

MODUL WORKSHOP ANTARES



ANTARES

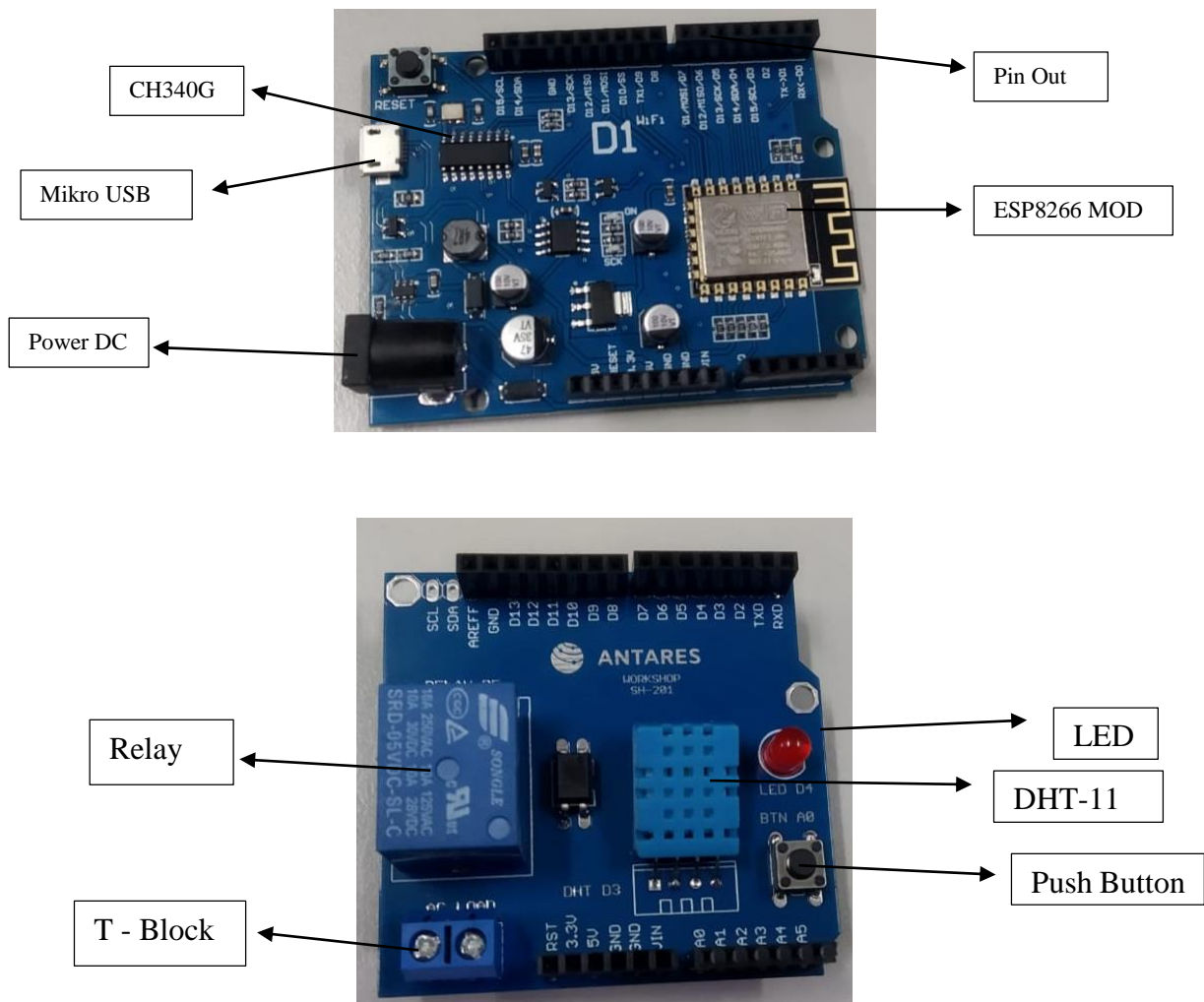
NAMA : _____

WORKSHOP ANTARES

Hardware

Kelengkapan Workshop :

1. ESP8266 (Wemos)
2. DHT11 (Temperature + Humidity Sensor)
3. Relay
4. Kabel Data
5. LED
6. Casing

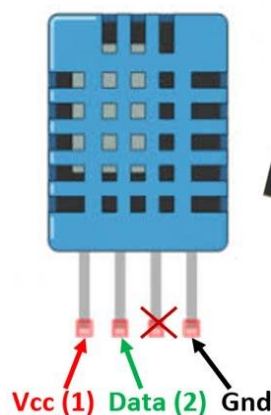


Gambar 1. Wemos R1 dan Shield Antares

ESP8266 (WEMOS)

WEMOS R1 ESP8266 adalah sebuah development board seperti Arduino dan dikembangkan khusus untuk Internet of Things aplikasi dan solusi. Jenis WEMOS ini sangat cocok digunakan untuk pembelajaran dan hobby project Internet Of Things. Dengan sebuah arsitektur 32 bit yang lebih baik diandingkan Arduino dan WEMOS ini memiliki konektivitas WiFi. WEMOS ini dapat deprogram menggunakan Arduino IDE dan Bahasa Lua.

DHT11



Gambar 2. Sensor Kelembapan dan Temperature DHT11

DHT11 merupakan sensor kelembapan dan temperature yang memiliki spesifikasi seperti berikut :

- Ultra low cost
- 3 to 5V power and I/O
- 2.5mA max current use during conversion (while requesting data)
- Good for 20-80% humidity readings with 5% accuracy
- Good for 0-50°C temperature readings $\pm 2^{\circ}\text{C}$ accuracy
- No more than 1 Hz sampling rate (once every second)
- Body size 15.5mm x 12mm x 5.5mm
- 4 pins with 0.1" spacing

Relay



Gambar 3. Modul Relay

Relay adalah suatu peranti yang menggunakan elektromagnet untuk mengoperasikan seperangkat kontak sakelar. Relay terdiri dari kumparan kawat penghantar yang dililit pada inti besi. Bila kumparan ini diberikan energi, medan magnet yang terbentuk menarik armatur berporos yang digunakan sebagai pengungkit mekanisme sakelar magnet.

Software

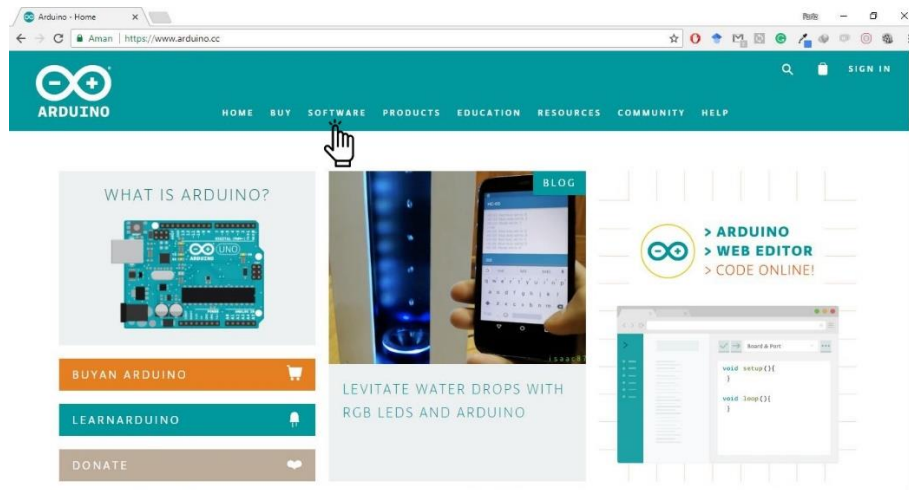
Perangkat lunak (Software) yang dibutuhkan :

1. Arduino IDE
2. Library ESP8266
3. Library Pendukung (DHT, MQTT, dll)
4. Library Antares
5. Android Studio

Install Arduino IDE

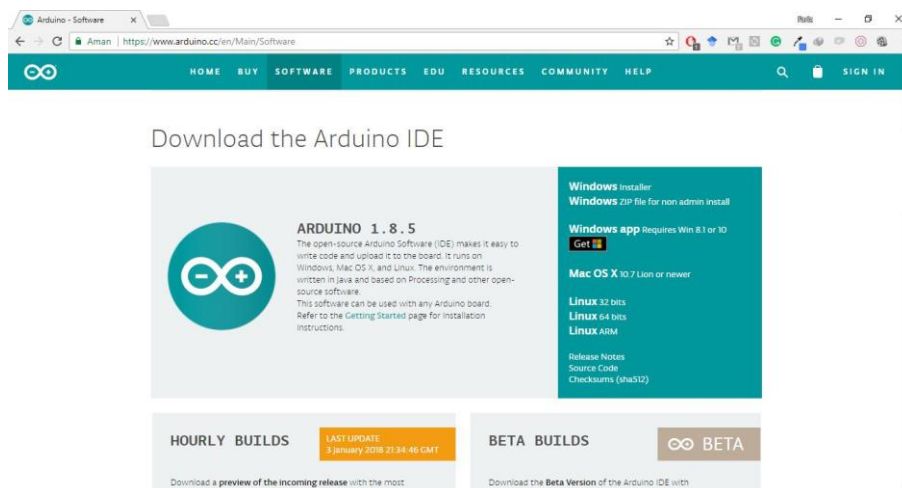
Arduino IDE digunakan untuk membuat/menuliskan sebuah program yang selanjutnya akan di-compile dan di-flash ke sebuah chip microcontroller contohnya Arduino (atmega, atmega, dll), WEMOS (esp8266, esp32), dan lain-lain. Untuk melakukan instalasi software Arduino IDE, langkah-langkahnya sebagai berikut :

1. Download software di Arduino.cc



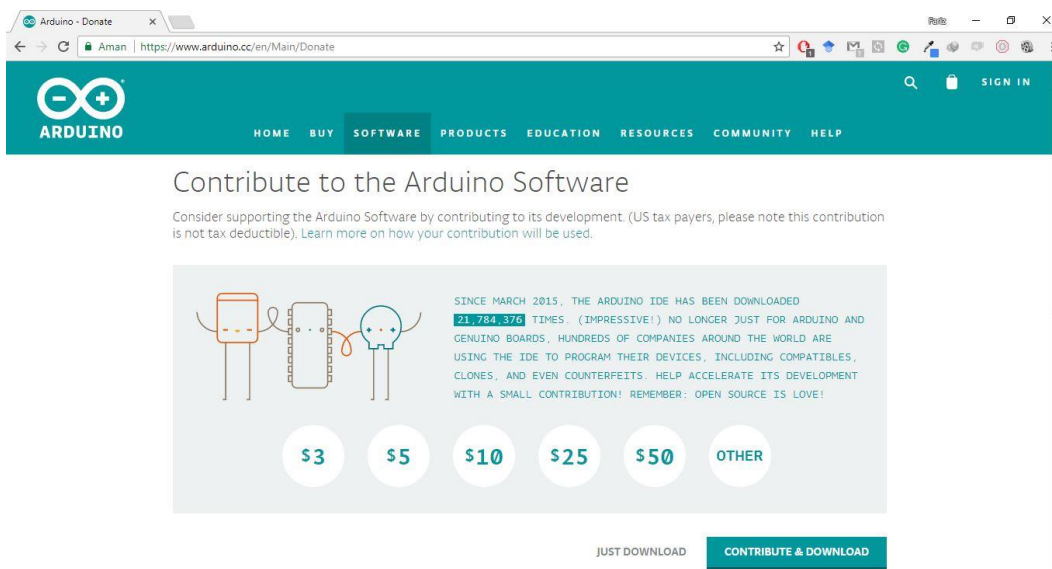
Gambar 4. Arduino Installation Step (1)

2. Pilih Menu Software



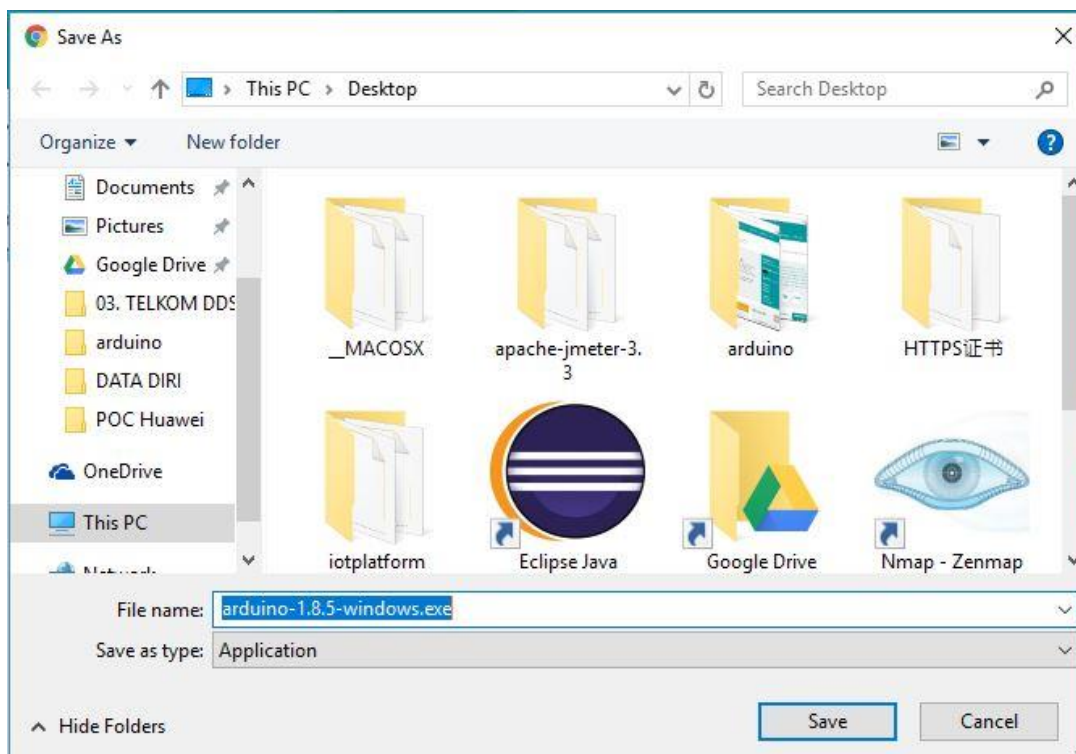
Gambar 5. Arduino Installation Step (2)

3. Pilih system operasi yang sesuai



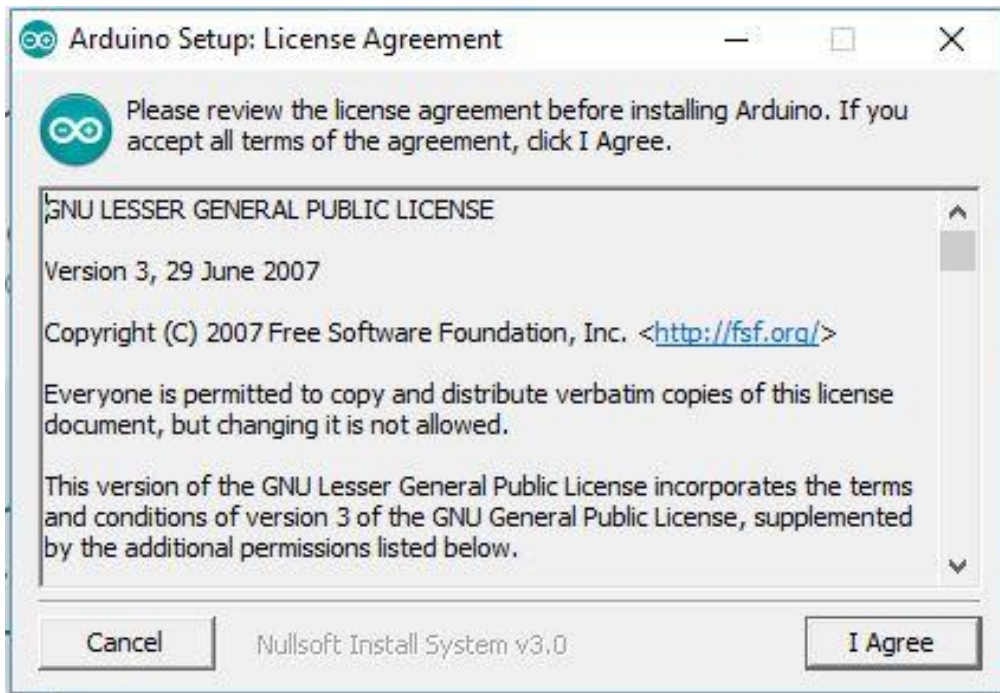
Gambar 6. Arduino Installation Step (3)

4. Pilih direktori penyimpanan



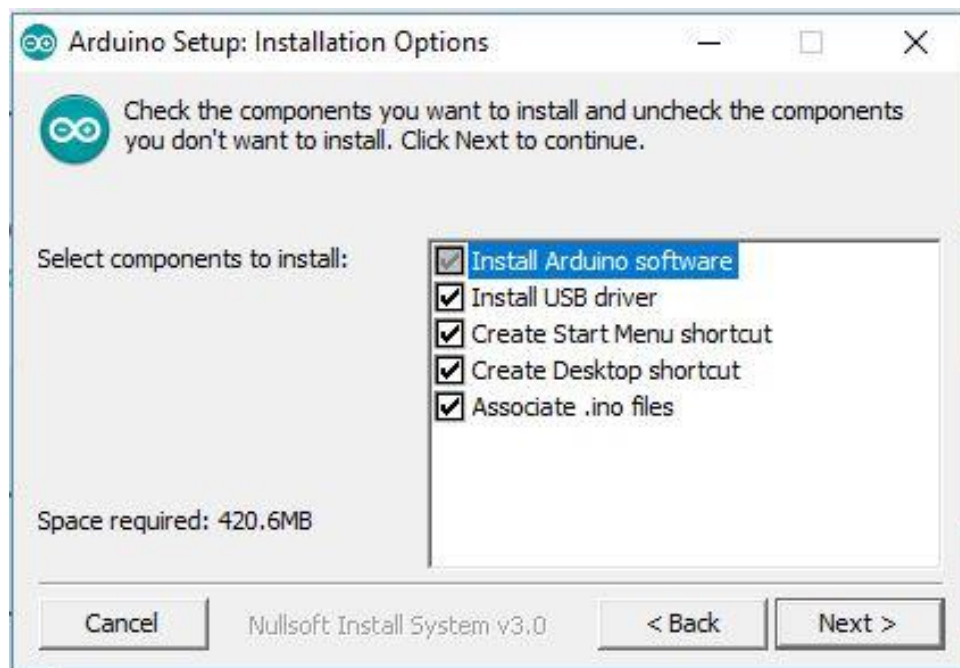
Gambar 7. Arduino Installation Step (4)

5. Setelah download selesai, Lakukan instalasi dan pilih I agree



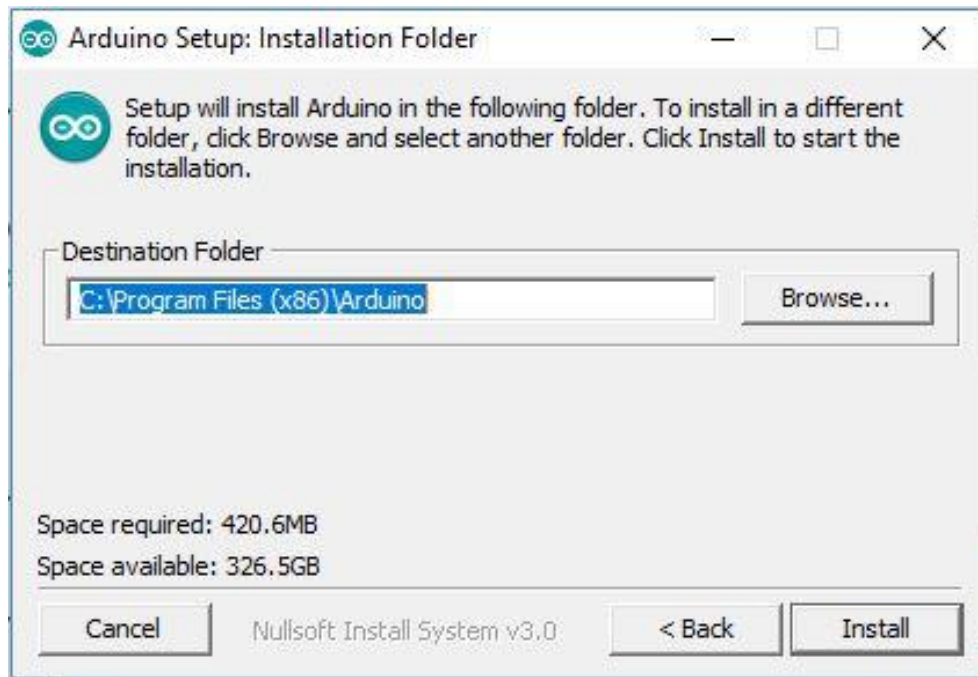
Gambar 8. Arduino Installation Step (5)

6. Ceklis semua pilihan, lalu Next



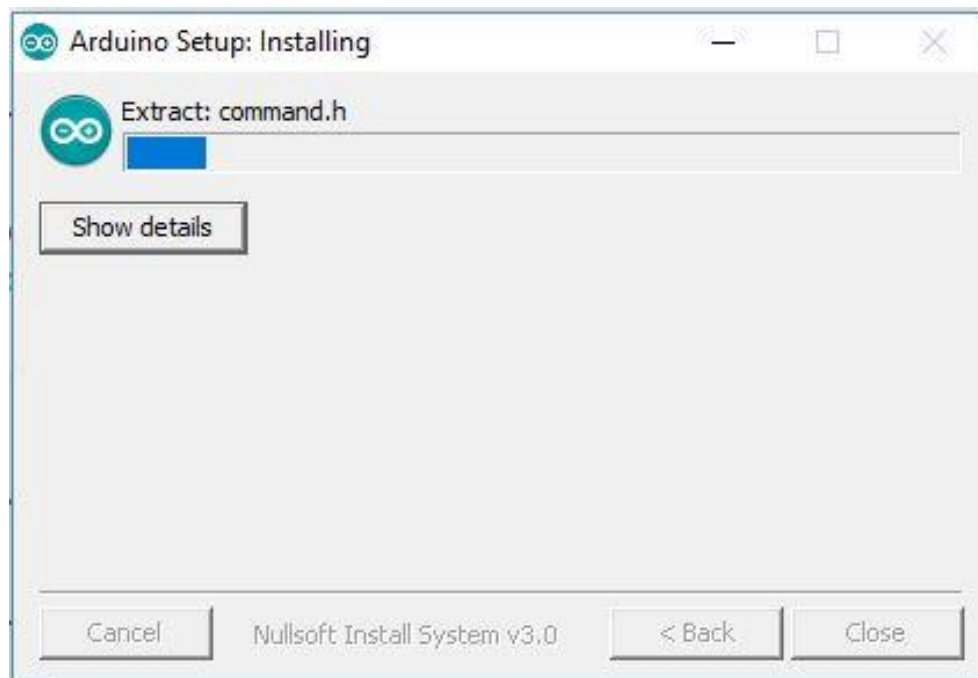
Gambar 9. Arduino Installation Step (6)

7. Pilih destinasi penyimpanan



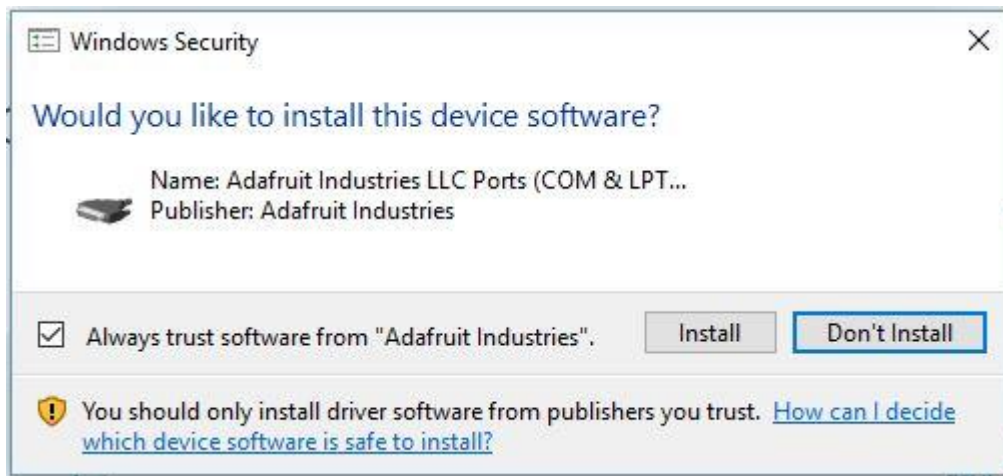
Gambar 10. Arduino Installation Step (7)

8. Tunggu hingga selesai instalasi



Gambar 11. Arduino Installation Step (8)

9. Pilih Install apabila muncul kotak dialog seperti gambar dibawah.



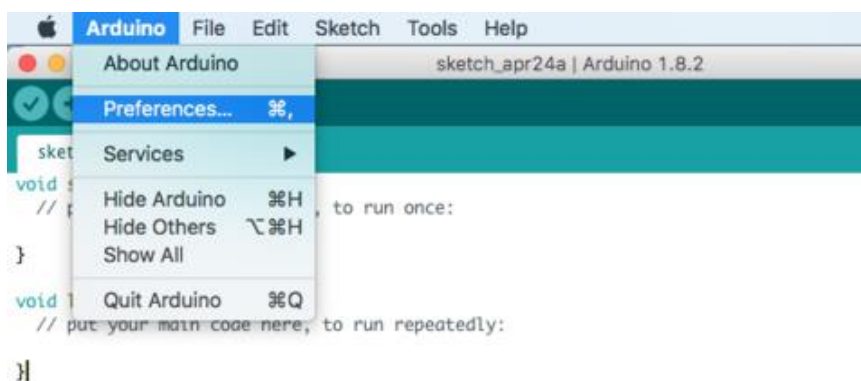
Gambar 12. Arduino Installation Step (9)

Arduino IDE telah terinstall dan siap untuk digunakan untuk membuat project.

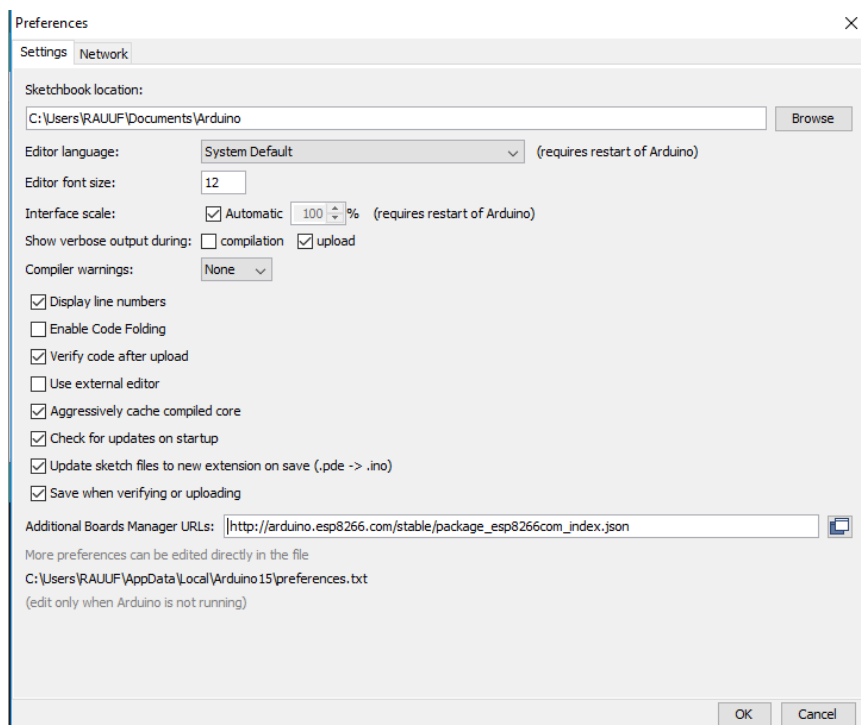
Install Board & Library ESP8266

Untuk memulai membuat program pada WEMOS, hal yang perlu dilakukan yaitu melakukan instalasi library esp8266 pada Arduino IDE. Instalasi library dilakukan dengan langkah sebagai berikut :

1. Buka aplikasi Arduino IDE
2. Pilih Files -> Preferences dan masukan URL di kolom Additional Boards Manager URL
http://arduino.esp8266.com/stable/package_esp8266com_index.json

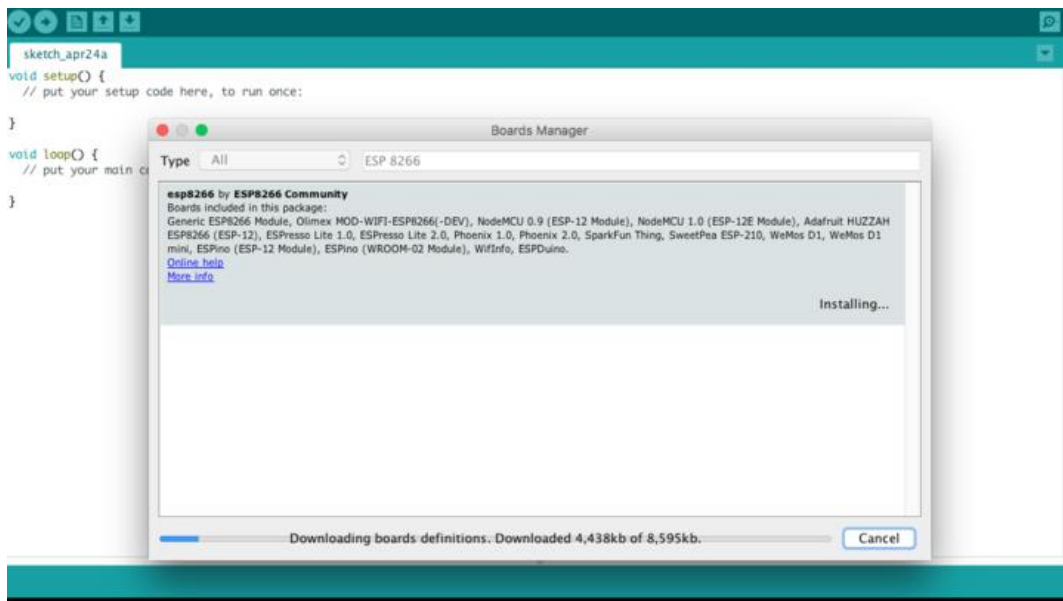


Gambar 13. ESP8266 Board & Library Installation Step (1)



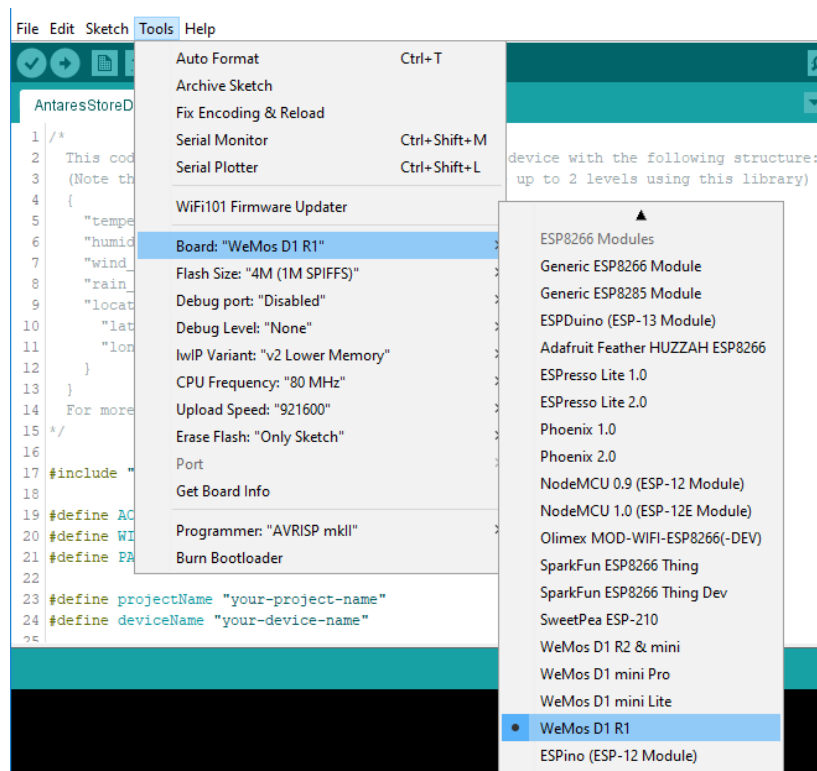
Gambar 14. ESP8266 Board & Library Installation Step (2)

3. Pilih Tools -> Boards -> Boards Manager dan install esp8266



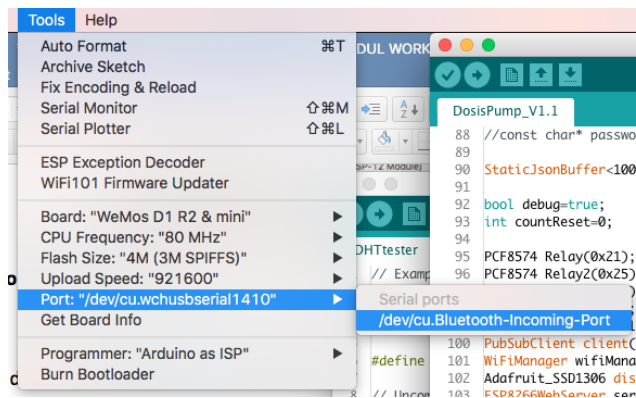
Gambar 15. ESP8266 Board & Library Installation Step (3)

4. Pilih WEMOS D1 R1 di menu Tools -> Board



Gambar 16. ESP8266 Board & Library Installation Step (4)

5. Pilih **Tools > Port > COM(X)**. Nilai X menyesuaikan nomor yang tampil di port.



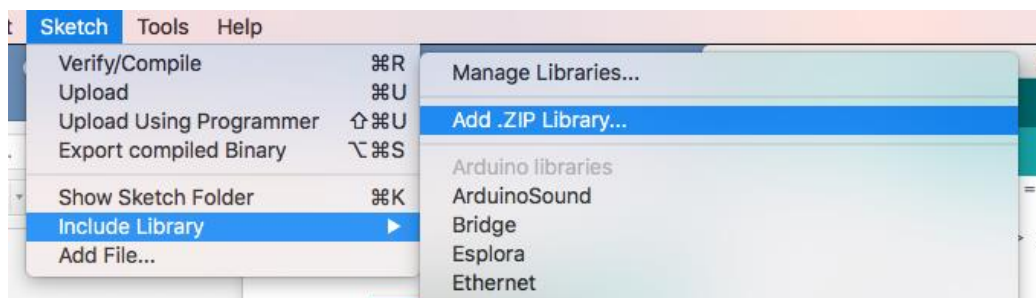
Gambar 17. ESP8266 Board & Library Installation Step (5)

6. Tutup dan buka kembali Arduino IDE

Install Library Pendukung

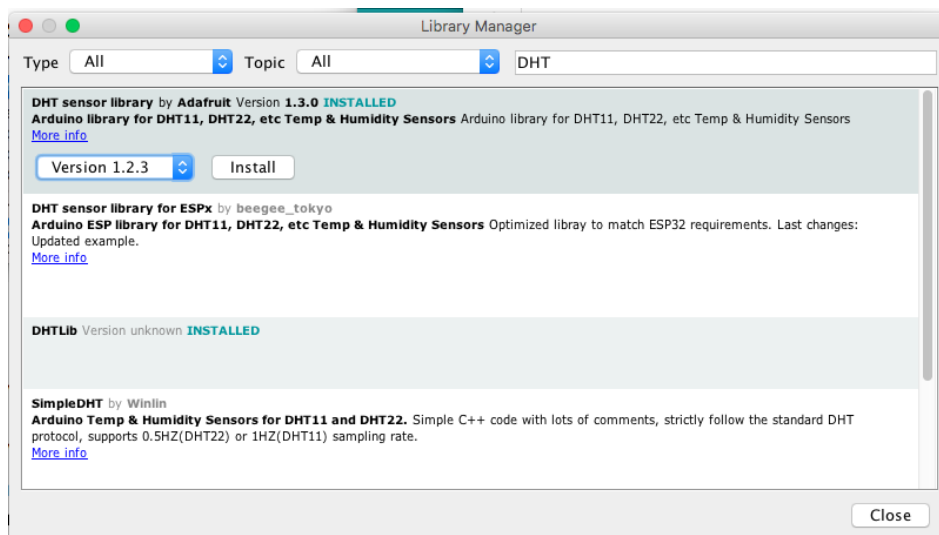
Library pendukung ini digunakan untuk mempermudah dalam membuat sebuah program. Di dalam library tersebut terdapat beberapa fungsi yang dapat digunakan untuk melakukan akses terhadap hardware atau software tertentu. Untuk menambahkan library pendukung ke Arduino IDE, dapat dilakukan dengan cara :

1. Pilih Sketch -> Include Library -> Manage Librares, Atau dapat pilih Add .ZIP Library apabila library sudah ada dalam computer



Gambar 18. Extra Library Installation Step (1)

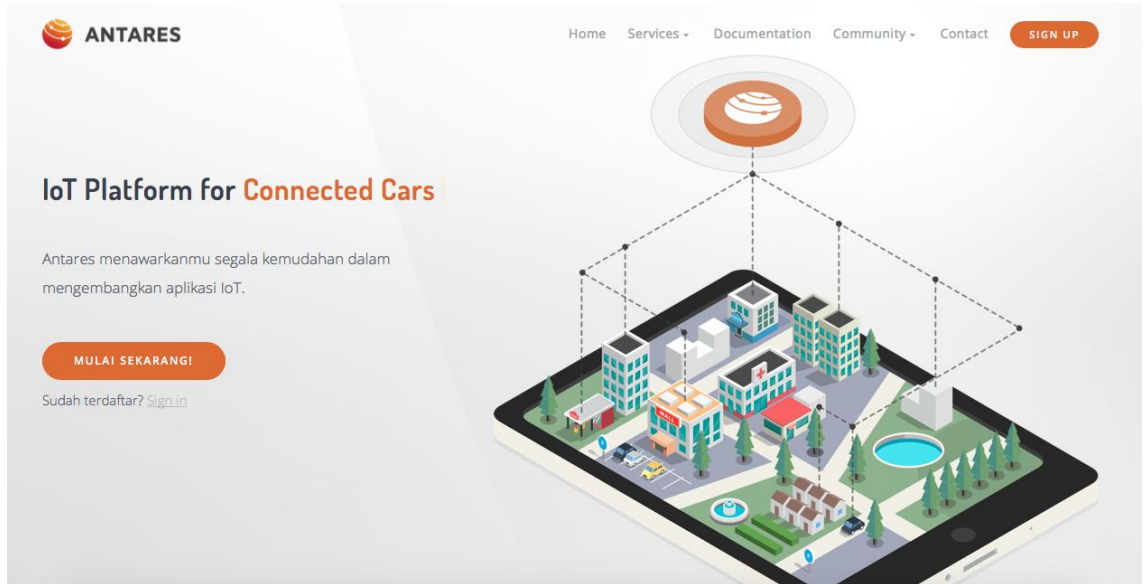
2. Cari library yang akan di install
3. Ketikkan DHT di kolom search
4. **Pilih versi 1.2.2 atau 1.2.3 pada kolom version**



Gambar 19. Extra Library Installation Step (2)

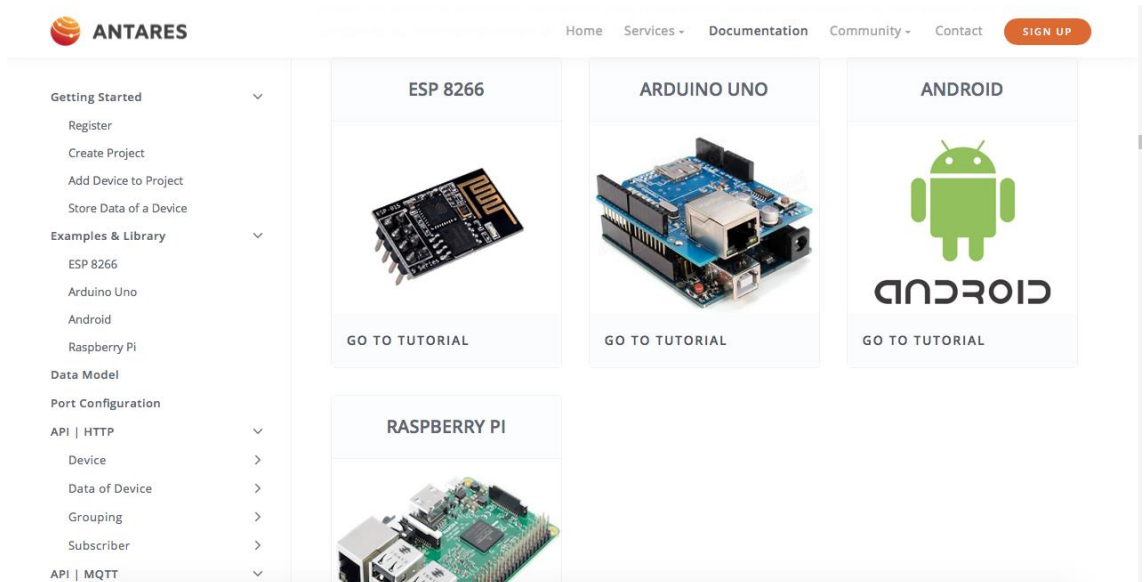
Install Library Antares For ESP8266

1. Masuk Ke Antares.id



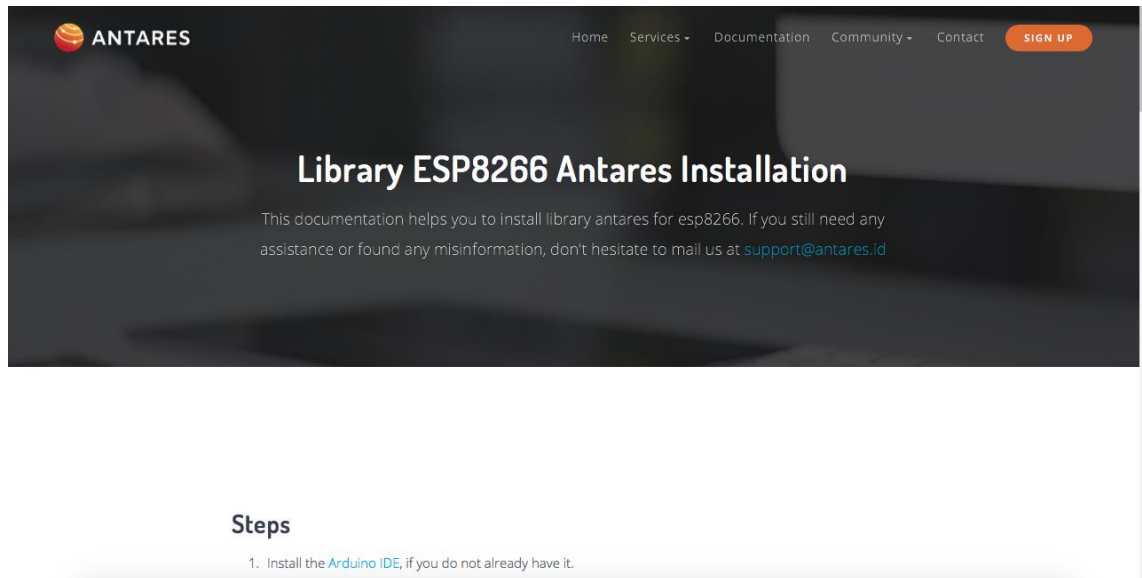
Gambar 20. Library Antares HTTP for ESP8266 Installation Step (1)

2. Pilih Documentation



Gambar 21. Library Antares HTTP for ESP8266 Installation Step (2)

3. Klik Go To Tutorial ESP8266



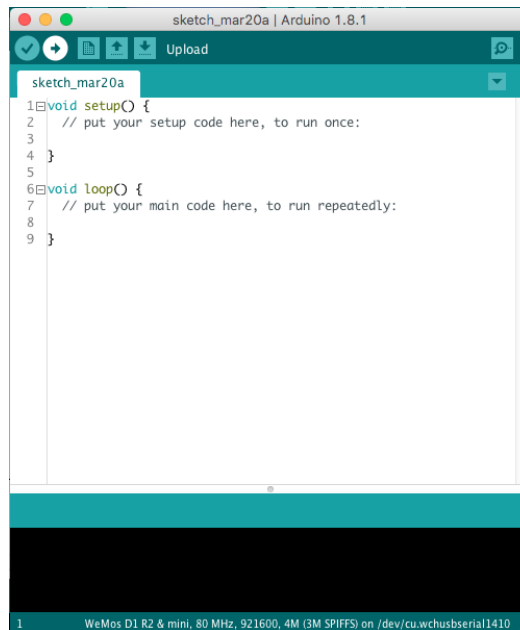
Gambar 22. Library Antares HTTP for ESP8266 Installation Step (3)

4. Ikuti Instruksi hingga selesai.


Membuat Program Sederhana

Untuk membuat program langkah-langkah yang dilakukan yaitu :

- a. Buka aplikasi Arduino IDE
- b. Pilih File -> New
- c. Jendela editor akan terbuka



Gambar 23. Jendela Editor Arduino IDE

- d. Setelah menuliskan baris program, langkah selanjutnya save program dan pilih icon .

1. Hardware Test Program

```
#include "DHT.h"

#define DHTTYPE DHT11
#define SENSOR_DHT D3
#define LEDPIN D4
#define RELAY D5

DHT dht(SENSOR_DHT, DHTTYPE);

void relayOn()
{
  digitalWrite(RELAY,HIGH);
}
```

```

void relayOff()
{
  digitalWrite(RELAY,LOW);
}

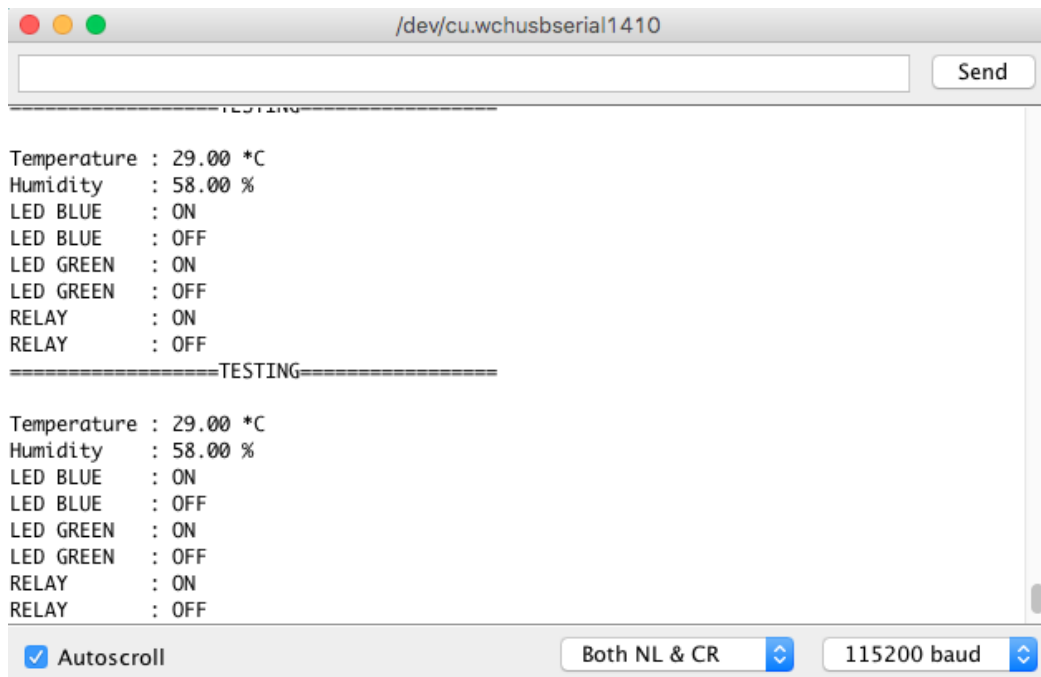
float getTemperature()
{
  float t = dht.readTemperature();
  if (isnan(t)) return 0;
  return t;
}

float getHumidity()
{
  float h = dht.readHumidity();
  if (isnan(h)) return 0;
  return h;
}

void setup() {
  Serial.begin(115200);
  pinMode(LEDPIN,OUTPUT);
  pinMode(RELAY,OUTPUT);
  dht.begin();
  Serial.println("Test Hardware Workshop Antares ...");
}

void loop() {
  Serial.println("=====TESTING=====");
  Serial.println("Temperature : " + (String)getTemperature()+" *C");
  Serial.println("Humidity   : " + (String)getHumidity()+" %");
  delay(1000);
  Serial.println("LED       : ON");
  digitalWrite(LEDPIN,LOW);
  delay(1000);
  Serial.println("LED       : OFF");
  digitalWrite(LEDPIN,HIGH);
  delay(1000);
  Serial.println("RELAY    : ON");
  relayOn();
  delay(1000);
  Serial.println("RELAY    : OFF\n");
  relayOff();
  delay(2000);
}

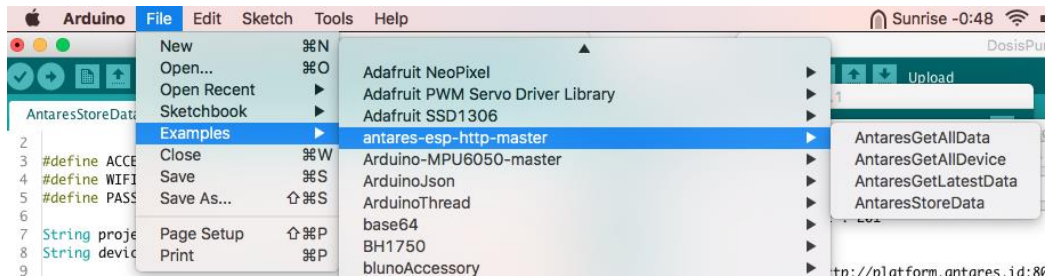
```



Gambar 24. Jendela Serial Monitor Arduino IDE

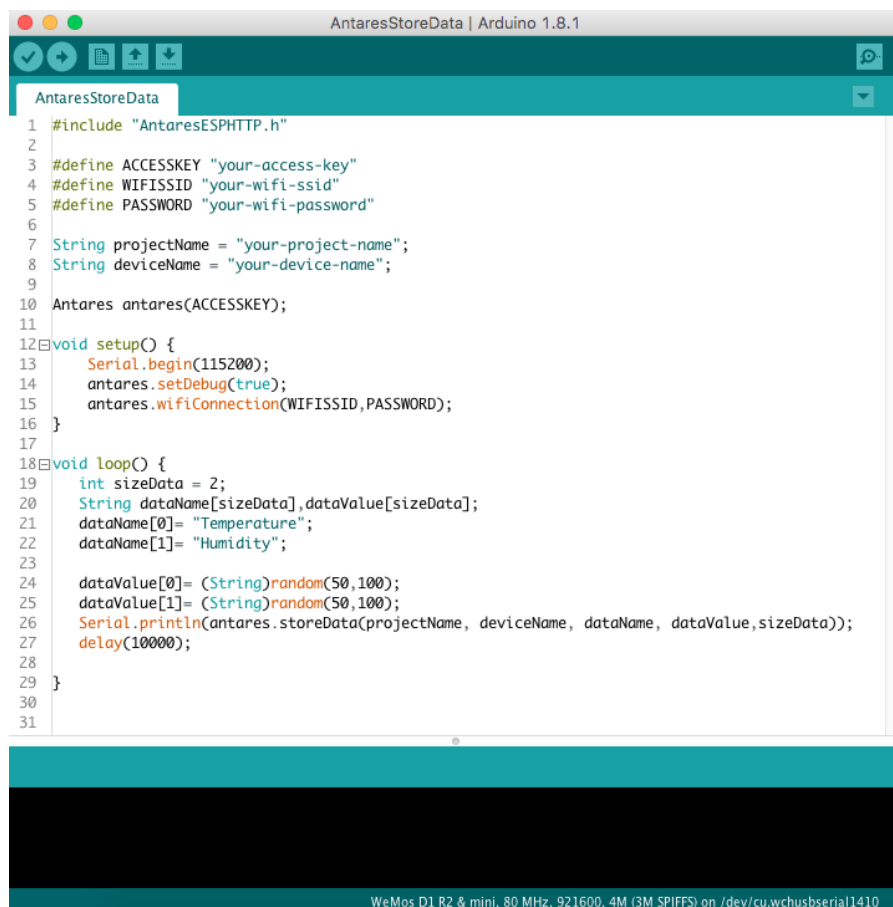
2. Mengirim Data Ke Antares

1. Buka source example di File -> Examples -> Antares-esp-http-master -> Antares Store Data



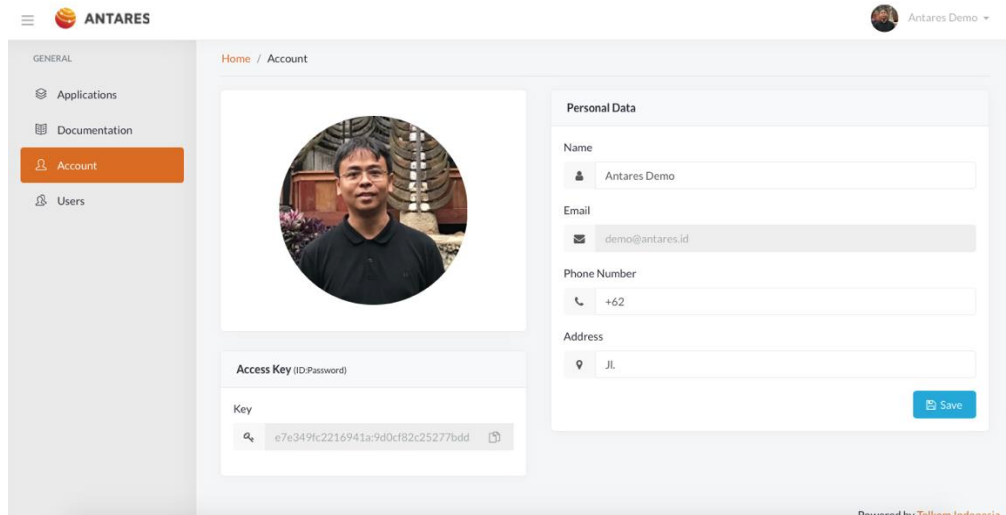
Gambar 25. Step (1) Send Data to Antares

2. Isi AccessKey, SSID, Password, projectName, deviceName sesuai dengan user account Antares.



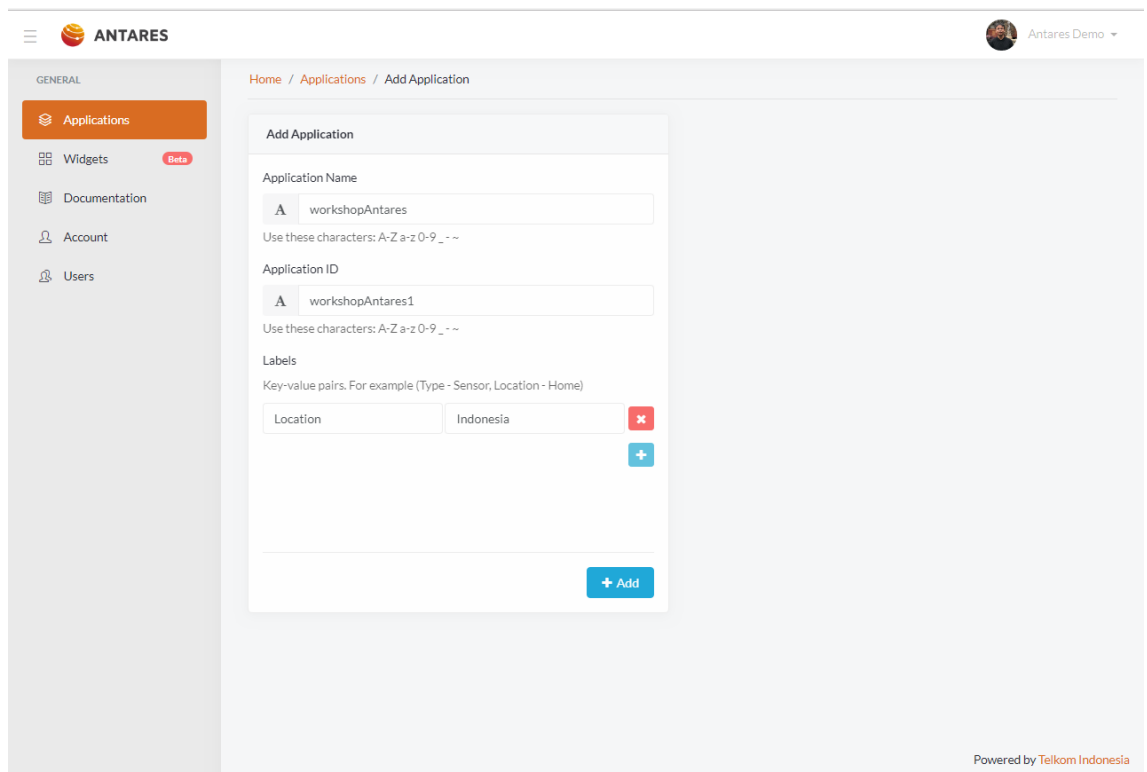
Gambar 26. Step (2) Send Data to Antares

3. Mendapatkan AccessKey, Login di console.antares.id -> Account



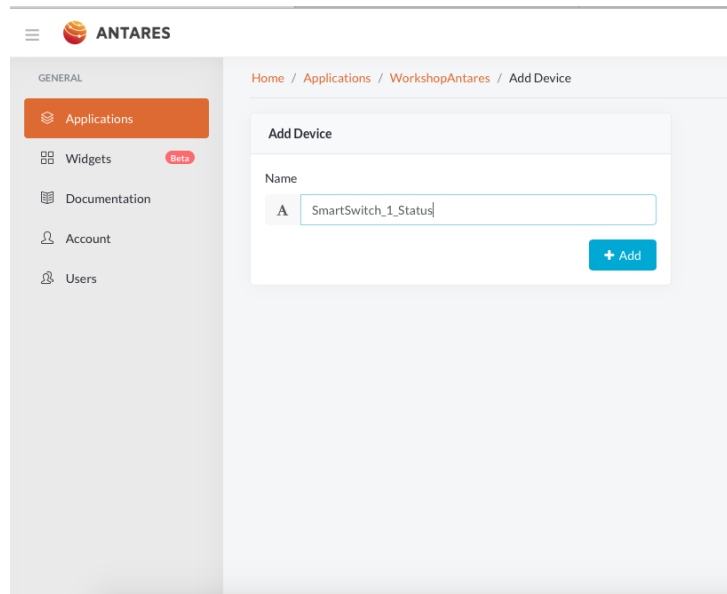
Gambar 27. Step (3) Send Data to Antares

4. Membuat Application/Project

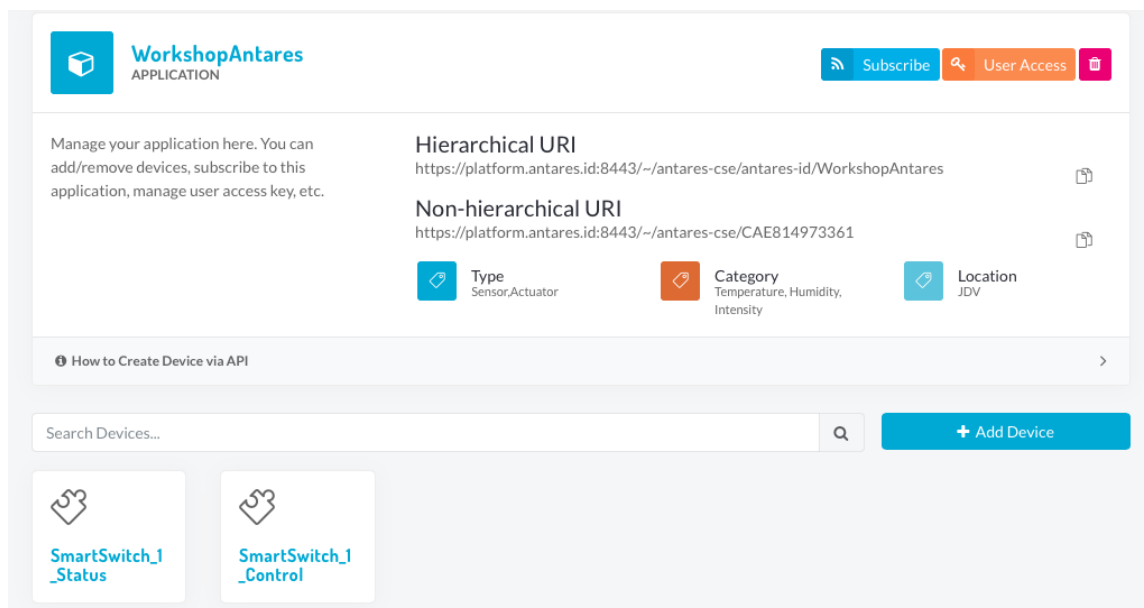


Gambar 28. Step (4) Send Data to Antares

5. Membuat Device



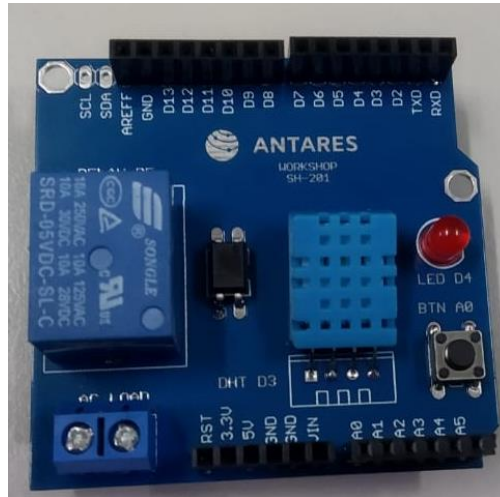
Gambar 29. Step (5a) Send Data to Antares



Gambar 30. Step (5b) Send Data to Antares

3. Mengirim data sensor suhu dan kelembapan ke Antares

Gunakan device sebelumnya yang sudah di tambahkan di Antares. Sambungkan shield yang sudah disediakan. Dan Upload program seperti dibawah.



Gambar 31. Wiring Smart Switch Untuk Lampu

```
#include "DHT.h"
#include "AntaresESP8266HTTP.h"

//IO Define
#define DHTTYPE DHT11
#define SENSOR_DHT D3
#define RELAY D5

//ANTARES & Wifi Data
#define ACCESSKEY "your-access-key"
#define WIFISSID "yout-wifi-ssid"
#define PASSWORD "your-wifi-password"

String projectName = "your-application-name";
String deviceName = "your-device-name";

DHT dht(SENSOR_DHT, DHTTYPE);

AntaresESP8266HTTP antares(ACCESSKEY);

float getTemperature()
{
    float t = dht.readTemperature();
    if (isnan(t)) return 0;
}
```



```

    return t;
}

float getHumidity()
{
    float h = dht.readHumidity();
    if (isnan(h)) return 0;
    return h;
}

void setup() {
    Serial.begin(115200);
    dht.begin();
    antares.setDebug(true);
    antares.wifiConnection(WIFISSID,PASSWORD);
    Serial.println("Test Hardware Workshop Antares ...");
}

void loop() {
    Serial.println("=====TESTING=====");
    Serial.println("Temperature : " + (String)getTemperature()+" *C");
    Serial.println("Humidity   : " + (String)getHumidity()+" %");

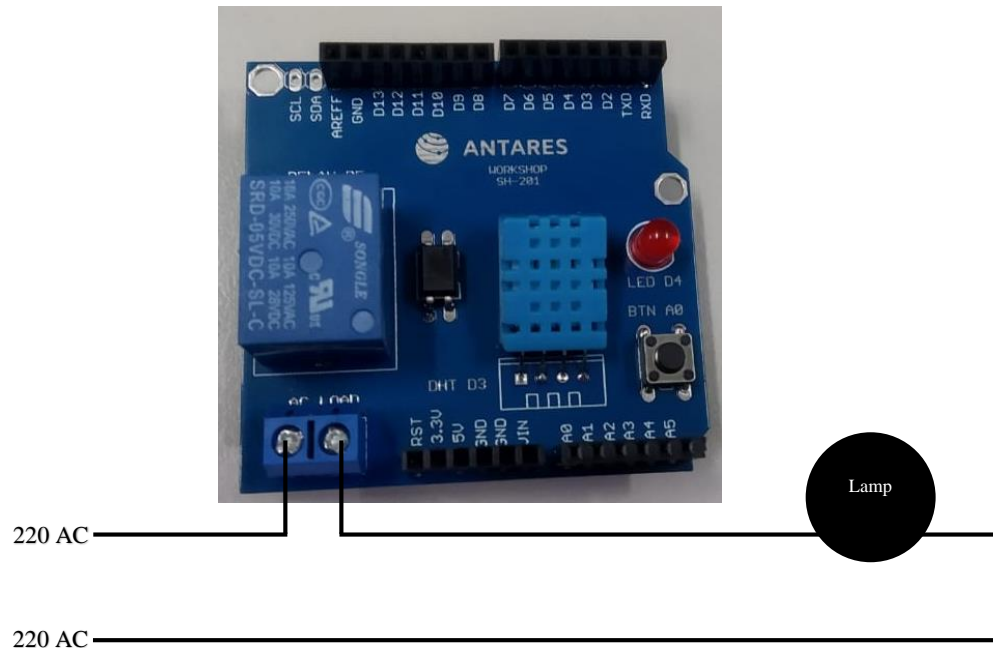
    int sizeData = 2;
    String dataName[sizeData],dataValue[sizeData];
    dataName[0]= "Temperature";
    dataName[1]= "Humidity";

    dataValue[0]= (String)getTemperature();
    dataValue[1]= (String)getHumidity();
    Serial.println(antares.storeData(projectName, deviceName, dataName,
    dataValue,sizeData));
    delay(10000);
}

```

4. Membuat Project SmartSwitch

Setelah membuat device **SmartSwitchStatus** dan **SmartSwitchControl** di console Antares. Maka selanjutnya membuat wiring dan program di hardware.



Gambar 32. Wiring Smart Switch Untuk Lampu

```
#include "DHT.h"
#include <ArduinoJson.h>
#include "AntaresESP8266HTTP.h"

//IO Define
#define DHTTYPE DHT11
#define SENSOR_DHT D3
#define LEDPIN D4
#define RELAY D5

//ANTARES & Wifi Data
#define ACCESSKEY "your-access-key"
#define WIFISSID "your-wifi-ssid"
#define PASSWORD "your-wifi-password"

StaticJsonBuffer<10000> jsonBuffer;

String projectName = "your-application-name";
String deviceNameStatus = "your-status-project-name";
```

```

String deviceNameControl = "yout-control-project-name";
String Value,Unit,Label;

unsigned long intervalSend=10000; // the time we need to wait
unsigned long previousMillis=0;
int statusSwitch=0;

DHT dht(SENSOR_DHT, DHTTYPE);

AntaresESP8266HTTP antares(ACCESSKEY);

void ledOn()
{
  digitalWrite(LEDPIN,LOW);
}

void ledOff()
{
  digitalWrite(LEDPIN,HIGH);
}

float getTemperature()
{
  float t = dht.readTemperature();
  if (isnan(t)) return 0;
  return t;
}

float getHumidity()
{
  float h = dht.readHumidity();
  if (isnan(h)) return 0;
  return h;
}

void setup() {
  Serial.begin(115200);
  pinMode(RELAY,OUTPUT);
  pinMode(LEDPIN,OUTPUT);
  dht.begin();
  antares.setDebug(true);
  antares.wifiConnection(WIFISSID,PASSWORD);
  Serial.println("Test Hardware Workshop Antares ...");
}

```

```

}

void loop() {
  unsigned long currentMillis = millis();
  if (antares.checkWifiConnection())
  {
    if ((unsigned long)(currentMillis - previousMillis) >= intervalSend) {
      float humidity = dht.readHumidity();
      float temperature = dht.readTemperature();

      int sizeData = 3;
      String dataName[sizeData],dataValue[sizeData];
      dataName[0]= "temperature";
      dataName[1]= "humidity";
      dataName[2]= "status";

      dataValue[0]= (String)humidity;
      dataValue[1]= (String)temperature;
      dataValue[2]= (String)statusSwitch;

      String responseCode = antares.storeData(projectName,
deviceNameStatus, dataName, dataValue, sizeData);

      if (responseCode=="201")
      {
        Serial.println("SENDING SUCCESS");
        ledOn();
        delay(1000);
        ledOn();
      }
      else
      {
        Serial.println("SENDING FAILED");
        ledOff();
        delay(1000);
        ledOff();
      }
      previousMillis = millis();
    }
  }
}

```

```

String dataFromAntares =
antares.retrieveLatestData(projectName,deviceNameControl);

if (dataFromAntares != "")
{
dataFromAntares.replace(" ", "");
dataFromAntares.replace("\n", "");
dataFromAntares.replace("\r", "");
dataFromAntares.replace("\\\\"", " ");
dataFromAntares.replace("[", "");
dataFromAntares.replace("]", "");
Serial.println(dataFromAntares);
JsonObject& root = jsonBuffer.parseObject(dataFromAntares);
JsonObject& root4 =
jsonBuffer.parseObject(root["m2m:cin"].as<String>());
Label = root4["pi"].as<String>();

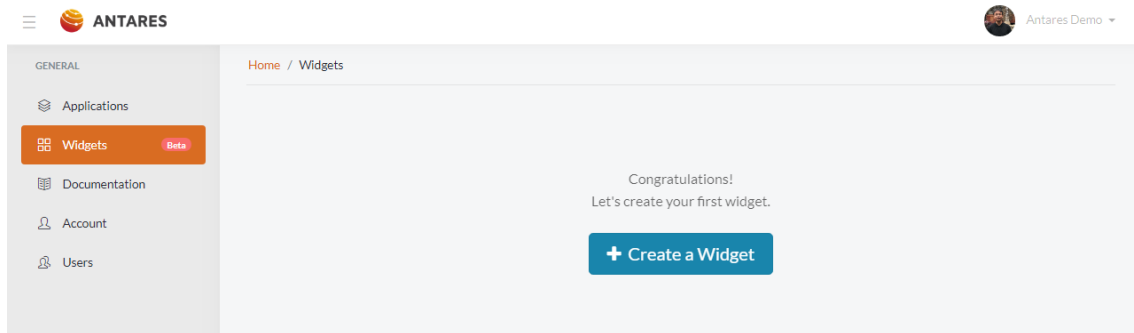
String con = root4["con"].as<String>();
con.replace(" ", "\\");

JsonObject& contentAntares = jsonBuffer.parseObject(con);
String statusSwitchs = contentAntares["status"].as<String>();
Serial.println(Label);
//if (Label == URI_PROJECT_CONTROL)
//{
if (statusSwitchs == "1")
{
digitalWrite(RELAY,HIGH);
statusSwitch=0;
}
else
{
digitalWrite(RELAY,LOW);
statusSwitch=1;
}
//}
jsonBuffer.clear();
dataFromAntares="";
}
}
delay(1000);
}

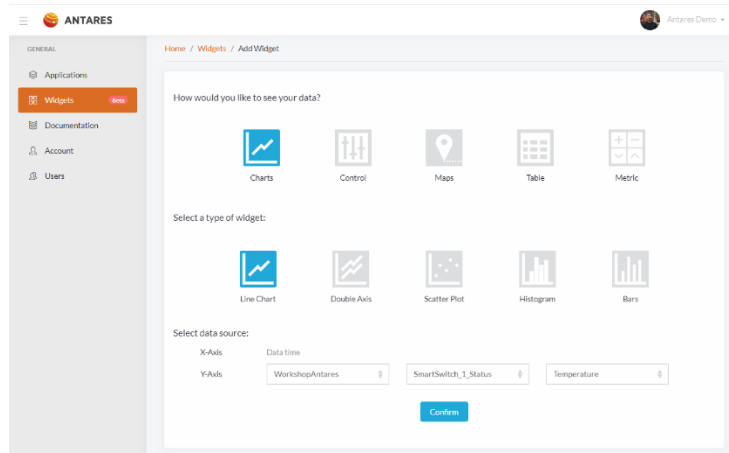
```

Membuat Widget

1. Pilih Menu Widget -> Create a Widget



2. Pilih Jenis Widget



3. Data akan tampil pada widget

