

---

---

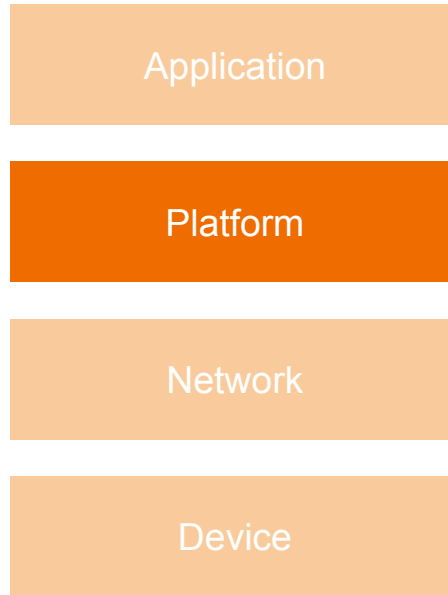
# IoT Platform

— Fariz Alemuda —

---

---

# IoT Stack



# Apa itu IoT Platform?

IoT Platform merupakan jembatan antara Device ke Application

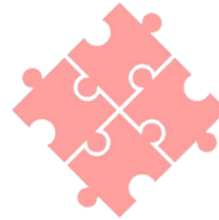
IoT Platform memiliki fitur-fitur dan fungsi umum seperti:



Secure



Device Management



Interoperability

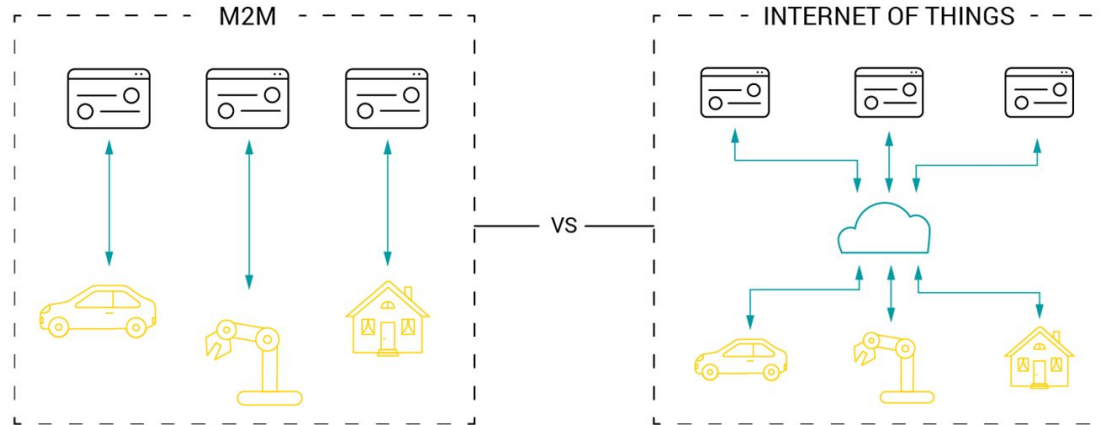


Data Management

# Mengapa IoT Platform dibutuhkan?



# IoT Platform vs M2M

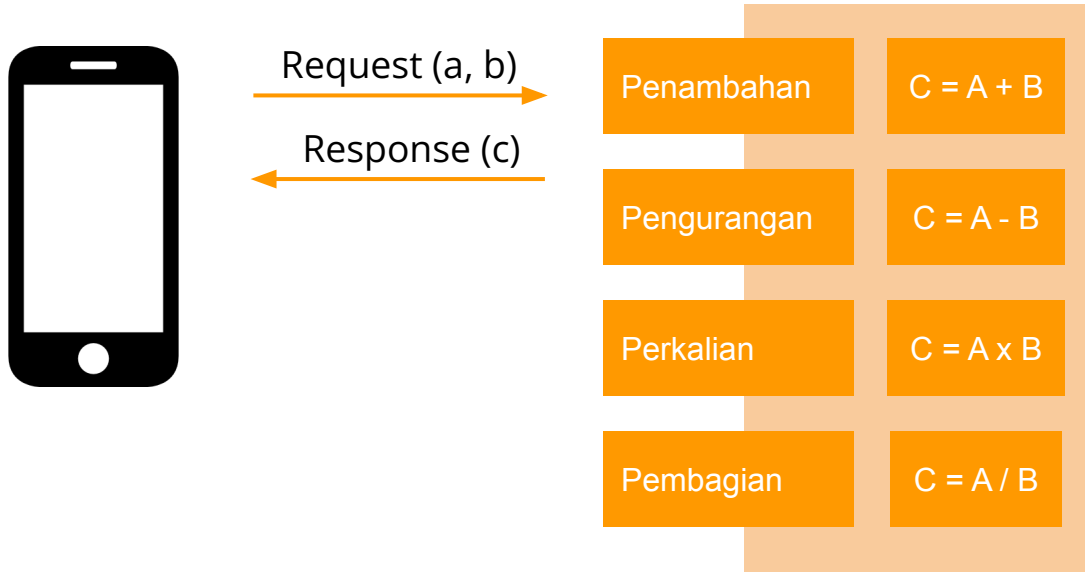


IoT memiliki penekanan pada sharing data agar menghasilkan nilai tambah kepada pelanggan. Kombinasi data dari seluruh komponen memungkinkan system yang lebih efektif dan efisien.

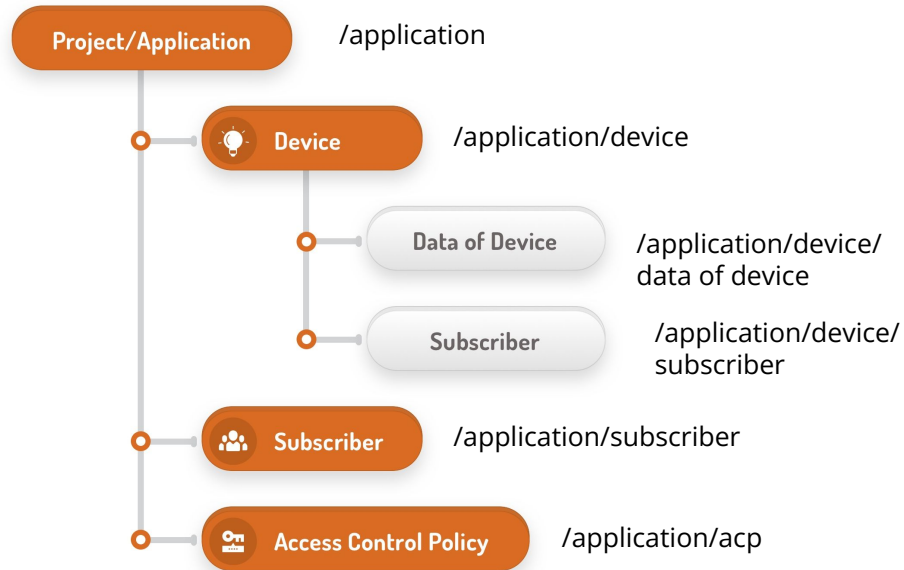
# IoT Platform Interfaces



# API (Application Programmable Interface)



# HTTP API Antares




- Application
  - Retrieve
- Device
  - Create
  - Retrieve
  - Update
  - Delete
- Data of Device
  - Create
  - Retrieve
- Subscriber
  - Create
  - Retrieve
  - Update
  - Delete
- Grouping
  - Create
  - Retrieve



# Registrasi Akun

# Register Account

 **ANTARES**

## Login

Sign In to your account

@

🔒

Login


[Forgot password?](#)

## Sign up

Are you in need of an enabler for your smart home, smart city, industrial IoT, etc.? Connect all your IoT Devices to the real IoT Platform, ANTARES.

Register Now!

# Register Account

 **ANTARES**

## Register

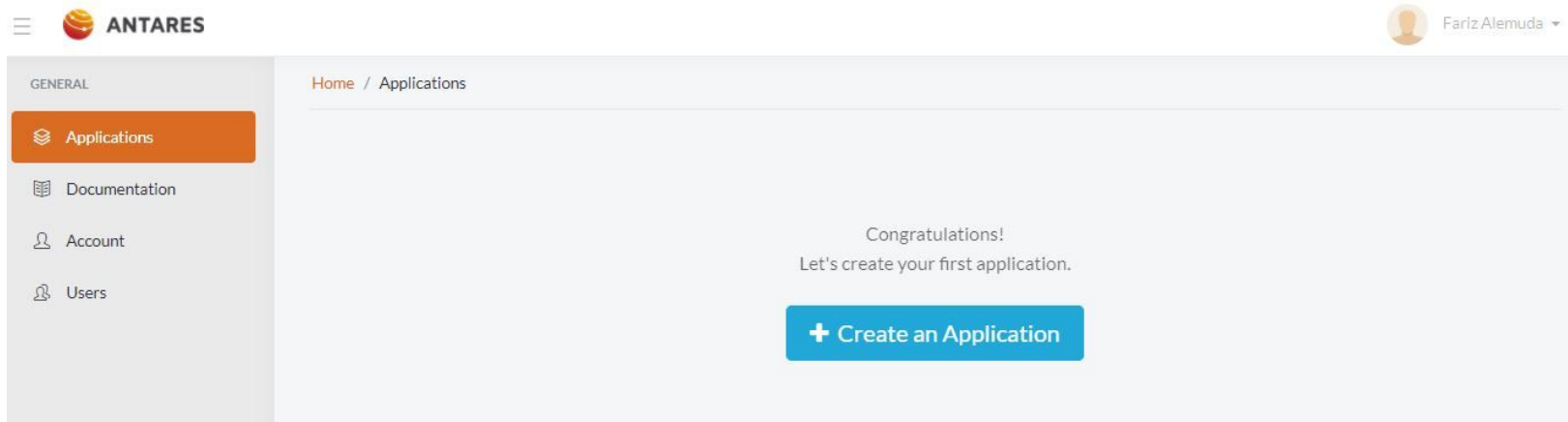
Create your account

Create Account

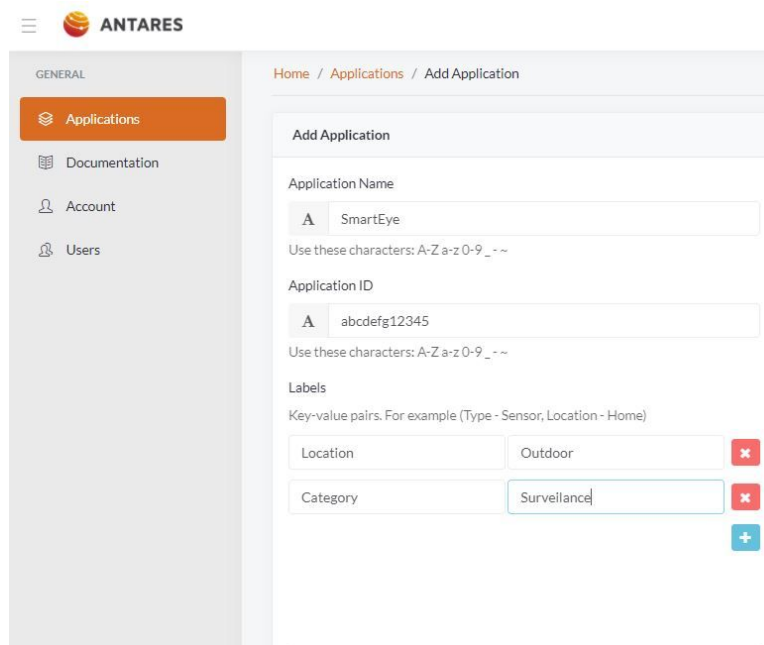
Already have an account? [Login](#)

# Provisioning

# Membuat Application



# Membuat Application



The screenshot displays the ANTARES web application interface. On the left is a sidebar with a 'GENERAL' section containing links for 'Applications' (highlighted in orange), 'Documentation', 'Account', and 'Users'. The main content area shows a breadcrumb trail 'Home / Applications / Add Application' and a form titled 'Add Application'. The form includes two text input fields: 'Application Name' with the value 'SmartEye' and 'Application ID' with the value 'abcdefg12345'. Below these fields are instructions: 'Use these characters: A-Z a-z 0-9 \_ - ~'. The 'Labels' section contains a description 'Key-value pairs. For example (Type - Sensor; Location - Home)' and two input pairs: 'Location' with the value 'Outdoor' and 'Category' with the value 'Surveillance'. Each input pair has a red 'X' button to its right, and a blue '+' button is located at the bottom right of the labels section.

ANTARES

GENERAL

- Applications
- Documentation
- Account
- Users

Home / Applications / Add Application

Add Application

Application Name

A SmartEye

Use these characters: A-Z a-z 0-9 \_ - ~

Application ID

A abcdefg12345

Use these characters: A-Z a-z 0-9 \_ - ~

Labels

Key-value pairs. For example (Type - Sensor; Location - Home)

Location Outdoor X


Category Surveillance X

+

# Membuat Application

The screenshot displays the ANTARES web application interface. At the top left, there is a menu icon and the ANTARES logo. The top right shows a user profile for 'Fariz Alemuda'. The left sidebar contains a 'GENERAL' section with links to 'Applications', 'Documentation', 'Account', and 'Users'. The main content area is titled 'Home / Applications / SmartEye'. It features a 'SmartEye PROJECT' header with 'Subscribe', 'User Access', and a trash icon. Below this, a text box explains that users can manage devices, subscribe, and manage access keys. Two URI types are listed: 'Hierarchical URI' (https://platform.antares.id:8443/~/-/antares-cse/antares-id/SmartEye) and 'Non-hierarchical URI' (https://platform.antares.id:8443/~/-/antares-cse/CAE840214314), each with a copy icon. Below the URIs, there are two tags: 'Location Outdoor' and 'Category Surveillance'. A link 'How to Create Device via API' is also present. At the bottom, there is a search bar for devices and a '+ Add Device' button. The footer indicates 'Powered by Telkom Indonesia'.

# Menambahkan Device

☰  **ANTARES**

Home / Applications / SmartEye / Add Device

Fariz Alemuda ▾

GENERAL

- Applications
- Documentation
- Account
- Users

**Add Device**


Name

A SmartEye1

+ Add



# Menambahkan Device

☰  **ANTARES**

Home / Applications / SmartEye / Add Device

Fariz Alemuda ▾

GENERAL

- Applications
- Documentation
- Account
- Users

**Add Device**

Name

A SmartEye1

+ Add

# Menambahkan Device

Add Device

Device SmartEye1 successfully added to SmartEye

# Menambahkan Device

The screenshot displays the ANTARES web application interface. On the left is a sidebar with a 'GENERAL' section containing links for 'Applications', 'Documentation', 'Account', and 'Users'. The 'Applications' link is highlighted. The main content area shows the 'SmartEye' project page. At the top right of this section are buttons for 'Subscribe', 'User Access', and a document icon. Below this, there is a description: 'Manage your application here. You can add/remove devices, subscribe to this application, manage user access key, etc.' This is followed by two URI types: 'Hierarchical URI' with the URL 'https://platform.antares.id:8443/~/antares-cse/antares-id/SmartEye' and 'Non-hierarchical URI' with the URL 'https://platform.antares.id:8443/~/antares-cse/CAE840214314'. Below the URIs are two tags: 'Location Outdoor' and 'Category Surveillance'. A link 'How to Create Device via API' is at the bottom of the main content area. At the very bottom of the interface is a search bar labeled 'Search Devices...' and a blue button labeled '+ Add Device'. A 'SmartEye1' device card is partially visible at the bottom left.

# Let's Do Some Hands-On

# Sesi Hands-On

1

Kirim Data Dummy ke ANTARES

2

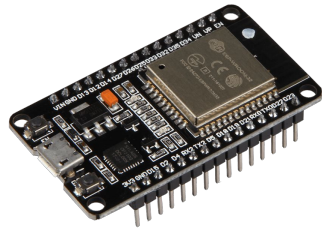
Kirim Data Sensor Environment ke ANTARES

3

Ambil Data Terakhir Sensor Environment di ANTARES

# Mengirimkan Data Dummy

# Mengirimkan Data Dummy



POST



ANTARES

# Mengirimkan Data ke ANTARES

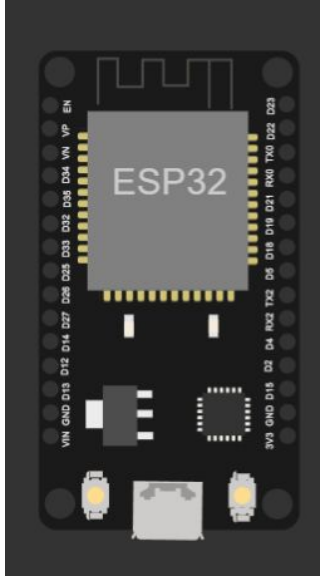
Store Data of a Particular Device

• Request XML JSON

Field	Value								
URL	<code>https://platform.antares.id:8443/~/ antares-cse/ antares-id/ your-application-name/ your-device-name</code> or <code>https://platform.antares.id:8443/~/ antares-cse/ your-device-ID</code>  Note: <code>your-device-ID</code> must be starting with <code>CNT</code> . e.g. <code>CNT-842419289</code>								
Method	<code>POST</code>								
Header	<table><thead><tr><th>Key</th><th>Value</th></tr></thead><tbody><tr><td>X-M2M-Origin</td><td><code>access-id:access-password</code></td></tr><tr><td>Content-Type</td><td><code>application/json;ty=4</code></td></tr><tr><td>Accept</td><td><code>application/json</code></td></tr></tbody></table>	Key	Value	X-M2M-Origin	<code>access-id:access-password</code>	Content-Type	<code>application/json;ty=4</code>	Accept	<code>application/json</code>
Key	Value								
X-M2M-Origin	<code>access-id:access-password</code>								
Content-Type	<code>application/json;ty=4</code>								
Accept	<code>application/json</code>								
Body	<pre>1 { 2   "m2m:cin": { 3     "con": "[{"key1\\":integer-value, \\\"key2\\\":\\\"string-value\\\", \\\"keyN\\\":\\\"valueN\\\"}]" 4   } 5 }</pre>								



# Mengirimkan Data Dummy



```
#include <WiFi.h>
#include <HTTPClient.h>

const char* ssid = "Wokwi-GUEST";
const char* password = "";

// url endpoint data of device
const char* serverName = "http://platform.antares.id:8080/~/antares-cse/antares-id/app-name/device-name";

// inisiasi variable
unsigned long lastTime = 0;
// Setting timer 5 detik
unsigned long timerDelay = 5000;

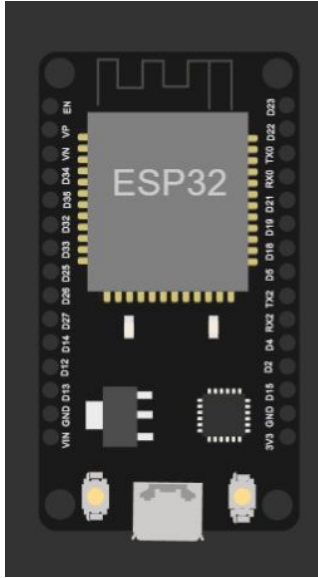
void setup() {
  // inisiasi serial port
  Serial.begin(115200);

  // inisiasi WiFi Client
  WiFi.begin(ssid, password);
  Serial.println("Connecting");
  while(WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.print("Connected to WiFi network with IP Address: ");
  Serial.println(WiFi.localIP());

  Serial.println("Timer set to 5 seconds (timerDelay variable), it will take 5 seconds before publishing the first reading.");
}
```



# Mengirimkan Data Dummy



```
void loop() {
    //rutin melakukan pengiriman data setiap waktu yang diset
    if ((millis() - lastTime) > timerDelay) {
        // mengecek koneksi wifi
        if(WiFi.status()== WL_CONNECTED){
            WiFiClient client;
            HTTPClient http;

            // inisiasi komunikasi http
            http.begin(client, serverName);

            // inisiasi http header
            http.addHeader("X-M2M-Origin", "access-key-kamu-cek-di-antares-account");
            http.addHeader("Content-Type", "application/json;ty=4");
            http.addHeader("Accept", "application/json");
            // inisiasi data yang dikirim di restful api
            String httpRequestData = "{\"m2m:cin\": { \"con\": \"{{{\"status\\\\\\\\\":\\\\\\\\\"0\\\\\\\\\"}}}\"}";
            // mengirimkan HTTP POST request
            int httpResponseCode = http.POST(httpRequestData);

            Serial.print("HTTP Response code: ");
            Serial.println(httpResponseCode);

            // menutup koneksi
            http.end();
        }
        else {
            Serial.println("WiFi Disconnected");
        }
        lastTime = millis();
    }
}
```

# Output Mengirimkan Data Dummy

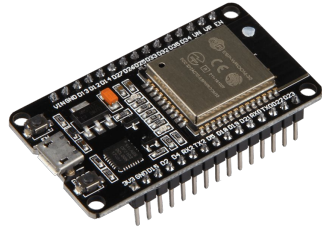
2022-07-22 10:52:50

/antares-cse/cin-Kg8NG8hITSK5Yobd

```
{  
  "status": "0"  
}
```

## 2 Mengirimkan Data Sensor Environment

# Mengirimkan Data DHT22



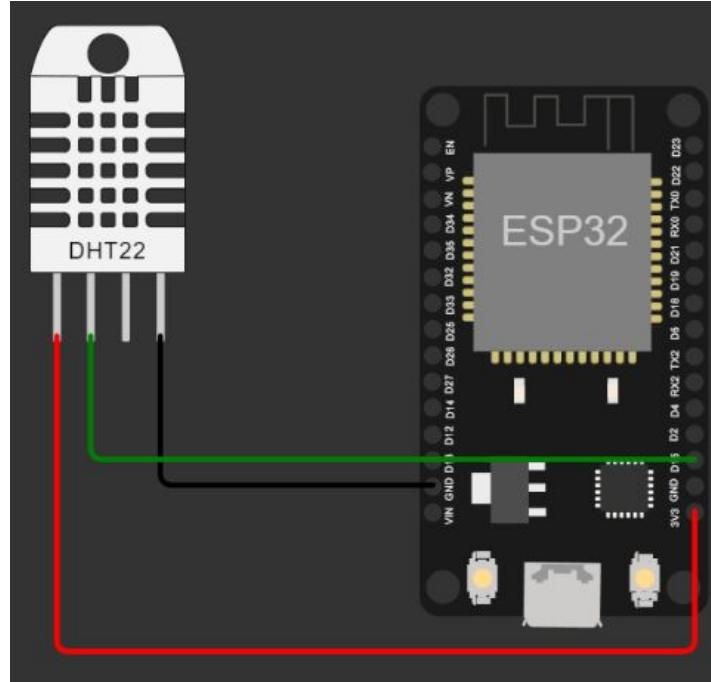
ESP32

POST →



DHT22

# Mengirimkan Data DHT22



# Mengirimkan Data DHT22

```
// proses include library
#include "DHTesp.h"
#include "WiFi.h"
#include "HTTPClient.h"

// deklarasi variable
// set pin yang digunakan
#define DHTPIN 15

const char* ssid = "Wokwi-GUEST";
const char* password = "";

// deklarasi object sensor
// set tipe DHT dan pin yang digunakan
DHTesp dht;

// url endpoint data of device
const char* serverName =
"http://platform.antares.id:8080/~/antares-cse/antares-id/app-name/dev-na
me";

// inisiasi variable
unsigned long lastTime = 0;
// Setting timer 5 detik
unsigned long timerDelay = 5000;
```

```
void setup() {
  // inisiasi interface serial
  Serial.begin(115200);

  Serial.println("EDSPERT - Akuisisi sensor DHT22 via ESP32");

  // inisiasi sensor DHT
  dht.setup(DHTPIN, DHTesp::DHT22);

  // inisiasi WiFi Client
  WiFi.begin(ssid, password);
  Serial.println("Connecting");
  while(WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.print("Connected to WiFi network with IP Address: ");
  Serial.println(WiFi.localIP());

  Serial.println("Timer set to 5 seconds (timerDelay variable), it will
take 5 seconds before publishing the first reading.");
}
```

# Mengirimkan Data DHT22

```
void loop() {  
  //rutin melakukan pengiriman data setiap waktu yang diset  
  if ((millis() - lastTime) > timerDelay) {  
    // deklarasi objek untuk menampung data  
    // temperatur dan kelembapan dari DHT22  
    TempAndHumidity data = dht.getTempAndHumidity();  
  
    // ekstrak data temperature  
    float temp = data.temperature;  
    // ekstrak data humidity  
    float hum = data.humidity;  
  
    // menampilkan data di serial  
    Serial.println("Suhu: " + String(temp, 2) + "°C");  
    Serial.println("Kelembaban: " + String(hum, 1) + "%");  
    Serial.println("----");  
  
    // waktu jeda sampling data  
    // minimal 2 detik  
    delay(5000);  
  
    // mengecek koneksi wifi  
    if(WiFi.status() == WL_CONNECTED){  
      WiFiClient client;  
      HTTPClient http;  
  
      // inisiasi komunikasi http  
      http.begin(client, serverName);
```

```
      // inisiasi http header  
      http.addHeader("X-M2M-Origin",  
        "access-key-kamu-cek-di-antares-account");  
      http.addHeader("Content-Type", "application/json;ty=4");  
      http.addHeader("Accept", "application/json");  
      // inisiasi data yang dikirim di restful api  
      String httpRequestData = "{\"m2m:cin\": { \"con\":  
        \"{{\"temp\\\":{{\"\"}}\"}}\"";  
      httpRequestData = httpRequestData + String(temp, 2);  
      httpRequestData = httpRequestData + "\"\",{{\"hum\\\":{{\"\"}}\"}}\"";  
      httpRequestData = httpRequestData + String(hum,1);  
      httpRequestData = httpRequestData + "\"\"}}}}\"";  
      // mengirimkan HTTP POST request  
      int httpResponseCode = http.POST(httpRequestData);  
  
      Serial.print("HTTP Response code: ");  
      Serial.println(httpResponseCode);  
  
      // menutup koneksi  
      http.end();  
    }  
    else {  
      Serial.println("WiFi Disconnected");  
    }  
    lastTime = millis();  
  }  
}
```

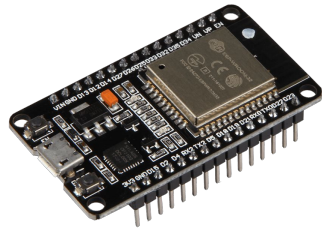


# Output Mengirimkan Data DHT22

Time (WIB)	Resource Index (ri)	Data
2022-07-22 18:19:56	/antares-cse/cin-p21frdfhRVm5jAQg	<pre>{   "temp": "24.00",   "hum": "40.0" }</pre>

# **Mendapatkan Data Sensor Environment**

# Mendapatkan Data DHT22



GET



# Mendapatkan Data DHT22

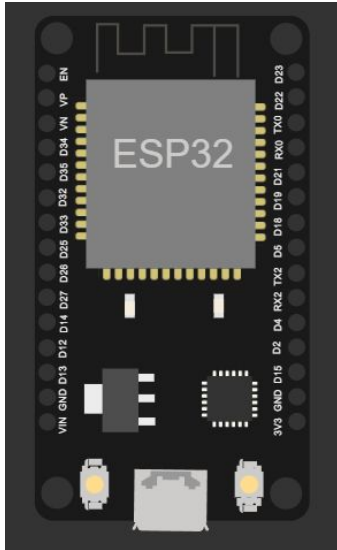
## Retrieve Latest Data

### • Request

XML JSON

Field	Value								
URL	<code>https://platform.antares.id:8443/~/{antares-cse/antares-id/your-application-name/your-device-name/la</code>								
Method	<code>GET</code>								
Header	<table><tr><th>Key</th><th>Value</th></tr><tr><td>X-M2M-Origin</td><td><code>access-id:access-password</code></td></tr><tr><td>Content-Type</td><td><code>application/json;ty=4</code></td></tr><tr><td>Accept</td><td><code>application/json</code></td></tr></table>	Key	Value	X-M2M-Origin	<code>access-id:access-password</code>	Content-Type	<code>application/json;ty=4</code>	Accept	<code>application/json</code>
Key	Value								
X-M2M-Origin	<code>access-id:access-password</code>								
Content-Type	<code>application/json;ty=4</code>								
Accept	<code>application/json</code>								

# Mendapatkan Data DHT22



```
#include <WiFi.h>
#include <HttpClient.h>

const char* ssid = "Wokwi-GUEST";
const char* password = "";

const String url =
  "http://platform.antares.id:8080/~/.antares-cse/antares-id/app-name/device-
  -name/la";

void setup() {
  Serial.begin(115200);
  WiFi.begin(ssid, password);
}

void loop() {
  Serial.print("Connecting to WiFi");
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }

  Serial.print("OK! IP=");
  Serial.println(WiFi.localIP());

  Serial.print("Fetching " + url + "... ");
```

# Mendapatkan Data DHT22

```
HTTPClient http;
http.begin(url);
http.addHeader("X-M2M-Origin",
"access-key-kamu-cek-di-antares-account");
http.addHeader("Content-Type", "application/json;ty=4");
http.addHeader("Accept", "application/json");

int httpResponseCode = http.GET();
if (httpResponseCode > 0) {
  Serial.print("HTTP ");
  Serial.println(httpResponseCode);
  String payload = http.getString();
  Serial.println();
  Serial.println(payload);
}
else {
  Serial.print("Error code: ");
  Serial.println(httpResponseCode);
  Serial.println(":-(");
}

delay(5000);
}
```

# Output Mendapatkan Data DHT22

```
Fetching http://platform.antares.id:8080/~antares-cse/antares-id/workshop-widyaedu/dht22/la... HTTP 200
```

```
{
  "m2m:cin" : {
    "rn" : "cin_c3Qqh0PjTrWZI340",
    "ty" : 4,
    "ri" : "/antares-cse/cin-c3Qqh0PjTrWZI340",
    "pi" : "/antares-cse/cnt-vzQ19SSmSJmT1G2H",
    "ct" : "20220722T184110",
    "lt" : "20220722T184110",
    "st" : 0,
    "cnf" : "text/plain:0",
    "cs" : 29,
    "con" : "{\"temp\": \"24.00\", \"hum\": \"40.0\"}"
  }
}
```

# TUGAS

Membuat sistem alarm suhu  $>40^{\circ}\text{C}$ , maka LED Merah menyala. Gunakan data yang tersimpan di Cloud ANTARES.

HINT

Latest Data of Device ANTARES



# PERSIAPAN PERTEMUAN KE 7

- Instalasi [Driver Serial](#)
- Instalasi [Arduino IDE v1.8](#)
- Instalasi [Board ESP32](#)
- Instalasi [Library LoRaWAN Antares](#)