MODUL 3

Perancangan Node Device menggunakan Raspberry Pi 3 (Publisher)

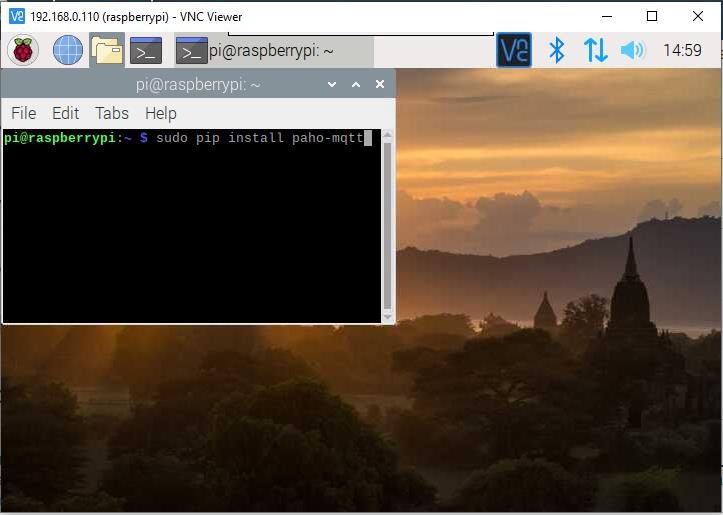
# Percobaan 1

**Programming Raspberry Pi**

1. Gunakan monitor atau VNC yang sudah tersedia

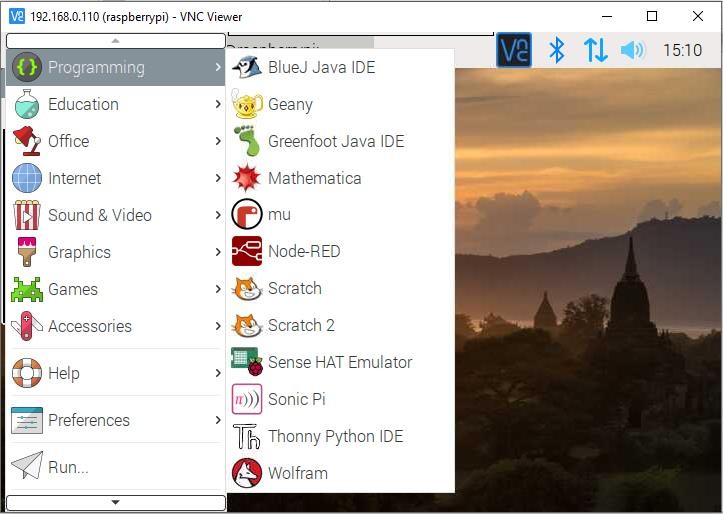
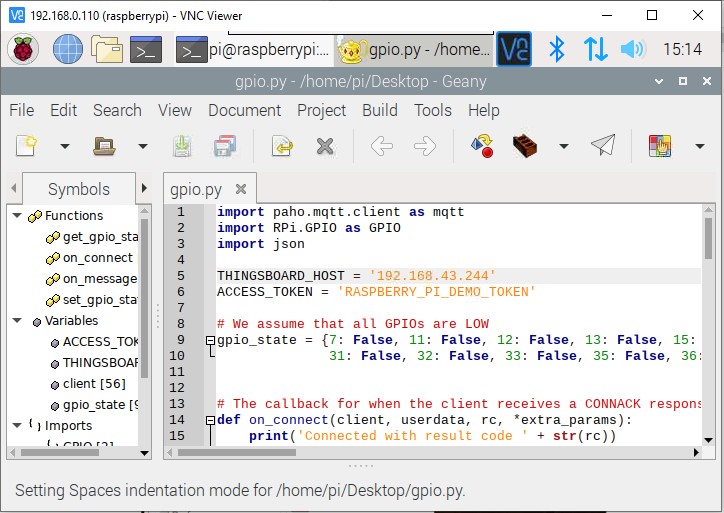
## Pada jendela raspberry pi akan tampil seperti berikut ini

1. Kemudian buka terminal. Kemudian ketik dan install



**sudo pip install paho-mqtt**

Apabila ditanyakan password ketikkan : **raspberry**

1. Buka programing IDE geany untuk memasukkan program
2. Maka akan muncul halaman pemrograman berikut ini

Masukkan program berikut ini

#Libraries import os import time import sys

import paho.mqtt.client as mqtt import json

import RPi.GPIO as GPIO import time

#GPIO Mode (BOARD / BCM)

GPIO.setmode(GPIO.BCM) #set GPIO Pins GPIO\_TRIGGER = 18

GPIO\_ECHO = 24

#set GPIO direction (IN / OUT) GPIO.setup(GPIO\_TRIGGER, GPIO.OUT) GPIO.setup(GPIO\_ECHO, GPIO.IN)

THINGSBOARD\_HOST = 'SERVER IP ADDRESS' ACCESS\_TOKEN = 'TOKEN\_ACCESS'

# Data capture and upload interval in seconds. Less interval will eventually.

INTERVAL=1

sensor\_data = {'distance': 0} next\_reading = time.time() client = mqtt.Client()

# Set access token client.username\_pw\_set(ACCESS\_TOKEN)

# Connect to ThingsBoard using default MQTT port and 60 seconds keepalive interval

client.connect(THINGSBOARD\_HOST, 1883, 60) client.loop\_start()

def distance():

# set Trigger to HIGH

GPIO.output(GPIO\_TRIGGER, True)

# set Trigger after 0.01ms to LOW time.sleep(0.00001) GPIO.output(GPIO\_TRIGGER, False) StartTime = time.time()

StopTime = time.time() # save StartTime

while GPIO.input(GPIO\_ECHO) == 0: StartTime = time.time()

# save time of arrival

while GPIO.input(GPIO\_ECHO) == 1: StopTime = time.time()

# time difference between start and arrival TimeElapsed = StopTime - StartTime

# multiply with the sonic speed (34300 cm/s) # and divide by 2, because there and back distance = (TimeElapsed \* 34300) / 2

return distance

if name == ' main ': try:

while True:

dist = distance()

print ("Measured Distance = %.1f cm" % dist) sensor\_data['distance'] = dist client.publish('v1/devices/me/telemetry',

json.dumps(sensor\_data), 1)

next\_reading += INTERVAL

sleep\_time = next\_reading-time.time() if sleep\_time > 0:

time.sleep(sleep\_time)

# Reset by pressing CTRL + C except KeyboardInterrupt:

print("Measurement stopped by User") GPIO.cleanup()

client.loop\_stop() client.disconnect()

1. **Kemudian buatlah device pada thingsboard terlebih dahulu**

## Hasil gambar untuk srf05 raspberry piGunakan breadboard dan sensor jarak dan buatlah rangkaiannya terlebih dahulu

GPIO 18/PIN 12 – TRIGGER

GPIO 24/PIN 18 – ECHO (VOLTAGE DIVIDER) 5V – VCC

GND - GND

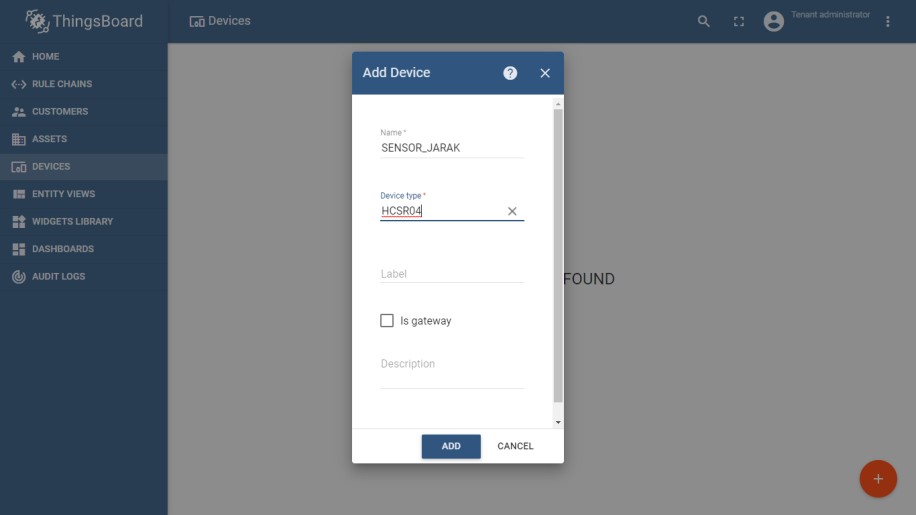
Percobaan 2

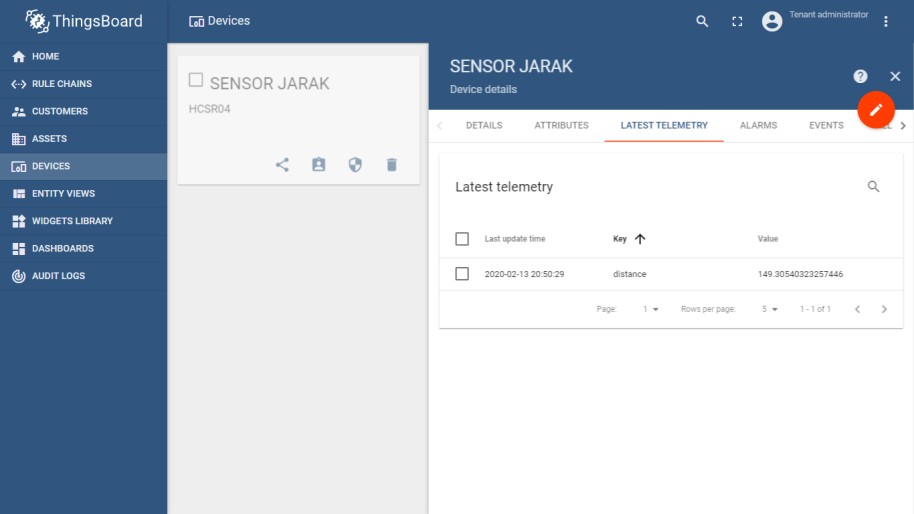
**Halaman Dashboard**

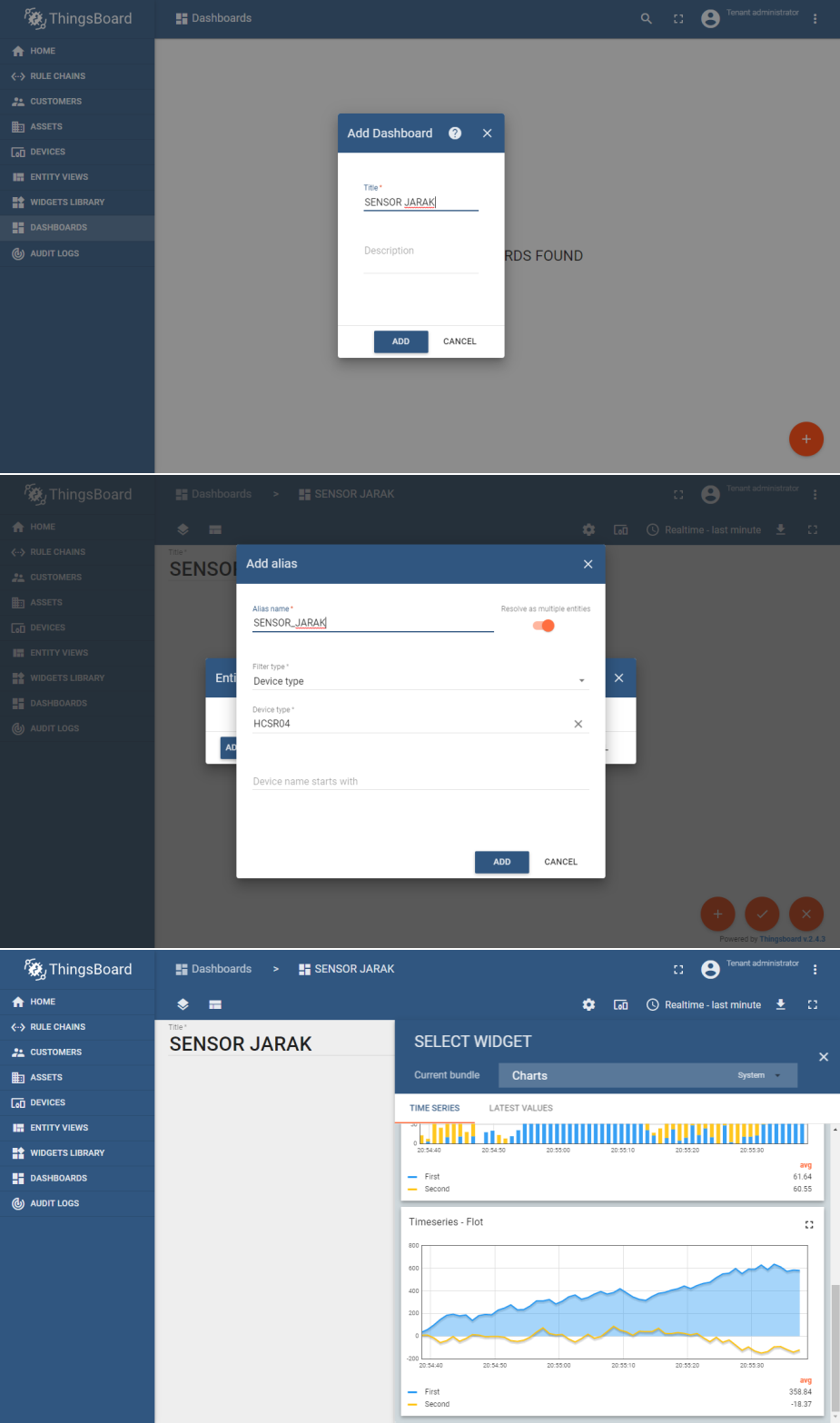
1. Buka dan buat akun di Alamat IP yang ada di papan tulis
2. Masuk dengan user yang sudah tersedia sebagai berikut: Email : [praktikum\_iot\_1@thingsboard.com](mailto:praktikum_iot_1@thingsboard.com)

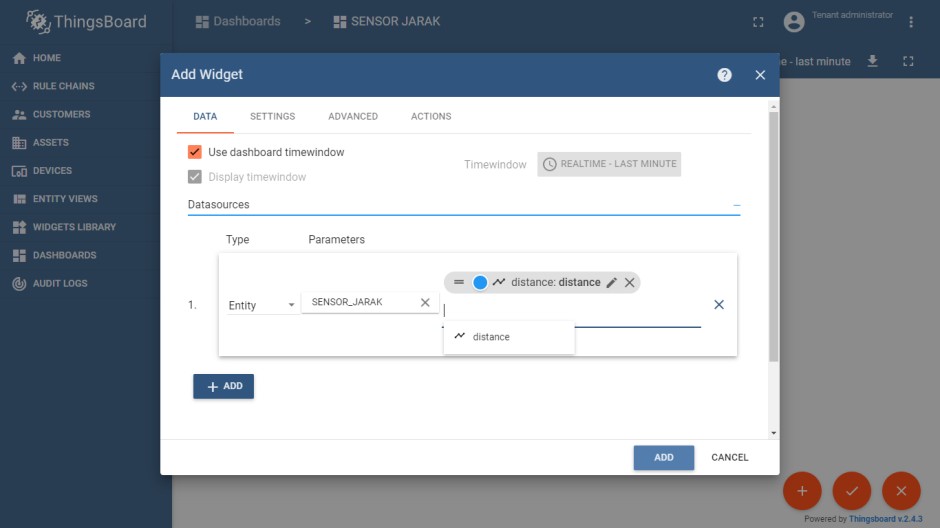
Pass : praktikumiot

\*untuk email yang digunakan sesuai dengan kelompok praktikum, jika kelompok 2 maka emailnya adalah [praktikum\_iot\_2@thingsboard.com,](mailto:praktikum_iot_2@thingsboard.com) dan seterusnya. Untuk password adalah sama.

1. Buatlah device dengan nama berikut:
2. Kemudian copy token dari device dan jalankan program yang ada di **Python.** Kemudian cek data pada device sudah masuk atau belum. Jika sudah kemudian buatlah dashboard.

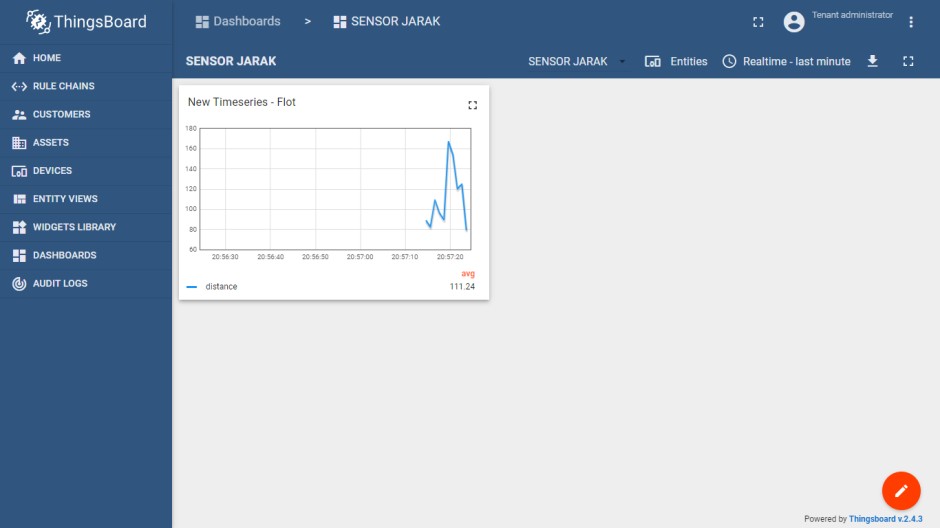


1. Buatlah dashboard seperti berikut ini



Tugas

1. Maka akan muncul dashboard seperti berikut ini apabila sudah berhasil



* 1. Buatlah tampilan dashboard data untuk pembacaan sensor DHT11 dengan Raspberry Pi

|  |  |
| --- | --- |
| DHT-11 Data | Raspberry Pi GPIO 4 |
| DHT-11 VCC | Raspberry Pi 3.3V |
| DHT-11 GND (-) | Raspberry Pi GND |

**sudo apt-get install python-dev**

**git clone https://github.com/adafruit/Adafruit\_Python\_DHT.git cd Adafruit\_Python\_DHT**

**sudo python setup.py install**

**import os import time import sys**

**import Adafruit\_DHT as dht import paho.mqtt.client as mqtt import json**

***# Data capture and upload interval in seconds. Less interval will eventually hang the DHT11.***

**try:**

**while True:**

**humidity,temperature = dht.read\_retry(dht.DHT11, 4) humidity = round(humidity, 2)**

**temperature = round(temperature, 2) print(u"Temperature: {:g}\u00b0C, Humidity:**

**{:g}%".format(temperature, humidity))**

**next\_reading += INTERVAL**

**sleep\_time = next\_reading-time.time() if sleep\_time > 0:**

**time.sleep(sleep\_time) except KeyboardInterrupt:**

**pass**

* 1. Buatlah dashboard yang menampilkan data gauge digital & analog untuk DHT11 dan HCSR04

