," + suhu;  
80     Serial.println("Publish message: ");  
81     Serial.println(payload);  
82     client.publish("/myskill/tugas26/distance", jarak.c_str());  
83     client.publish("/myskill/tugas26/temperature", suhu.c_str());  
84     client.publish("/myskill/tugas26/PERANGKAT", payload.c_str());  
85     delay(1000);  
86 }  
87  
88 //Program diatas menggunakan loop() yang berjalan secara terus menerus selama program berjalan. Did  
89
```

The simulation window displays the hardware setup with an ESP32 board connected to an HC-SR04 ultrasonic sensor and a DHT22 sensor. The publish log shows the following output:

```
Publish message: 330.99,8.40  
Publish message: 330.99,8.40  
Publish message: 330.99,8.40  
Publish message: 330.97,8.40  
Publish message: 330.99,8.40  
Publish message: 330.99,8.40  
Publish message: 330.99,8.40
```

<https://wokwi.com/projects/364235370957399041>

# Node-RED

127.0.0.1:1880/#flow/a742baaf187bd020

Deploy

Node-RED

Flow 1

filter nodes

common

- inject
- debug
- complete
- catch
- status
- link in
- link call
- link out
- comment

/myskill/tugas26/distance  
connected

/myskill/tugas26/temperature  
connected

/myskill/tugas26/PERANGKAT  
connected

Distance

Temperature

Jumlah Perangkat Terhubung abc

function

switch

Layout Site Theme

dashboard

Tabs & Links

- Home
- Distance
- Temperature
- Perangkat IOT
- Jumlah Perangkat Terhubung

The screenshot shows the Node-RED interface with a flow titled 'Flow 1'. The flow consists of three parallel input nodes: '/myskill/tugas26/distance', '/myskill/tugas26/temperature', and '/myskill/tugas26/PERANGKAT'. These nodes are connected to a 'Distance' node, a 'Temperature' node, and a 'function' node respectively. The output of the 'PERANGKAT' node is also connected to the 'function' node. The 'function' node has an output labeled 'Jumlah Perangkat Terhubung' with a value 'abc'. On the left, there is a sidebar with a 'common' section containing nodes like inject, debug, complete, catch, status, link in, link call, link out, and comment. Below that is a 'function' section with function and switch nodes. On the right, there is a dashboard with tabs for Layout, Site, and Theme, and a sidebar for Tabs & Links listing Home, Distance, Temperature, Perangkat IOT, and Jumlah Perangkat Terhubung.

# Node-RED

127.0.0.1:1880/#flow/a742baaf187bd020

Node-RED

Flow 1

```
graph LR; inject1[/inject/] --> mqtt1[/myskill/tugas26/distance/]; mqtt1 --> function1[function]; mqtt2[/myskill/tugas26/temperature/]; mqtt2 --> function1;
```

The screenshot shows a Node-RED flow titled "Flow 1". It consists of four nodes: an "inject" node, three MQTT nodes, and one function node. The first MQTT node subscribes to the topic "/myskill/tugas26/distance" and has its output connected to the function node. The second MQTT node subscribes to the topic "/myskill/tugas26/temperature" and also has its output connected to the function node. The function node is labeled "function".

Edit mqtt in node

Properties

- Server: broker.hivemq.com:1883
- Action: Subscribe to single topic
- Topic: /myskill/tugas26/distance
- QoS: 0
- Output: auto-detect (parsed JSON object, string or buffer)
- Name: Name

Enabled: Enabled

dashboard

Edit gauge node

Properties

- Group: [Home] Distance
- Size: auto
- Type: Gauge
- Label: Distance
- Value format: {{value}}
- Units: units
- Range: min 2 max 400
- Colour gradient:
- Sectors: 2 optional optional ... 400
- Fill centre:

Enabled: Enabled

# Node-RED

127.0.0.1:1880/#flow/a742baaf187bd020

Node-RED

Flow 1

Edit mqtt in node

Properties

- Server: broker.hivemq.com:1883
- Action: Subscribe to single topic
- Topic: /myskill/tugas26/temperature
- QoS: 0
- Output: auto-detect (parsed JSON object, string or buf)
- Name: Name

Done

Deploy

dashboard

Layout Site Theme

Tabs & Links

- Home
  - Distance
  - Temperature
  - Perangkat IOT
  - Jumlah Perangkat Terhubung

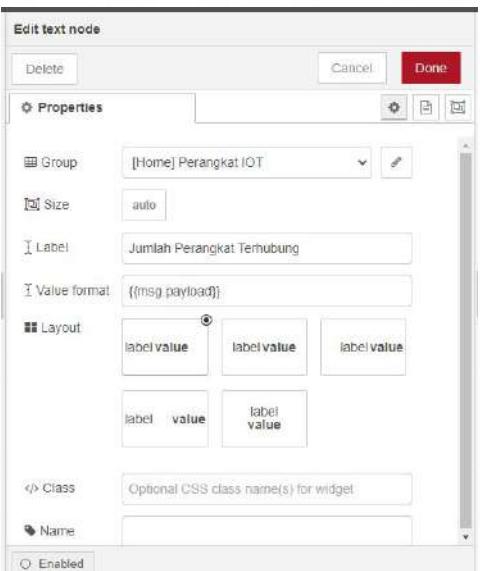
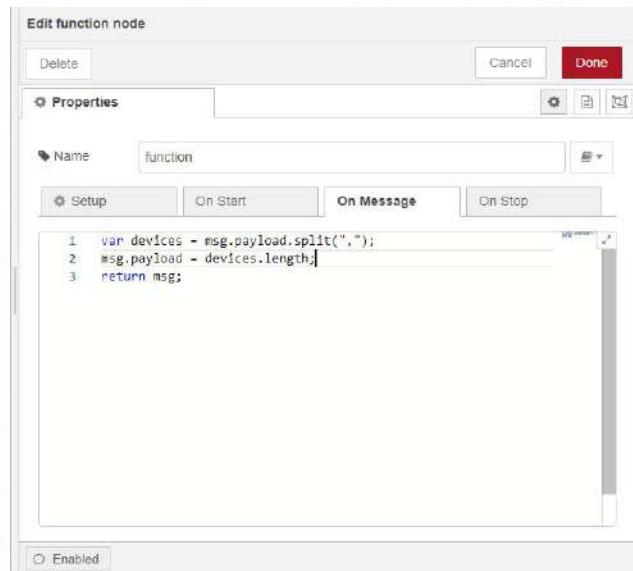
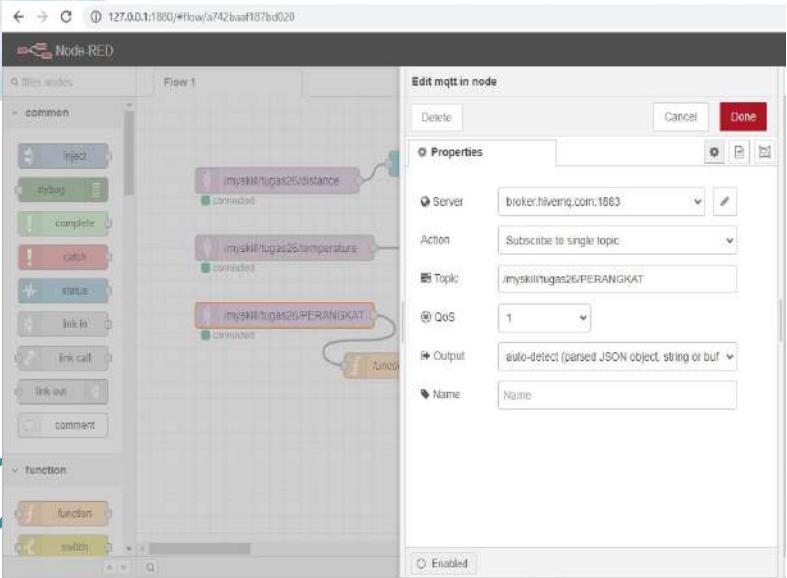
Edit gauge node

Properties

- Group: [Home] Temperature
- Size: auto
- Type: Gauge
- Label: Temperature
- Value format: {{value}}
- Units: units
- Range: min -40 max 80
- Colour gradient:
- Sectors: -40 optional optional 80
- Fill gauge from centre:
- Enabled:

Cancel Done

# Node-RED



# MQTTX

The screenshot shows the MQTTX application interface. On the left, a sidebar titled "Connections" lists two entries: "akbar@broker.hivemq.com" and "akbarTUGAS26@brok...". The "akbarTUGAS26" entry is highlighted with a light blue background. The main panel is titled "akbarTUGAS26" and shows a list of received messages. A green button labeled "+ New Subscription" is visible at the top of the message list. The messages are listed in a scrollable area:

- Topic: /myskill/tugas26/PERANGKAT  
QoS: 0  
330.96, 8.40  
2023-05-09 16:48:37:141
- Topic: /myskill/tugas26/PERANGKAT  
QoS: 0  
330.99, 8.40  
2023-05-09 16:48:39:441
- Topic: /myskill/tugas26/PERANGKAT  
QoS: 0  
331.01, 8.40  
2023-05-09 16:51:27:841

At the bottom of the message list, there are buttons for "Payload", "Plaintext", "QoS: 1", "Retain" (which is checked), and "Meta". Below these buttons, the message "/myskill/tugas26/PERANGKAT" is repeated. There are also three small circular icons with arrows pointing left, right, and up-right.

# Node-RED Dashboard



# HASIL

 WOKWI SAVE SHARE

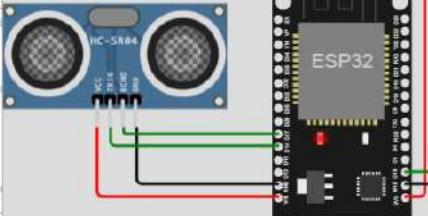
sketch.ino diagram.json libraries.txt

Library Manager

```
//A39-AKBAR FADILA
1 #include <WiFi.h>
2 #include <PubSubClient.h>
3 #include "DHTesp.h"
4 //Pada baris di atas, disediakan peralatan yang diperlukan untuk
5 //mendukung ESP32
6 //Baris ini mendefinisikan tiga konstanta
7 //yang diperlukan untuk menghubungkan
8 //ESP32 ke WiFi
9 #define DHTPIN 15
10 #define ECHO_PIN 27
11 #define TRIG_PIN 14
12 //Baris ini mendefinisikan tiga konstanta
13 //yang diperlukan untuk mendukung
14 //dengan sensor HC-SR04
15 //Baris ini mendefinisikan tiga konstanta
16 //yang diperlukan untuk mendukung
17 //dengan sensor DHT
18 WiFiClient espClient;
19 PubSubClient client(espClient);
20 //Pada baris di atas, dibuat objek
21 //WiFiClient dan PubSubClient
22 unsigned long lastMsg = 0;
23 #define MSG_BUFFER_SIZE (90)
24 char msg[MSG_BUFFER_SIZE];
25 DHTesp dht;
26 //Pada baris ini, didefinisikan bahwa
27 //ESP32 akan terhubung ke WiFi
28 void setup_wifi() { //perintah koneksi
29   delay(10);
30   serial.println();
31   Serial.print("Connecting to ");
32   Serial.println(ssid);
33   WiFi.mode(WIFI_STA); //setting WiFi
34   WiFi.begin(ssid, password); //koneksi WiFi
35   while (WiFi.status() != WL_CONNECTED) {
36     delay(500);
37 }
```

Simulation

⌚ 03:44:740 ⚡ 96%



Distance

Distance

331.01

Temperature

Temperature

8.4

Perangkat IOT

Jumlah Perangkat Terhubung 2

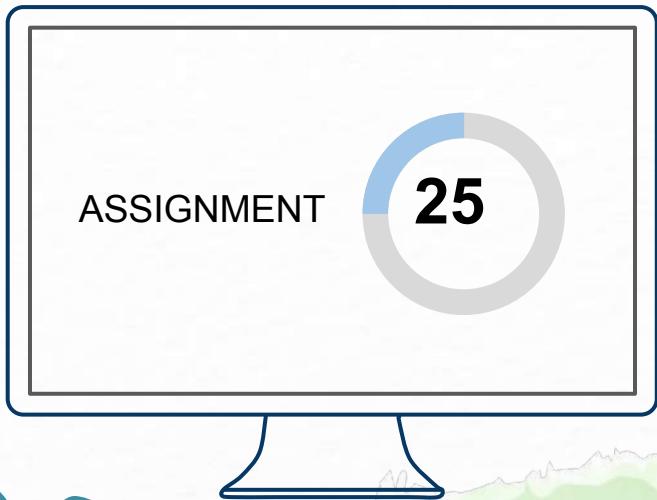
Node-RED Dashboard

localhost:1880/ui/#/0?socket...

# THANKS



## A39 - AKBAR FADILA



# Tugas



- Buatlah mqtt code untuk mengendalikan led yang mana hidup dan mati.

#RintikKarinImpian

# Codingan Wokwi

wokwi.com/projects/363030277211301889

WOKWI SAVE SHARE MQTT A39 akbar tgs 25 Docs A

sketch.ino • diagram.json • Libraries.txt • Library Manager

```
1 #include <WiFi.h>
2 #include <PubSubClient.h>
3 //dilakukan pendefinisi pin LED yang akan digunakan serta konfigurasi WiFi dan MQTT broker yang akan
4 #define red 2
5 #define yellow 4
6 #define green 18
7 #define blue 19
8
9 // Update these with values suitable for your network.
10
11 const char* ssid = "Wokwi-GUEST";
12 const char* password = "";
13 const char* mqtt_server = "broker.hivemq.com";
14
15 WiFiClient espClient;
16 PubSubClient client(espClient);
17
18 //pada fungsi setup_wifi(), dilakukan proses koneksi ke jaringan WiFi yang telah ditentukan dan menu
19 void setup_wifi() { //paprtant koneksi wifi
20     delay(10);
21     // We start by connecting to a WiFi network
22     Serial.print("Connecting to ");
23     Serial.println(ssid);
24
25     WiFi.mode(WIFI_STA); //setting wifi chip sebagai station/client
26     WiFi.begin(ssid, password); //koneksi ke jaringan wifi
27
28     while (WiFi.status() != WL_CONNECTED) { //perintah tunggu esp32 sampai terkoneksi ke wifi
29         delay(500);
30         Serial.print(".");
31     }
32
33     randomSeed(micros());
34
35     Serial.println("");
36     Serial.println("WiFi connected");
37 }
```

Simulation

05:26:970 100%

Connecting to Wokwi-GUEST  
....  
Wifi connected:  
IP address:  
10.10.6.2  
Attempting MQTT connection...Connected

<https://wokwi.com/projects/363030277211301889>

# Perintah LED Merah menyala

The image shows a screenshot of the MQTTX application and a browser window displaying MQTT messages, alongside a schematic diagram of an ESP32 connected to five LEDs.

**MQTTX Application:**

- Connections:** akbar
- Subscriptions:** /myskill/p/mqtt/# QoS 0
- Published Messages:**
  - Topic: /myskill/p/mqtt/ledmerah QoS: 0  
Payload: 1 (published at 2023-04-30 21:17:58:101)
  - Topic: /myskill/p/mqtt/ledmerah QoS: 0  
Payload: 1 (published at 2023-04-30 21:18:15:944)
  - Topic: /myskill/p/mqtt/ledmerah QoS: 0  
Payload: 1 (published at 2023-04-30 21:18:16:227)

**Browser Window:**

- URL:** wokwi.com/...
- Code:** 01:04.079 (92%)
- Circuit Diagram:** An ESP32 microcontroller is connected to five LEDs. The ESP32's digital pins 2, 3, 4, 5, and 6 are connected to the阳极 (Anode) of each LED. The 阴极 (Cathode) of all LEDs is connected to ground through a common resistor.
- Terminal Output:** connection...Connected  
connection...Connected  
[/myskill/p/mqtt/ledmerah] 1

# Perintah LED Merah mati

The image shows two windows side-by-side. On the left is the MQTTX application, which displays a connection named 'akbar' with a single subscription to the topic '/myskill/p/mqtt/ledmerah/#'. It shows three messages received at different times. On the right is the Wokwi IDE, which contains a simulation of an ESP32 microcontroller connected to four LEDs. The circuit diagram shows the ESP32 pins D1, D2, D3, and D4 connected to the respective red LEDs via green wires. Below the circuit diagram, the Wokwi terminal window shows the text 'connection...Connected' repeated four times, followed by '[/myskill/p/mqtt/ledmerah] 0'.

MQTTX

File Edit View Window Help

Connections

akbar 14

+ New Subscription

/myskill/p/mqtt/ledmerah/# QoS 0

Topic: /myskill/p/mqtt/ledmerah QoS: 0  
2023-04-30 21:36:32:418

Topic: /myskill/p/mqtt/ledmerah QoS: 0  
2023-04-30 21:38:15:684

Topic: /myskill/p/mqtt/ledmerah QoS: 0  
2023-04-30 21:38:15:995

Payload: JSON QoS: 0 Retain

/myskill/p/mqtt/ledmerah 0

Wokwi.com...

SAVE SHARE

simulation Code

00:20.882 17%

ESP32

D1 D2 D3 D4

connection...Connected  
connection...Connected  
connection...Connected  
connection...Connected  
[/myskill/p/mqtt/ledmerah] 0  
connection...Connected

# Perintah LED Kuning menyala

The image shows two windows side-by-side. On the left is the MQTTX application interface, which includes a sidebar with connection management icons and a main pane for viewing connections and messages. A message from the topic `/myskill/p/mqtt/ledmerah` is shown with a payload of 0. Another message from the topic `/myskill/p/mqtt/ledkuning` is shown with a payload of 1. On the right is a browser window displaying a Wokwi simulation. The simulation shows an ESP32 microcontroller connected to four LEDs via breadboard wires. The LEDs are colored red, green, blue, and yellow. The yellow LED is labeled "LED Kuning". The browser's developer tools panel shows the serial port output with the text: "connection...Connected", "connection...Connected [ /myskill/p/mqtt/ledmerah ] 0", "connection...Connected", "connection...Connected [ /myskill/p/mqtt/ledkuning ] 1", and "connection...Connected".

MQTTX

File Edit View Window Help

Connections

akbar 16

+ New Subscription

/myskill/p/mqtt/# QoS 0

Topic: /myskill/p/mqtt/ledmerah QoS: 0  
0  
2023-04-30 21:38:15.995

Topic: /myskill/p/mqtt/ledkuning QoS: 0  
1  
2023-04-30 21:39:25.170

Topic: /myskill/p/mqtt/ledmerah QoS: 0  
1  
2023-04-30 21:39:25.473

Payload: JSON QoS: 0 Retain

/myskill/p/mqtt/ledmerah

/myskill/p/mqtt/ledkuning

Wokwi.com ...

SAVE SHARE

Code

00:31,149 17%

ESP32

LED Kuning

connection...Connected

connection...Connected [ /myskill/p/mqtt/ledmerah ] 0

connection...Connected

connection...Connected [ /myskill/p/mqtt/ledkuning ] 1

connection...Connected

# Perintah LED Kuning mati

The image shows two windows side-by-side. On the left is the MQTTK application, which displays a connection named 'akbar' with a green status. It shows a subscription to the topic '/myskill/p/mqtt/#' with QoS 0. There are three messages received from this topic:

- Topic: /myskill/p/mqtt/ledkuning  
QoS: 0  
Payload: 1  
Timestamp: 2023-04-30 21:39:25:473
- Topic: /myskill/p/mqtt/ledkuning  
QoS: 0  
Payload: 0  
Timestamp: 2023-04-30 21:40:24:139
- Topic: /myskill/p/mqtt/ledkuning  
QoS: 0  
Payload: 0  
Timestamp: 2023-04-30 21:40:24:436

The bottom of the MQTTK window shows a message input field with the text '/myskill/p/mqtt/ledkuning' and a green send button.

On the right is the Wokwi IDE, which contains a screenshot of an ESP32 microcontroller connected to four LEDs (red, yellow, green, blue) via breadboard wires. The code editor shows the following code:ESP32
connection...Connected
connection...Connected
[/myskill/p/mqtt/ledkuning] 1
connection...Connected
connection...Connected
connection...Connected
[/myskill/p/mqtt/ledkuning] 0

# Perintah LED Hijau menyala



The screenshot shows the MQTTX application interface. On the left is a sidebar with icons for connection management, subscriptions, and other tools. The main area displays a connection named "akbar" with 20 topics. One topic, "/myskill/p/mqtt/ledkuning", has a message payload of "0". Another topic, "/myskill/p/mqtt/ledhijau", has two messages with payloads "1" and "0" respectively. At the bottom, there's a terminal window showing command-line output related to the MQTT topics.

MQTTX

File Edit View Window Help

Connections

akbar 20

+ New Subscription

/myskill/p/mqtt/# QoS 0

Topic: /myskill/p/mqtt/ledkuning  
QoS: 0  
0  
2023-04-30 21:40:24:436

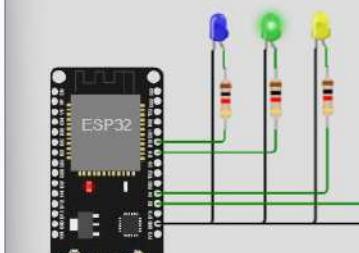
Topic: /myskill/p/mqtt/ledhijau  
QoS: 0  
1  
2023-04-30 21:42:19:260

Topic: /myskill/p/mqtt/ledhijau  
QoS: 0  
1  
2023-04-30 21:42:19:642

Payload: JSON QoS: 0 Retain

/myskill/p/mqtt/ledhijau  
1

```
[/myskill/p/mqtt/ledkuning] 1
connection...Connected
connection...Connected
[/myskill/p/mqtt/ledhijau] 0
connection...Connected
connection...Connected
[/myskill/p/mqtt/ledhijau] 1
```



# Perintah LED Hijau mati

The image shows a dual-monitor setup. On the left monitor, the MQTTX application is open, displaying a connection to 'akbar' with a subscription to the topic '/myskill/p/mqtt/ledhijau'. Three messages are listed, all with QoS 0 and a payload of 1, timestamped 2023-04-30 21:42:19.642. On the right monitor, the Wokwi IDE is open, showing a code editor with the following code:

```
connection...Connected  
[ /myskill/p/mqtt/ledkuning ] 0  
connection...Connected  
connection...Connected  
[ /myskill/p/mqtt/ledhijau ] 1  
connection...Connected  
[ /myskill/p/mqtt/ledhijau ] 0
```

Below the code editor is a schematic diagram of an ESP32 microcontroller connected to four LEDs (red, yellow, green, blue) via a breadboard. The red LED is connected to digital pin 13. The yellow LED is connected to digital pin 12. The green LED is connected to digital pin 11. The blue LED is connected to digital pin 10. All four pins are connected to ground through 220 ohm resistors.

# Perintah LED Biru menyala

MQTTX

File Edit View Window Help

Connections

akbar 24

+ New Subscription

/myskill/p/mqtt/# QoS 0

Plaintext

All Received Published

Topic: /myskill/p/mqtt/ledhijau QoS 0  
S: 0  
0  
2023-04-30 21:42:45:875

Topic: /myskill/p/mqtt/ledbiru QoS 0  
S: 0  
1  
2023-04-30 21:44:46:764

Topic: /myskill/p/mqtt/ledbiru QoS 0  
S: 0  
1  
2023-04-30 21:44:47:060

Payload: JSON QoS: 0 Retain

Meta

/myskill/p/mqtt/ledbiru

connection...Connected  
connection...Connected  
[/myskill/p/mqtt/ledhijau] 1  
connection...Connected  
[/myskill/p/mqtt/ledhijau] 0  
connection...Connected  
[/myskill/p/mqtt/ledbiru] 1

wokwi.com...

LIVE SHARE

Code

00:53:548 21%

```
graph LR; ESP32[ESP32] --- D2[D2]; D2 --- BlueLED((Blue LED)); BlueLED --- GND1[GND];
```

# Perintah LED Biru mati

MQTTX

File Edit View Window Help

Connections

akbar 26

+ New Subscription

/myskill/p/mqtt/# QoS 0

Topic: /myskill/p/mqtt/ledbiru QoS: 0  
1  
2023-04-30 21:44:47:060

Topic: /myskill/p/mqtt/ledbiru QoS: 0  
0  
2023-04-30 21:45:53:310

Topic: /myskill/p/mqtt/ledbiru QoS: 0  
0  
2023-04-30 21:45:53:598

Payload: JSON QoS: 0 Retain

/myskill/p/mqtt/ledbiru  
0

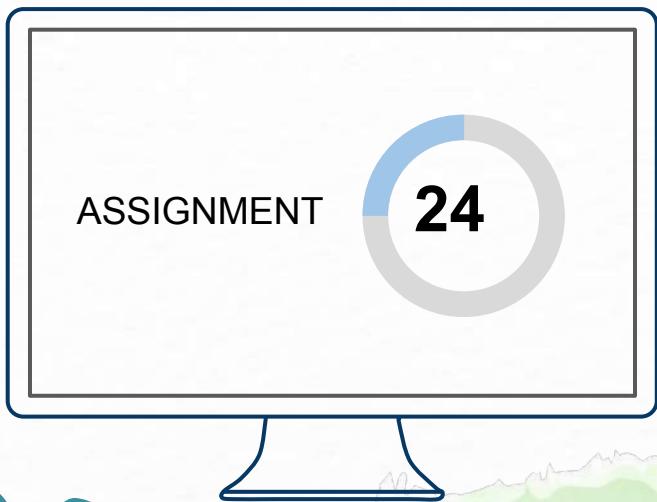
wokwi.com...  
SAVE SHARE  
Code  
01:08.097 20%

```
[/myskill/p/mqtt/ledbiru] 0 connection..Connected  
[/myskill/p/mqtt/ledbiru] 1 connection..Connected  
connection..Connected  
connection..Connected  
connection..Connected  
[/myskill/p/mqtt/ledbiru] 0
```

# THANKS



## A39 – AKBAR FADILA



# Tugas



MySkill

MSIB

Kampus  
Merdeka

- Buatlah http code untuk mengambil data dari : <https://pokeapi.co/>  
Merupakan sebuah api public berupa data pokemon
- Buatlah mqtt code mengirim pesan Ke <https://www.hivemq.com/public-mqtt-broker/> sebagai broker, dan Tampilkan datanya, contoh bisa memakai mqtt client online

#KintikKarirImpian



Home

About

API v2

GraphQL v1beta

PokeAPI proudly announces Beta support for GraphQL. Access our free console at [beta.pokeapi.co/graphql/console](https://beta.pokeapi.co/graphql/console) and take a look at the [documentation](#)



The RESTful Poké**M**on API

Serving over 330,000,000 API calls each month!

All the Poké**M**on data you'll ever need in one place,  
easily accessible through a modern RESTful API.

[Check out the docs!](#)

Try it now!

<https://pokeapi.co/api/v2/>

pokemon/ditto



Submit

Need a hint? Try [pokemon/ditto](#), [pokemon-species/aegislash](#), [type/3](#), [ability/battle-armor](#), or

## HTML CODE

```
④ index.html > ⑤ html > ⑥ head > ⑦ style > ⑧ ^  
1  <!DOCTYPE html>  
2  <html lang="en">  
3  <head>  
4      <meta charset="UTF-8">  
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">  
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">  
7      <script src="script.js"></script>  
8      <title>POKEMON API</title>  
9      <style>  
10         <!--  
11             background-color: #00ffff;  
12         -->  
13         h1{  
14             text-align: center;  
15         }  
16         .columns{display: flex;  
17             justify-content: space-around;  
18         }  
19         ul, #pokemonList{  
20             cursor: pointer;  
21         }  
22         h2{  
23             text-align: center;  
24         }  
25         .columns{display: flex;  
26             justify-content: space-around;  
27         }  
28         ul, #pokemonList{  
29             cursor: pointer;  
30         }  
31     </style>  
32  </head>  
33  </body>  
34  <h1>DATA POKEMON API</h1>  
35  <h2>(A39-AKBAR FADILA)</h2>  
36  <div class="columns">  
37      <ul id="pokemonList">  
38          <li>POKEMON</li>  
39      </ul>  
40      <div id="result">  
41          <h3>RESULT</h3>  
42          <div id="detail">  
43              <h4>DETAIL</h4>  
44          </div>  
45      </div>  
46  </div>  
47  </div>  
48  </div>  
49  </body>  
50  </html>
```



## JAVASCRIPT CODE

```
js script.js > detail > then0 callback > then() callback
1   fetch ('https://pokeapi.co/api/v2/pokemon')
2   .then(
3     function(response){
4       if(response.status != 200){
5         console.log('error' + response.status)
6         return
7       }
8       response.json().then(function(data){
9         const pokemons = data.results
10        pokemons.forEach(pokemon => {
11          document.getElementById( pokemonList )
12            .insertAdjacentHTML('beforeend',
13              `<li onclick = 'detail("${pokemon.url}")'> ${pokemon.name}</li>` )
14        })
15      })
16    })
17  .catch(function(error){
18    console.log(error)
19  })
20}
21
22 function detail(url){
23   fetch(url).then(function(response){
24     response.json().then(function(pokemon){
```

```
25
26   document.getElementById('detail').innerHTML=' '
27
28   document.getElementById('detail')
29     .insertAdjacentHTML('beforeend',
30       `<h3>${pokemon.name}</h3>
31       <img src = '${pokemon.sprites.front_default}'>
32       <p>Height : ${pokemon.height}</p>
33       <p>Weight : ${pokemon.weight}</p>
34     `)
35   }
36 }
37
38
39 }
40 }
41 }
```



# RESULT

← → C File | C:/Users/akbar/Pictures/New%20folder/index.html

## DATA POKEMON API

(A39-AKBAR FADILA)

### POKEMON

- bulbasaur
- ivysaur
- venusaur
- charmander
- charmeleon
- charizard
- squirtle
- wartortle
- blastoise
- caterpie
- metapod
- butterfree
- weedle
- kakuna
- beedrill
- pidgey
- pidgeotto
- pigeon
- rattata
- raticate

### RESULT

butterfree



Height : 11

Weight : 320

Buatlah mqtt code mengirim pesan  
Ke

<https://www.hivemq.com/public-mqtt-broker/> sebagai broker, dan  
Tampilkan datanya, contoh bisa  
memakai mqtt client online

<https://wokwi.com/projects/363490300910118913>

wokwi.com/projects/363490300910118913

**WOKWI** SAVE SHARE STAR IMAGE PDF A ⋮

**A39 AKBAR T24** EDIT

sketch.ino diagram.json libraries.txt Library Manager Docs A

```
1 //A39- Akbar Tugas 24
2 #include <WiFi.h>
3 #include <PubSubClient.h>
4
5 // Update these with values suitable for your network.
6
7 const char* ssid = "WOKWI-GUEST";
8 const char* password = "";
9 const char* mqtt_server = "broker.hivemq.com";
10
11 WiFiClient espClient;
12 PubSubClient client(espClient);
13 unsigned long lastMsg = 0;
14 #define MSG_BUFFER_SIZE (50)
15 char msg[MSG_BUFFER_SIZE];
16 int value = 0;
17
18 void setup_wifi() { //perintah koneksi wifi
19     delay(10);
20     // We start by connecting to a WiFi network
21     Serial.println();
22     Serial.print("Connecting to ");
23     Serial.println(ssid);
24
25     WiFi.mode(WIFI_STA); //setting wifi chip sebagai station/client
26     WiFi.begin(ssid, password); //koneksi ke jaringan wifi
27
28     while (WiFi.status() != WL_CONNECTED) { //perintah tunggu esp32 sampai terkoneksi ke wifi
29         delay(500);
30         Serial.print(".");
31     }
32
33     randomSeed(micros());
34
35     Serial.println("");
36     Serial.println("wifi connected");
37     Serial.println("IP address: ");
```

**Simulation**

00:47.966 30%



Connecting to WOKWI-GUEST  
....  
Wifi connected  
IP address:  
10.10.0.2  
Attempting MQTT connection...Connected

wokwi.com/projects/363490300910118913

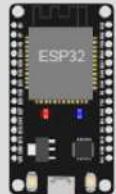
WOKWI SAVE SHARE A39 AKBAR T24

sketch.ino diagram.json libraries.txt Library Manager Simulation 00:55.666 20%

```
37 Serial.println("IP address: ");
38 Serial.println(WiFi.localIP());
39 }
40
41 void callback(char* topic, byte* payload, unsigned int length) { //perintah untuk menampilkan data ke serial
42     Serial.print("Message arrived [");
43     Serial.print(topic);
44     Serial.print("] ");
45     for (int i = 0; i < length; i++) { //mengecek jumlah data yang ada di topik mqtt
46         Serial.print((char)payload[i]);
47     }
48     Serial.println();
49
50     // Switch on the LED if an 1 was received as first character
51     if ((char)payload[0] == '1') {
52         digitalWrite(2, LOW); // Turn the LED on (Note that LOW is the voltage level
53         // but actually the LED is on; this is because
54         // it is active low on the ESP-01)
55     } else {
56         digitalWrite(2, HIGH); // Turn the LED off by making the voltage HIGH
57     }
58 }
59
60 void reconnect() { //perintah koneksi esp32 ke mqtt broker baik itu sebagai publisher atau subscriber
61     // Loop until we're reconnected
62     while (!client.connected()) {
63         Serial.print("Attempting MQTT connection...");
64         // perintah membuat client id agar mqtt broker mengenali board yang kita gunakan
65         String clientId = "ESP32Client";
66         clientId += String(random(0xffff), HEX);
67         // Attempt to connect
68         if (client.connect(clientId.c_str())) {
69             Serial.println("Connected");
70             // Once connected, publish an announcement...
71             client.publish("/MSIBMySkill/p/mqtt", "A39-Akbar Fadila"); //perintah publish data ke alamat
72             // ... and resubscribe
73             client.subscribe("/MSIBMySkill/p/mqtt"); //perintah subscribe data ke mqtt broker
    }
}

```

Attempting MQTT connection...Connected  
Publish message: A39-Akbar Fadila #1  
Message arrived [/MSIBMySkill/p/mqtt] A39-Akbar Fadila #1  
Publish message: A39-Akbar Fadila #2  
Message arrived [/MSIBMySkill/p/mqtt] A39-Akbar Fadila #2  
Attempting MQTT connection...Connected  
Publish message: A39-Akbar Fadila #3



WOKWI

SAVE

SHARE



A39 AKBAR T24

Docs

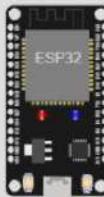


```
72     // ... and resubscribe
73     client.subscribe("/MSIBMySkill/p/mqtt"); //perintah subscribe data ke mqtt broker
74 } else {
75     serial.print("failed, rc=");
76     Serial.print(client.state());
77     Serial.println(" try again in 5 seconds");
78     // wait 5 seconds before retrying
79     delay(5000);
80 }
81
82 }
83
84 void setup() {
85     pinMode(2, OUTPUT);      // inisialisasi pin 2 / ledbuiltin sebagai output
86     Serial.begin(115200);
87     setup_wifi(); //memanggil void setup_wifi untuk dieksekusi
88     client.setServer(mqtt_server, 1883); //perintah connecting / koneksi awal ke broker
89     client.setCallback(callback); //perintah menghubungkan ke mqtt broker untuk subscribe data
90 }
91
92 void loop() {
93     if (!client.connected()) {
94         reconnect();
95     }
96     client.loop();
97
98     unsigned long now = millis();
99     if (now - lastMsg > 2000) { //perintah publish data
100         lastMsg = now;
101         ++value;
102         sprintf (msg, MSG_BUFFER_SIZE, "A39-Akbar Fadila #%d", value); //perintah mempersiapkan data u
103         Serial.print("Publish message: ");
104         Serial.println(msg);
105         client.publish("/MSIBMySkill/p/mqtt", msg); //perintah publish data ke mqtt broker, yang di publ
106
107     }
108 }
```

Simulation



⌚ 01:05.533 ⚡ 32%

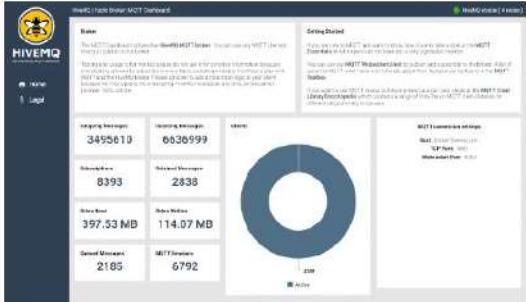


Message arrived [/MSIBMySkill/p/mqtt] A39-Akbar Fadila #4  
Publish message: A39-Akbar Fadila #5  
Message arrived [/MSIBMySkill/p/mqtt] A39-Akbar Fadila #5  
Publish message: A39-Akbar Fadila #6  
Message arrived [/MSIBMySkill/p/mqtt] A39-Akbar Fadila #6  
Publish message: A39-Akbar Fadila #7  
Message arrived [/MSIBMySkill/p/mqtt] A39-Akbar Fadila #7



# Hivemq

Public MQTT Broker

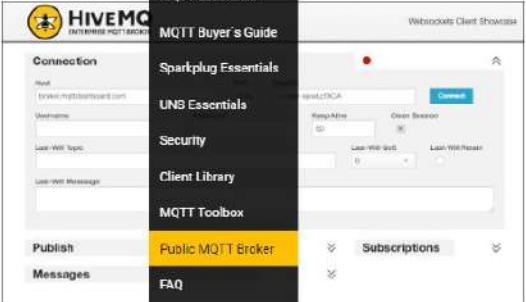


The MQTT Dashboard shows the following metrics:

Topic	Subscribers	Publishers
topic1	3495610	6636599
topic2	8393	2838
Total	397.53 MB	114.07 MB
Client Count	2165	6792

Setting Started

MQTT Browser Client



The MQTT Browser Client interface includes sections for:

- Connection (Host: broker.hivemq.com, Port: 1883, Username: guest, Password: guest)
- Sparkplug Essentials
- UNB Essentials
- Security
- Client Library
- MQTT Toolbox
- Publish (highlighted)
- Subscriptions
- Messages
- FAQ

The HiveMQ MQTT Browser Client is an MQTT WebSocket client interface. Use any modern browser on any device as a full-fledged MQTT client and take full advantage of the MQTT protocol.

You can access the broker at:

Broker: [broker.hivemq.com](https://broker.hivemq.com)

TCP Port: 1883

<https://www.hivemq.com/public-mqtt-broker/>

Try MQTT Browser Client

hivemq.com/demos/websocket-client/

# HIVEMQ

Websockets Client Showcase

Need a fully managed MQTT broker?  
Get your own Cloud broker and connect up to 100 devices for free.

Get your free account

### Connection

Host: broker.hivemq.com Port: 8884 ClientID: clientId-TjrX78SiMe connected

Username:  Password:  Keep Alive: 60 SSL:  Clean Session:

Last-Will Topic:  Last-Will QoS: 0 Last-Will Retain:

Last-Will Message:

### Publish

Topic: testtopic/1 QoS: 0 Retain:  Publish

Message:

### Subscriptions

Add New Topic Subscription

### Messages

hivemq.com/demos/websocket-client/

### Publish

Topic: /MSIBMyskill/p/mqtt QoS: 2 Retain:  Publish

Message:

---

### Subscriptions

Add New Topic Subscription

QoS: 2 /MSIBMyskill/p/mqtt X

---

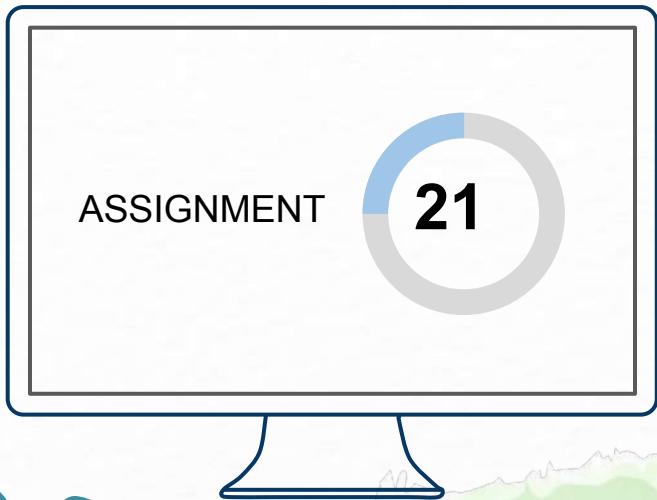
### Messages

Date	Topic	QoS
2023-05-06 10:32:36	/MSIBMyskill/p/mqtt	QoS: 0
2023-05-06 10:31:23	/MSIBMyskill/p/mqtt	QoS: 0
2023-05-06 10:31:17	/MSIBMyskill/p/mqtt	QoS: 0
2023-05-06 10:31:11	/MSIBMyskill/p/mqtt	QoS: 0
2023-05-06 10:31:05	/MSIBMyskill/p/mqtt	QoS: 0
2023-05-06 10:30:59	/MSIBMyskill/p/mqtt	QoS: 0

# THANKS



## A39 - AKBAR FADILA



Kampus  
Merdeka  
INDONESIA JAYA

# Tugas



- Silahkan cobakan git tadi atau update code yang terbaru dari repository ini
  - Lakukan fork ke repo/akun github pribadi
  - Lakukan clone dari fork repo tersebut
  - Lakukan merge ke master dan hapus branch feature/esp-wifi-codes setelah merge
  - Screenshot repository yang telah di merge dan di fork

<https://github.com/farizalemuda/myskill-iot.git>

#RintikKariImpian

- Lakukan fork ke repo/akun github pribadi

Code Issues Pull requests Actions Projects Security Insights

## Create a new fork

A *fork* is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. [View existing forks.](#)

---

Owner \* Repository name \*

 akbarfadila / myskill-iot 

By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.

Description (optional)

Contains source codes for MySkill IoT Hands-on.

---

Copy the `master` branch only  
Contribute back to farizalemuda/myskill-iot by adding your own branch. [Learn more.](#)

---

 You are creating a fork in your personal account.

---

**Create fork**

## ● hasil

github.com/akbarfadila/myskill-iot

Search or jump to... Pull requests Issues Codespaces Marketplace Explore

akbarfadila / myskill-iot Public forked from farizalemuda/myskill-iot

Code Pull requests Actions Projects Wiki Security Insights Settings

master 2 branches 0 tags Go to file Add file Code About

Your master branch isn't protected Protect this branch

This branch is up to date with farizalemuda/myskill-iot:master. Contribute Sync fork

farizalemuda Create sesi-17-rgb-led cfsca0 2 weeks ago 18 commits

sesi-10-adc-slide-switch1	Create sesi-10-adc-slide-switch1	3 weeks ago
sesi-10-adc-slide-switch2	Create sesi-10-adc-slide-switch2	3 weeks ago
sesi-10-ldr	Create sesi-10-ldr	3 weeks ago
sesi-11-pwm-buzzer	Create sesi-11-pwm-buzzer	3 weeks ago
cod-11-pwm-led	Create cod-11-pwm-led	3 weeks ago

About

Contains source codes for MySkill IoT Hands-on.

0 stars 0 watching 22 forks

Releases

No releases published Create a new release

Packages

No packages published Publish your first package

- Lakukan clone dari fork repo tersebut

The screenshot shows a Windows desktop environment with several open windows:

- File Explorer:** A window titled "myskill-iot" showing the contents of the folder at "C:\Users\akbar\myskill-iot\myskill-iot". The folder contains subfolders like ".git", "myskill-iot", and "sesi-6-serial-hello", along with various files named "sesi-8-active-high", "sesi-8-active-low", etc.
- Terminal:** A window titled "MINGW64:c/Users/akbar/myskill-iot/myskill-iot" displaying a command-line session. The user runs "git clone" to clone the repository from GitHub, then changes directory to "myskill-iot", lists branches, and switches to the "feature/esp-wifi-codes" branch. They then merge this branch into "master" and update their local repository.
- Word Document:** A Microsoft Word document titled "Document1" containing the text "20 items".

At the bottom of the screen, there is a navigation bar with icons for Notes, Comments, and other document functions, along with a status bar showing "Slide 3 of 7", "English (Australia)", and "Recovered".

- Lakukan merge ke master dan hapus branch feature/esp-wifi-codes setelah merge

The screenshot shows a Windows desktop environment with three windows open:

- File Explorer (Left):** Shows the directory structure under 'This PC > OS (C) > Users > akbar'. It lists various folders like Desktop, Videos, Music, oprek, JAV/A, workshop, HTML CSS JS, BAHASA C, CISCO, ZIP - RAR, PI, PHYTON, xampp, htdocs, images, and Electronics\_Parts\_Sales\_and\_Invent. A folder named 'myskill-iot' is selected.
- Terminal (Center):** A terminal window titled 'MINGW64:/c/Users/akbar/myskill-iot/myskill-iot'. It shows the following command history and output:
 

```
akbar@LAPTOP-2QON8946 MINGW64 ~/myskill-iot/myskill-iot (master)
$ git branch
feature/esp-wifi-codes
* master

akbar@LAPTOP-2QON8946 MINGW64 ~/myskill-iot/myskill-iot (master)
$ git merge feature/esp-wifi-codes
Updating cf5cbad..c732637
Fast-forward
  sesi-19-20/sesi-19/ap_and_stat/ap_and_stat.ino | 42 ++++++=====
.../sesi-19/check_wifi_state/check_wifi_state.ino | 54 ++++++=====
sesi-19-20/sesi-19/connect_wifi/connect_wifi.ino | 34 ++++++=====
.../sesi-19/get_ip_address/get_ip_address.ino | 27 ++++++=====
.../get_signal_strength/get_signal_strength.ino | 27 ++++++=====
sesi-19-20/sesi-19/scan_wifi/scan_wifi.ino | 41 ++++++=====
sesi-19-20/sesi-19/wifi_ap/wifi_ap.ino | 26 ++++++=====
.../sesi-20/disconnect_wifi/disconnect_wifi.ino | 37 ++++++=====
sesi-19-20/sesi-20/reconnect/reconnect.ino | 44 ++++++=====
sesi-19-20/sesi-20/set_hostname/set_hostname.ino | 32 ++++++=====
sesi-19-20/sesi-20/set_static_ip/set_static_ip.ino | 47 ++++++=====
sesi-19-20/sesi-20/wifi_multi/wifi_multi.ino | 63 ++++++=====
sesi-21/ble_server/ble_server.ino | 38 ++++++=====

13 files changed, 512 insertions(+)
create mode 100644 sesi-19-20/sesi-19/ap_and_stat/ap_and_stat.ino
create mode 100644 sesi-19-20/sesi-19/check_wifi_state/check_wifi_state.ino
create mode 100644 sesi-19-20/sesi-19/connect_wifi/connect_wifi.ino
create mode 100644 sesi-19-20/sesi-19/get_ip_address/get_ip_address.ino
create mode 100644 sesi-19-20/sesi-19/get_signal_strength/get_signal_strength.ino
create mode 100644 sesi-19-20/sesi-19/scan_wifi/scan_wifi.ino
create mode 100644 sesi-19-20/sesi-19/wifi_ap/wifi_ap.ino
create mode 100644 sesi-19-20/sesi-20/disconnect_wifi/disconnect_wifi.ino
create mode 100644 sesi-19-20/sesi-20/reconnect/reconnect.ino
create mode 100644 sesi-19-20/sesi-20/set_hostname/set_hostname.ino
create mode 100644 sesi-19-20/sesi-20/set_static_ip/set_static_ip.ino
create mode 100644 sesi-19-20/sesi-20/wifi_multi/wifi_multi.ino
create mode 100644 sesi-21/ble_server/ble_server.ino

akbar@LAPTOP-2QON8946 MINGW64 ~/myskill-iot/myskill-iot (master)
$ git push -u origin master
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/akbarfadila/myskill-iot.git
  cf5cbad..c732637  master -> master
branch 'master' set up to track 'origin/master'.
```
- File List (Right):** A window titled '20 items' showing a single yellow folder icon.

# Hasil Lakukan merge ke master

Searched or jump to... Pull requests Issues Codespaces Marketplace Explore

akbarfadila / myskill-iot Public  
forked from farizalemuda/myskill-iot

Code Pull requests Actions Projects Wiki Security Insights Settings

master 2 branches 0 tags Go to file Add file Code Protect this branch

Your master branch isn't protected Protect this branch

This branch is 5 commits ahead of farizalemuda:master. Contribute Sync fork

simabuaboy generic example of ble server c732637 5 days ago 23 commits

sesi-19-20 ini tambahan code dari sesi 19 ke 20 5 days ago

sesi-21/ble\_server generic example of ble server 5 days ago

sesi-10-adc-slide-switch1 Create sesi-10-adc-slide-switch1 3 weeks ago

sesi-10-adc-slide-switch2 Create sesi-10-adc-slide-switch2 3 weeks ago

sesi-10-ldr Create sesi-10-ldr 3 weeks ago

About

Contains source codes for MySkill IoT Hands-on.

0 stars 0 watching 22 forks

Releases

No releases published Create a new release

Packages

No packages published Publish your first package

# Hasil hapus branch feature/esp-wifi-codes setelah merge

github.com/akbarfadila/myskill-iot

Search or jump to... Pull requests Issues Codespaces Marketplace Explore

akbarfadila / myskill-iot Public forked from farizalemuda/myskill-ot

Code Pull requests Actions Projects Wiki Security Insights Settings

master 1 branch 0 tags

Switch branches/tags Find or create a branch...

Branches Tags default

✓ master View all branches sesi-21/ble\_server

sesi-10-adc-slide-switch1 Create sesi-10-adc-slide-switch1 3 weeks ago

sesi-10-adc-slide-switch2 Create sesi-10-adc-slide-switch2 3 weeks ago

sesi-10-ldr Create sesi-10-ldr 3 weeks ago

sesi-11-pwm-buzzer Create sesi-11-pwm-buzzer 3 weeks ago

sesi-11-pwm-led Create sesi-11-pwm-led 3 weeks ago

sesi-13-IR-Remote.ino Create sesi-13-IR-Remote.ino last month

sesi-14-imu.ino Create sesi-14-imu.ino 3 weeks ago

Go to file Add file < Code About

Contribute Sync fork

Contains source codes for MySkill IoT Hands-on.

0 stars 0 watching 22 forks

No releases published Create a new release

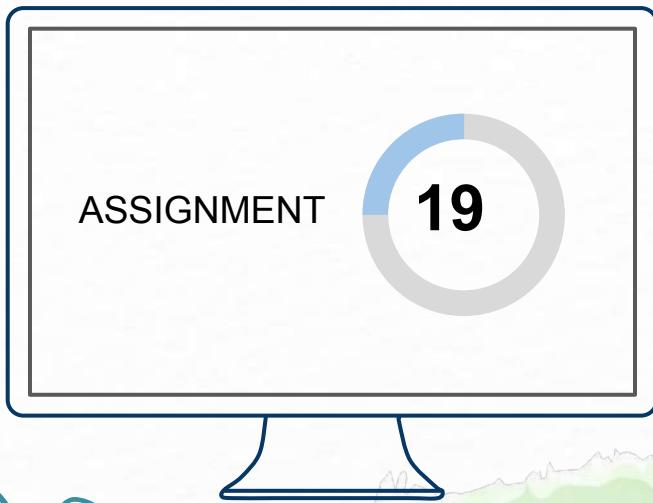
No packages published Publish your first package

Languages

# THANKS



## A39 - AKBAR FADILA



# Tugas



- Buatlah script untuk mendapatkan mac address dari esp32

Link

<https://wokwi.com/projects/361327947054162945>

wokwi.com/projects/361327947054162945

WOKWI SAVE SHARE tugas 12 akbar A39 Docs

sketch.ino • diagram.json libraries.txt Library Manager

Simulation

00:04.464 84%

```
1 void setup() { //Fungsi setup() merupakan fungsi yang dijalankan pertama kali ketika program dimulai.
2
3     Serial.begin(115200); //Pada baris ini, initialisasi komunikasi serial dilakukan dengan mengatur ke
4     esp_err_t ret = ESP_OK; //Variabel base_mac_addr merupakan array dengan 6 elemen bertipe da
5     ret = esp_efuse_mac_get_default(base_mac_addr); //Pada baris ini, fungsi esp_efuse_mac_get_default(
6
7     //Baris ini menampilkan alamat MAC yang telah dibaca pada monitor serial dengan format XX:XX:XX:XX
8     printf("MAC Address = %02x:%02x:%02x:%02x:%02x:%02x (ret: %d)\n",
9         base_mac_addr[0],
10        base_mac_addr[1],
11        base_mac_addr[2],
12        base_mac_addr[3],
13        base_mac_addr[4],
14        base_mac_addr[5],
15        ret
16    );
17 }
18
19 void loop() { //Fungsi loop() adalah fungsi yang dijalankan secara berulang-ulang setelah fungsi setu
20
21     delay(10);
22 }
23 }
```

ESP32

MAC Address = 24:0a:c4:00:01:10 (ret: 0)

## Program 2

<https://wokwi.com/projects/361335847950888961>

wokwi.com/projects/361335847950888961

WOKWI SAVE SHARE TUGAS 19 AKBAR TEST Docs A

sketch.ino • diagram.json • Library Manager

Simulation

00:12.116 65%

```
1 #include <WiFi.h>
2 #include <esp_wifi.h>
3 const char* ssid      = "Wokwi-GUEST";
4 const char* password  = "";
5
6 uint8_t newMACAddress[] = {0x32, 0xAE, 0xA4, 0xE7, 0x00, 0x66};
7
8 void setup() {
9   Serial.begin(115200);
10  Serial.print("ESP32 ");
11  Serial.println();
12  WiFi.begin(ssid, password, 6);
13  while (WiFi.status() != WL_CONNECTED) {
14    delay(500);
15    Serial.println("Connecting to WIFI....");
16  }
17  Serial.println("");
18  Serial.println("WiFi Connected");
19  Serial.print("IP address: ");
20  Serial.println(wifi.localIP());
21  WiFi.mode(WIFI_STA);
22  esp_wifi_set_mac(WIFI_IF_STA, &newMACAddress[0]);
23  Serial.print("ESP32 MAC Address: ");
24  Serial.println(wifi.macAddress());
25 }
26
27 void loop() {
28  delay(10);
29 }
```

ESP32

Connecting to WIFI....

Connecting to WIFI....

WiFi Connected

IP address: 10.10.0.2

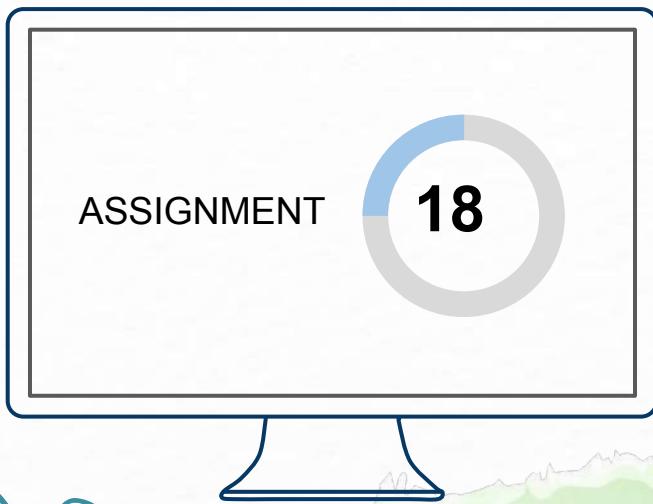
ESP32 MAC Address: 32:AE:A4:E7:0D:66

||/

# THANKS



## A39 - AKBAR FADILA



# Tugas

Buatlah aplikasi untuk melakukan koneksi wifi  
Di wokwi, lalu lakukan print waktu yang didapat  
Dari server NTP

Format output:

1. **GMT**
2. EPOCH (The Unix epoch (or Unix time or POSIX time or Unix timestamp))

Contoh:

1. Saturday, April 1, 2023 5:03:35 AM
2. 1680325459

[esp32 wifi wokwi](#)

Server NTP

#RintikKarinImpian

Link

<https://wokwi.com/projects/360972379951193089>

wokwi.com/projects/360972379951193089

WOKWI SAVE SHARE TUGAS 18 AKBAR FADILA Docs A

sketch.ino • diagram.json libraries.txt Library Manager

Simulation

01:18.594 91%

```
1 //A39 -AKBAR FADILA
2 #include <WiFi.h> //mengimport library WiFi untuk digunakan dalam program
3 #include <time.h> //engimport library time untuk digunakan dalam program
4 //wifi
5 const char* ssid      = "Wokwi-GUEST"; //deklarasi variabel untuk menyimpan nama jaringan WiFi
6 const char* password = ""; //deklarasi variabel untuk menyimpan password jaringan WiFi
7 //ntp
8 const char* ntpServer = "pool.ntp.org"; //deklarasi variabel untuk menyimpan alamat server NTP
9 const long gmtOffset_sec = 21600; //deklarasi variabel untuk menyimpan perbedaan waktu GMT (dalam detik)
10 const int  daylightOffset_sec = 3600; //deklarasi variabel untuk menyimpan perbedaan waktu musim panas (dalam detik)
11 //epochTime
12 unsigned long epochTime; //deklarasi variabel untuk menyimpan perbedaan waktu musim panas (dalam detik)
13 unsigned long getTime() { //fungsi untuk mendapatkan waktu epoch dari server NTP
14     time_t now;
15     struct tm timeinfo;
16     if (!getLocalTime(&timeinfo)) {
17         Serial.println("Failed to obtain time");
18         return(0);
19     }
20     time(&now);
21     return now;
22 }
23
24 // fungsi yang pertama kali dijalankan ketika program di-run, berisi koneksi ke WiFi dan inisialisasi
25 void setup(){
26     Serial.begin(115200);
27
28 // Connect to Wi-Fi
29 Serial.print("Connecting to ");
30 Serial.println(ssid);
31 WiFi.begin(ssid, password);
32 while (WiFi.status() != WL_CONNECTED) {
33     delay(500);
34     Serial.print(".");
35 }
36 Serial.println("");
37 Serial.println("WiFi connected.");
```

ESP32

Connecting to Wokwi-GUEST  
.....  
WiFi connected.  
Monday, April 03 2023 15:01:31  
Epoch Time: 1680508891

# Codingan

WOKWI SAVE SHARE ♥ TUGAS 18 AKBAR FADILA

sketch.ino diagram.json libraries.txt Library Manager

```
1 //A39 -AKBAR FADILA
2 #include <WiFi.h> //mengimport library WiFi untuk digunakan dalam program
3 #include "time.h"/engimport library time untuk digunakan dalam program
4 //wifi
5 const char* ssid      = "Wokwi-GUEST";//deklarasi variabel untuk menyimpan nama jaringan WiFi
6 const char* password = "";//deklarasi variabel untuk menyimpan password jaringan WiFi
7 //ntp
8 const char* ntpServer = "pool.ntp.org";//deklarasi variabel untuk menyimpan alamat server NTP
9 const long gmtOffset_sec = 21600; //deklarasi variabel untuk menyimpan perbedaan waktu GMT (dalam detik)
10 const int daylightOffset_sec = 3600; //deklarasi variabel untuk menyimpan perbedaan waktu musim panas (dalam detik)
11 //epochTime
12 unsigned long epochTime; //deklarasi variabel untuk menyimpan perbedaan waktu musim panas (dalam detik)
13 unsigned long getTime() {//fungsi untuk mendapatkan waktu epoch dari server NTP
14     time_now;
15     struct tm timeinfo;
16     if (!getLocalTime(&timeinfo)) {
17         Serial.println("Failed to obtain time");
18         return(0);
19     }
20     time(&now);
21     return now;
22 }
23
24 // fungsi yang pertama kali dijalankan ketika program di-run, berisi koneksi ke WiFi dan inisialisasi waktu dari server NTP
25 void setup(){
26     Serial.begin(115200);
27
28     // Connect to Wi-Fi
29     Serial.print("Connecting to ");
30     Serial.println(ssid);
31     WiFi.begin(ssid, password);
32     while (WiFi.status() != WL_CONNECTED) {
33         delay(500);
34         Serial.print(".");
35     }
36     Serial.println("");
37     Serial.println("Wifi connected.");
```

sketch.ino

diagram.json

libraries.txt

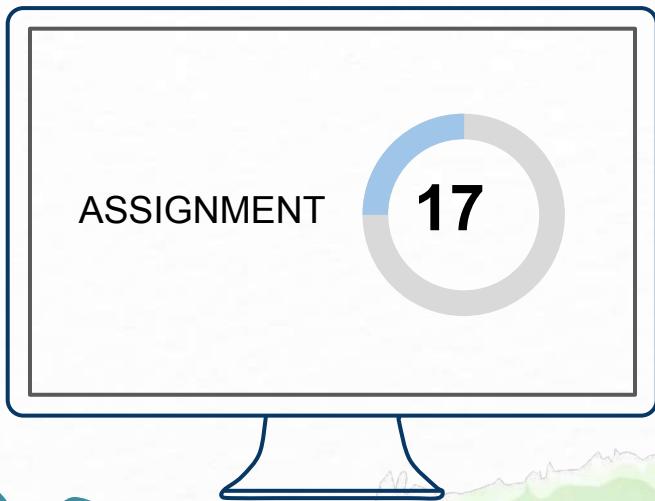
Library Manager

```
35  }
36  Serial.println("");
37  Serial.println("WiFi connected.");
38
39 // Init and get the time
40 configTime(gmtOffset_sec, daylightOffset_sec, ntpServer);
41 }
42
43 //fungsi yang dijalankan secara terus-menerus setelah fungsi setup(), berisi pemanggilan fungsi printLocalTime() dan getTime()
44 void loop(){
45  printLocalTime();
46  epochTime = getTime();
47  Serial.print("Epoch Time: ");
48  Serial.println(epochTime);
49  Serial.println();
50  delay(1000);
51 }
52
53 //fungsi untuk mencetak waktu lokal ke Serial Monitor dalam format yang ditentukan
54 void printLocalTime(){
55  struct tm timeinfo;
56  if(!getLocalTime(&timeinfo)){
57   Serial.println("Failed to obtain time");
58   return;
59  }
60  Serial.println(&timeinfo, "%A, %B %d %Y %H:%M:%S");
61 }
62 }
```

# THANKS



# A39 – AKBAR FADILA



## Tugas



- Mohon kerjakan Mini Task 1 & Mini Task 2, sertakan evidence (skrinsut) penggerjaannya
- Mohon lakukan praktikum 2, namun, ubah menggunakan basis mikrokontroler ESP32

# Mini Task 1

## DHT22 & Kipas

### Objektif

Hidupkan Kipas (diwakili dengan LED) jika suhu DHT22 lebih dari  $30^{\circ}$

### Pra-Syarat

Memiliki akun Wokwi



MySkill MSIA

Kampus  
Merdeka

<https://wokwi.com/projects/360963730108318721>

wokwi.com/projects/360963730108318721

WOKWI SAVE SHARE akbar sesi tugas sesi 17

sketch.ino • diagram.json • libraries.txt Library Manager Docs A

```
1 //Pertama-tama, kita memasukkan library DHTesp untuk dapat membaca data dari sensor DHT22.
2 #include "DHTesp.h" //proses include library
3 //Selanjutnya, kita mendeklarasikan pin DHT22 pada pin 15 dan objek sensor dengan tipe DHT.
4 const int DHTPIN = 15;
5 DHTesp dht;
6 //kita mendeklarasikan pin untuk kipas (dalam hal ini dimaklumi dengan LED) pada pin 19.
7 const int pinRelay = 19;
8
9 //kita melakukan inisialisasi serial monitor dan pin kipas sebagai output. Selain itu, kita juga mcl
10 void setup(){
11   Serial.begin(9600);
12   Serial.println("Assignment 17");
13   Serial.println("A39-AKBAR FADILA");
14   pinMode(pinRelay, OUTPUT);
15   dht.setup(DHTPIN, DHTesp::DHT22);
16 }
17
18 //kita mendapatkan data suhu dan kelembaban dari sensor DHT22. Kemudian, kita memeriksa apakah suhu
19 void loop(){
20   TempAndHumidity data = dht.getTempAndHumidity();
21   float temp = data.temperature;
22   if(temp >= 30){
23     digitalWrite(pinRelay, HIGH);
24     Serial.print("Temperature: ");
25     Serial.print(temp);
26     Serial.print("C ");
27     Serial.println("KIPAS HIDUP");
28     delay(500);
29   }else{
30     digitalWrite(pinRelay, LOW);
31     Serial.print("Temperature: ");
32     Serial.print(temp);
33     Serial.print("C ");
34     Serial.println("KIPAS MATI");
35     delay(500);
36   }
37 }
```

Simulation

Assignment 17  
A39-AKBAR FADILA  
Temperature: 11.10C KIPAS MATI  
Temperature: 11.10C KIPAS MATI  
Temperature: 11.10C KIPAS MATI  
Temperature: 11.10C KIPAS MATI  
Temperature: 48.30C KIPAS HIDUP  
Temperature: 48.30C KIPAS HIDUP

# Mini Task 2

## RGB LED & Potensiometer



MySkill

MSIB

Kampus  
Merdeka

### Objektif

Atur RGB LED berbasis Input Potensiometer Bulat

### Pra-Syarat

Memiliki akun Wokwi

#RintikKariImpian

WOKWI SAVE SHARE TUGAS 17 AKBAR FADILA Docs A

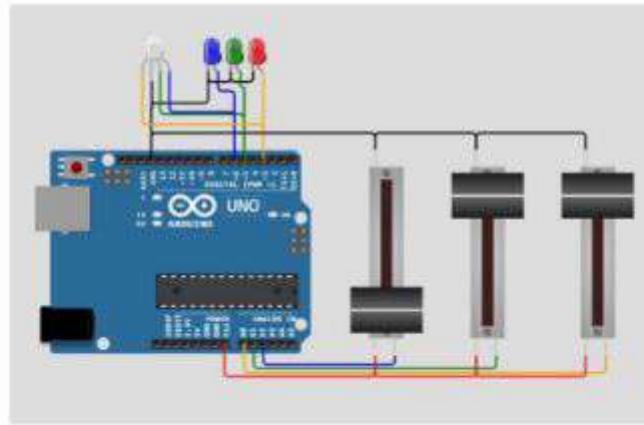
sketch.ino • diagram.json Library Manager

Simulation 01:12.114 74%

```
1 //Pertama-tama, kita mendefinisikan pin untuk masing-masing warna pada LED RGB dan pin untuk masing-
2 const int pinR = 3;
3 const int pinG = 5;
4 const int pinB = 6;
5 const int potR = A0;
6 const int potG = A1;
7 const int potB = A2;
8
9 //kita mengatur mode pin-pin yang akan digunakan. Pin untuk setiap warna LED diatur sebagai output,
10 void setup() {
11   pinMode(pinR, OUTPUT);
12   pinMode(pinG, OUTPUT);
13   pinMode(pinB, OUTPUT);
14   pinMode(potR, INPUT);
15   pinMode(potG, INPUT);
16   pinMode(potB, INPUT);
17 }
18
19 // fungsi readPot() yang digunakan untuk membaca nilai dari potensiometer pada pin tertentu. Fungsi
20 int readPot(int pin) {
21   return map(analogRead(pin), 0, 1023, 0, 255);
22 }
23
24 //kita membaca nilai dari masing-masing potensiometer dan mengubahnya menjadi nilai PWM (pulse-width
25 void loop() {
26   analogWrite(pinR, readPot(potR));
27   analogWrite(pinG, readPot(potG));
28   analogWrite(pinB, readPot(potB));
29 }
```

# Praktikum 2

## 3. Wiring



#RencanaImpian

wokwi.com/projects/360969868551635969

WOKWI SAVE SHARE tugas 17 akbar A39 Docs A

sketch.ino • diagram.json • Library Manager

```
1 //Mendefinisikan pin untuk masing-masing warna pada LED RGB dan pin untuk masing-masing potensiometer
2 const int pinR = 14;
3 const int ping = 12;
4 const int pinB = 13;
5 const int potR = 4;
6 const int potG = 2;
7 const int potB = 15;
8 void setup() {
9     //Mengatur mode pin-pin yang akan digunakan. Pin untuk setiap warna LED diatur sebagai output, sedangkan pin untuk potensiometer diatur sebagai input.
10    pinMode(pinR, OUTPUT);
11    pinMode(ping, OUTPUT);
12    pinMode(pinB, OUTPUT);
13    pinMode(potR, INPUT);
14    pinMode(potG, INPUT);
15    pinMode(potB, INPUT);
16 }
17 //fungsi readPot() yang digunakan untuk membaca nilai dari potensiometer pada pin tertentu. Fungsi ini mengembalikan nilai antara 0 dan 1023.
18 int readPot(int pin) {
19     return map(analogRead(pin), 0,1023, 0, 255);
20 }
21
22 //Membaca nilai dari masing-masing potensiometer dan mengubahnya menjadi nilai PWM (pulse-width modulation)
23 void loop() {
24     analogWrite(pinR, readPot(potR));
25     analogWrite(ping, readPot(potG));
26     analogWrite(pinB, readPot(potB));
27 }
```

Simulation

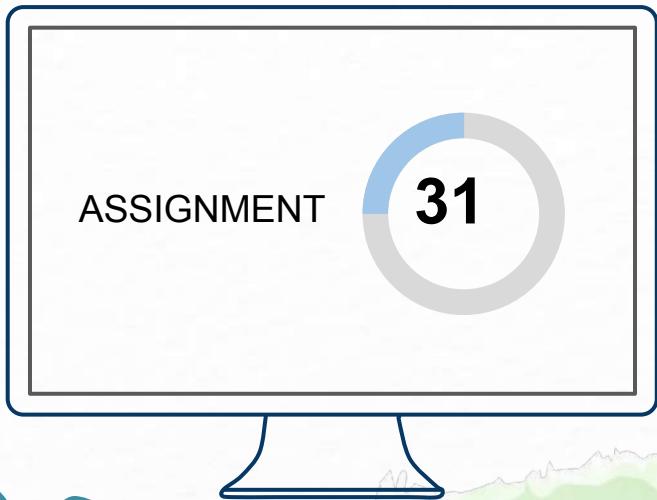
00:07.666 24%

The screenshot shows a Wokwi project interface. On the left, the code for a sketch in Arduino IDE is displayed, defining pins for an ESP32 and setting up an RGB LED driver and three potentiometers. The right side features a simulation window showing the physical circuit. The circuit consists of an ESP32 microcontroller at the center, connected to four RGB LEDs (red, green, blue, yellow) via a NE230 driver IC. Three potentiometers are connected to the microcontroller's analog pins. A 5V power source and ground connections are also present.

# THANKS



## A39 - AKBAR FADILA



Kampus  
Merdeka  
INDONESIA JAYA

# Tugas



- Buat code untuk mendapatkan data telemetry Pada wokwi

#RintikairImpian

wokwi.com/projects/364816827598593025

WOKWI SAVE SHARE tugas 31 A39 AKBAR FADILA Perbarui Docs

sketch.ino diagram.json libraries.txt Library Manager

Simulation 00:12.715 51%

```
1 //A39 AKBAR FADILA assignment 31
2
3 //mengimpor library yang diperlukan, yaitu library WiFi untuk mengatur koneksi WiFi, library ThingsBoard untuk berkomunikasi dengan server Thing
4 #include <WiFi.h>
5 #include "ThingsBoard.h"
6 #include "DHTesp.h"
7
8 //mendefinisikan SSID (nama jaringan WiFi) dan password WiFi yang akan digunakan untuk koneksi.
9 #define WIFI_SSID "Wokwi-GUEST"
10 #define WIFI_PASSWORD ""
11
12 //mendefinisikan token akses untuk perangkat di ThingsBoard (TOKEN) dan alamat server ThingsBoard (THINGSBOARD_SERVER).
13 #define TOKEN "TUGASKE31 AKBAR FADILA"
14 #define THINGSBOARD_SERVER "thingsboard.cloud"
15
16 //mendefinisikan pin yang digunakan untuk sensor DHT dan membuat objek dhtSensor dari kelas DHTesp.
17 const int DHT_PIN = 15;
18 DHTesp dhtSensor;
19
20 // mendefinisikan objek WiFiclient (espClient) untuk mengatur koneksi WiFi dan objek ThingsBoard (tb) untuk berkomunikasi dengan server ThingsB
21 WiFiclient espClient;
22 ThingsBoard tb(espClient);
23
24 //mendefinisikan variabel status dan menginisialisasinya dengan nilai WL_IDLE_STATUS. Variabel ini akan digunakan untuk memantau status koneksi
25 int status = WL_IDLE_STATUS;
26
27 //Ini adalah fungsi untuk menghubungkan ESP32 ke jaringan WiFi. Fungsi ini akan mencoba menghubungkan ESP32 ke jaringan WiFi yang ditentukan da
28 void connect()
29 {
30     Serial.println("Connecting to AP ...");
31
32     WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
33     while (WiFi.status() != WL_CONNECTED) {
34
35         delay(500);
36         Serial.print(".");
37     }
}
```

ESP32

DHT22

Connecting to AP ...
....Connected to AP
Connecting to: thingsboard.cloud with token TUGASKE31
AKBAR FADILA
Sending data...
Temperature : 9.30 Humidity : 83.00
Sending data...

# Device Groups

ThingsBoard Cloud Platform

Assignment 31 (A39 A...)

Assignment 31 (A39 A...)

TUGAS 28-A39 AKBAR FADILA

Devices-A39 AKBAR F...

All

Created time ↓ Name Device profile Label

2023-05-16 01:04:24 ASSIGNMENT 31 default

Items per page: 10 1 - 1 of 1 < > >>

Perbarui :

WOKWI

Simulation

Code

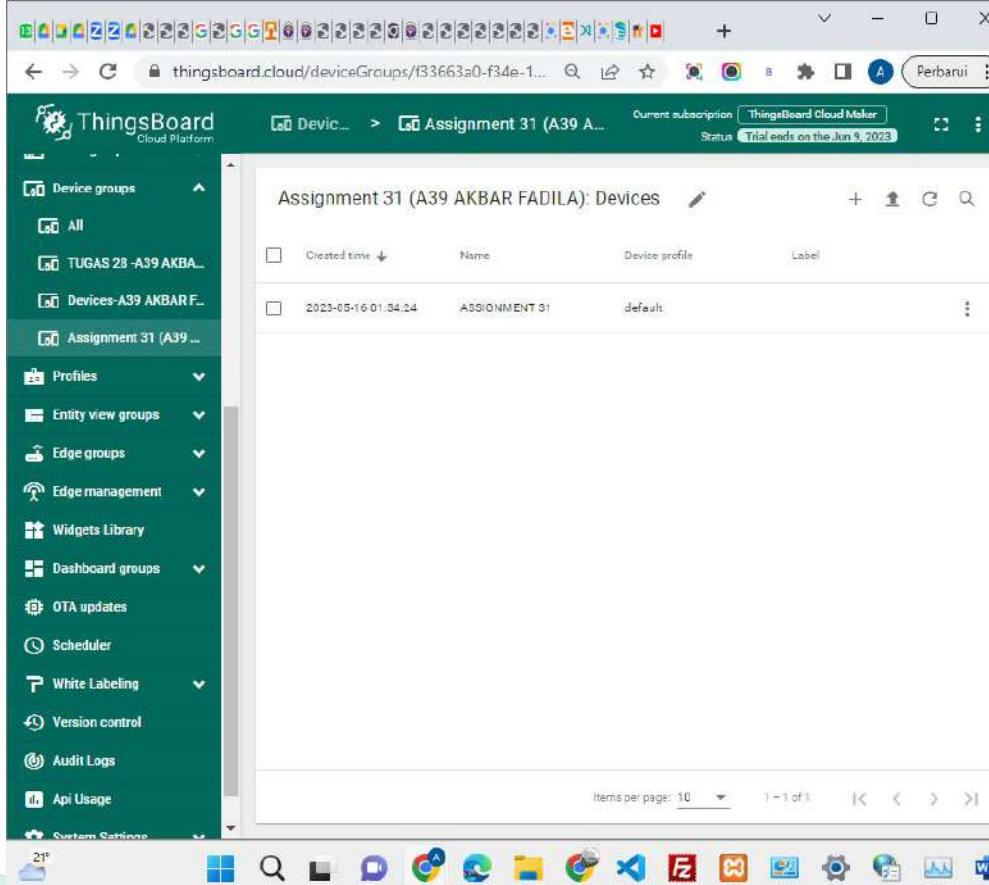
01:31:808 98%

ESP32

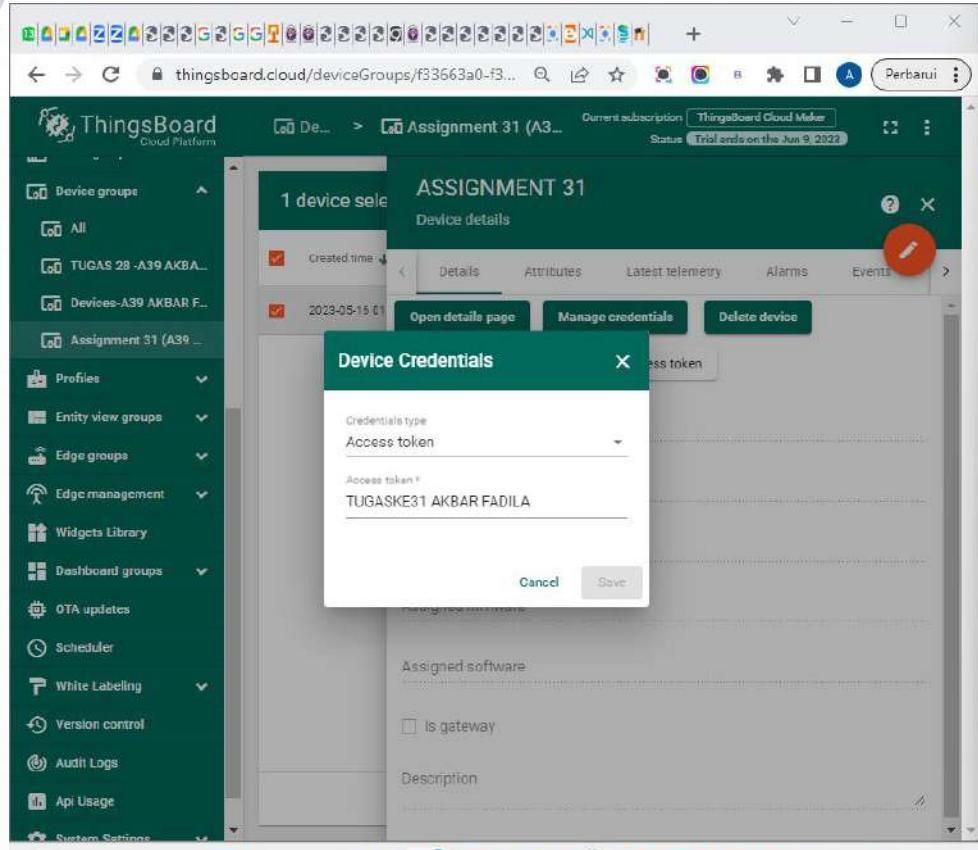
DHT22

Temperature : 9.30 Humidity : 83.00  
Sending data...  
Temperature : 9.30 Humidity : 83.00  
Sending data...  
Temperature : 9.30 Humidity : 83.00  
Sending data...  
Temperature : 9.30 Humidity : 83.00

2:41 AM 16-May-23



# Device credentials Access Token



ThingsBoard Cloud Platform

Assignment 31 (A39)

ASSIGNMENT 31

Device details

Created time: 2023-05-16 01

Open details page Manage credentials Delete device

Device Credentials

Credentials type: Access token

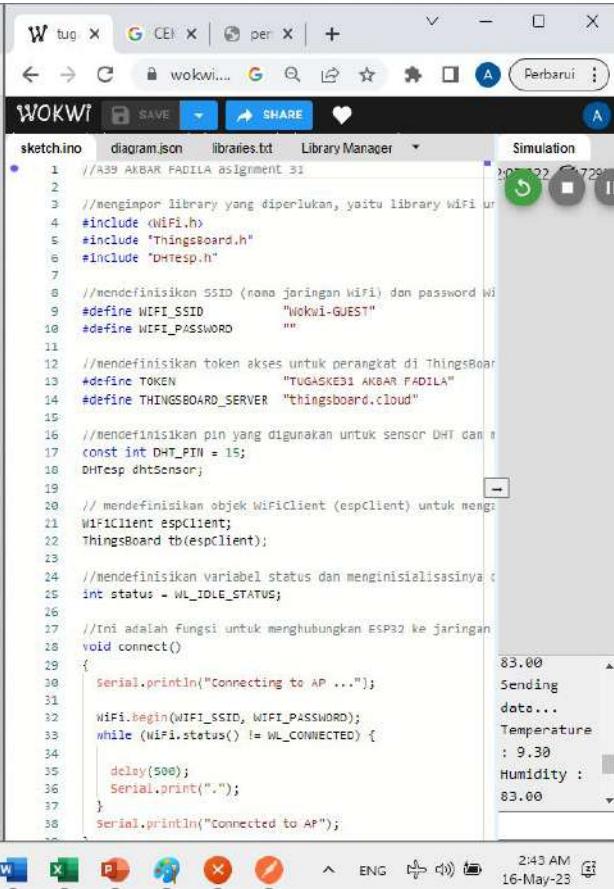
Access token: TUGASKE31 AKBAR FADILA

Assigned software

Is gateway:

Description:

Cancel Save



WOKWI

sketch.ino diagram.json libraries.txt Library Manager Simulation

```
//A39 AKBAR FADILA Assignment 31
//mengimport library yang diperlukan, yaitu library WiFi untuk koneksi ke jaringan WiFi
#include <WiFi.h>
#include "ThingsBoard.h"
#include "DHTesp.h"

//mendefinisikan SSID (nama jaringan WiFi) dan password WiFi
#define WIFI_SSID "Wokwi-GUEST"
#define WIFI_PASSWORD ""

//mendefinisikan token akses untuk perangkat di ThingsBoard
#define TOKEN "TUGASKE31 AKBAR FADILA"
#define THINGSBOARD_SERVER "thingsboard.cloud"

//mendefinisikan pin yang digunakan untuk sensor DHT dan LED
const int DHT_PIN = 15;
DHTesp dhtSensor;

// mendefinisikan objek WiFiClient (espclient) untuk menghubungkan ESP32 ke jaringan WiFi
WiFiClient espclient;
ThingsBoard tb(espclient);

//mendefinisikan variabel status dan menginisialisasinya dengan nilai WL_IDLE_STATUS
int status = WL_IDLE_STATUS;

//ini adalah fungsi untuk menghubungkan ESP32 ke jaringan WiFi
void connect()
{
    Serial.println("Connecting to AP ...");
    WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println("Connected to AP");
}

void setup() {
    // Set pins for DHT sensor
    dhtSensor.begin(DHT_PIN);
}

void loop() {
    float temperature = dhtSensor.readTemperature();
    float humidity = dhtSensor.readHumidity();

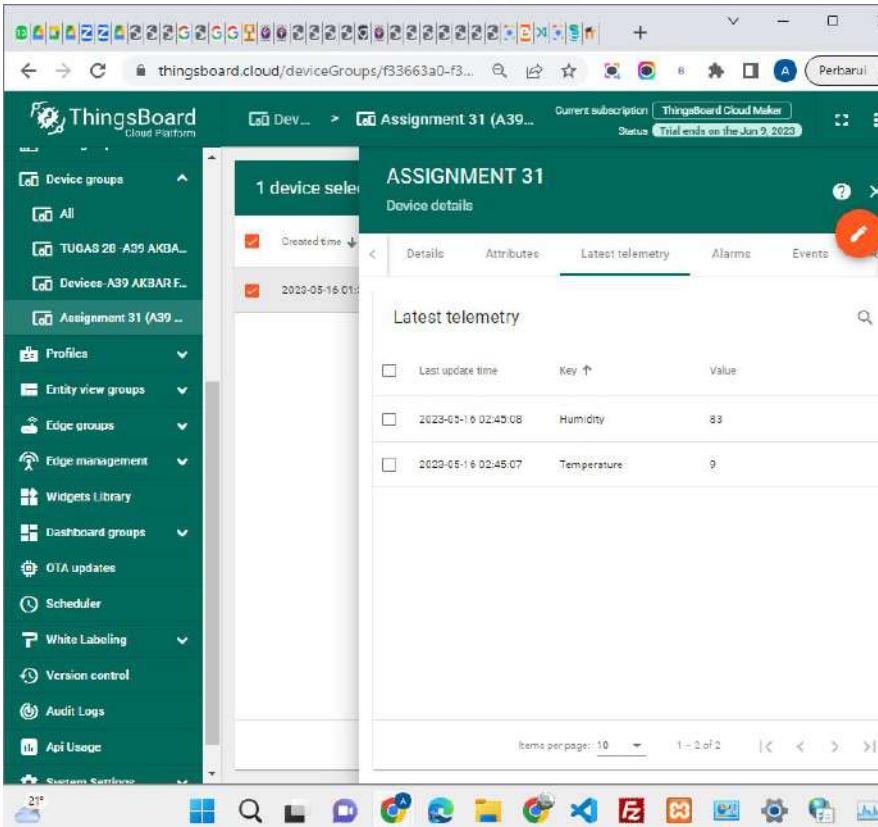
    if (isnan(temperature) || isnan(humidity)) {
        Serial.println("Failed to read from DHT sensor!");
        return;
    }

    // Print data to serial monitor
    Serial.print("Temperature : ");
    Serial.print(temperature);
    Serial.print("Humidity : ");
    Serial.print(humidity);
    Serial.println();
}
```

83.00  
Sending data...  
Temperature : 9.30  
Humidity : 83.00

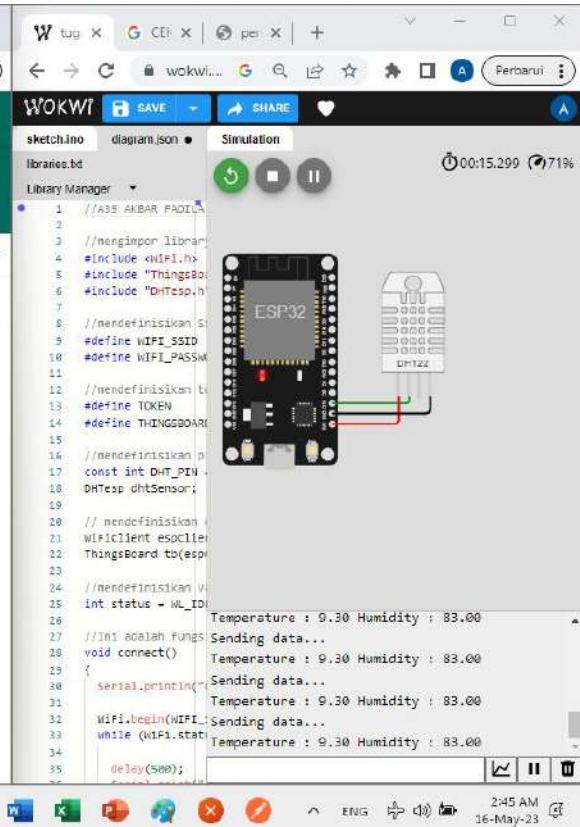
2:43 AM 16-May-23

# Data Latest Telemetry Wokwi



The screenshot shows the ThingsBoard Cloud Platform interface. On the left, the sidebar lists various sections like Device groups, Entity view groups, Edge management, and Widgets Library. In the center, a modal window titled "ASSIGNMENT 31" displays "1 device selected". Under "Device details", it shows "Created time: 2023-05-16 01:15:29". Below this, the "Latest telemetry" tab is active, showing a table with three rows of data:

Last update time	Key ↑	Value
2023-05-16 02:45:08	Humidity	83
2023-05-16 02:45:07	Temperature	9



The right side of the screen shows the Wokwi IDE. At the top, it says "WOKWI" and has tabs for "sketch.ino", "diagram.json", and "Simulation". The "Simulation" tab is active, showing a live feed of data. The code in "sketch.ino" is as follows:

```
//ASS AKBAR PADILAH
//ngeimport library
#include <WiFi.h>
#include "DHTesp.h"
//mendefinisikan S
#define WIFI_SSID
#define WIFI_PASSWD
//mendefinisikan TOKEN
#define TOKEN
#define THINGSBOARD_ID
//mendefinisikan pin
const int DHT_PIN = 4;
DHTesp dhtSensor;
// mendefinisikan WiFiClient
WiFiClient espClient;
ThingsBoard tb(espClient);
//mendefinisikan variabel
int status = WL_IDLE_STATUS;
Temperature : 9.30 Humidity : 83.00
//ini adalah fungsi
void connect()
{
    serial.println("Connecting to WiFi...");
    WiFi.begin(WIFI_SSID, WIFI_PASSWD);
    while (WiFi.status() != WL_CONNECTED)
        delay(500);
    serial.println("Connected to WiFi!");
    WiFiClient espClient;
    ThingsBoard tb(espClient);
    tb.begin();
    tb.setInterval(10000);
    tb.onDataReceived(onDataReceived);
}
```

The simulation window shows a DHT22 sensor connected to an ESP32 board. The live feed displays the temperature and humidity data from the sensor.

# Add to dashboard

The image shows a Windows desktop with two browser windows open. The left window is the ThingsBoard Cloud Platform interface, specifically the 'Device groups' section. A modal dialog titled '1 device selected' is open over a device named 'Assignment 31'. Inside the modal, under the 'Latest telemetry' tab, there is a button labeled 'Add to dashboard'. The right window is a Wokwi simulation environment. It displays a schematic of an ESP32 microcontroller connected to a DHT22 temperature and humidity sensor. The Wokwi code editor shows the following sketch:

```
//A39 AKBAR RADIAN
//Importing libraries
#include <WiFi.h>
#include "ThingBoard.h"
#include "DHTesp.h"

//Defining WiFi_SSID
#define WiFi_SSID "WIFI_SSID"
#define WiFi_PASSWORD "WIFI_PASSWORD"

//Defining TOKEN
#define TOKEN "THINGBOARD_TOKEN"

//Defining pins
const int DHT_PDN = 2;
DHTesp dhtSensor;

//Defining variables
int status = WL_IDLE_STATUS;
Temperature : 9.30 Humidity : 83.00
//This is a function
void connect()
{
    Serial.println("Connecting to WiFi");
    WiFi.begin(WiFi_SSID, WiFi_PASSWORD);
    while (WiFi.status() != WL_CONNECTED)
    {
        delay(500);
        Serial.print(".");
    }
    Serial.println("Connected to WiFi");
}

//Defining variables
int status = WL_IDLE_STATUS;
Temperature : 9.30 Humidity : 83.00
//This is a function
void connect()
{
    Serial.println("Connecting to WiFi");
    WiFi.begin(WiFi_SSID, WiFi_PASSWORD);
    while (WiFi.status() != WL_CONNECTED)
    {
        delay(500);
        Serial.print(".");
    }
    Serial.println("Connected to WiFi");
}
```

# Dashboard

thingsboard.cloud/dashboards/059b2710-f354-11ed-97ca-e57c88951dd5?edit=true

Perbarui

ThingsBoard Cloud Platform

TUGAS 31 (A39-AKBAR FADILA)

Current subscription: ThingsBoard Cloud Maker

Status: Trial ends on the Jun 9, 2023

Albar Fadila Tenant administrator

Home Plan and billing Solution templates NEW Rule chains Data converters Integrations Roles Customers hierarchy User groups Customer groups Asset groups Device groups Profiles Entity view groups Edge groups Edge management Widgets Library Dashboard groups OTA updates

TUGAS 31 (A39-AKBAR FADILA)

2023-05-16 02:53:38

Humidity: 83 Temperature: 9

avg 83 9

Humidity

Temperature

083

Timeseries table

Realtime - last minute

Timestamp Humidity Temperature

2023-05-16 02:52:05 83.

21°

Powered by Thingsboard v3.4.3PAAS

2:54 AM 16-May-23

# THANKS

