|  |
| --- |
| **Laporan Proyek**  **Teknologi IoT**  **Kelompok 2**  **Pendeteksi Hama** |
|  |
| **13321021\_Rico Situmorang**  **13321014\_Asita Tambunan** |

1. **Pendahuluan**

Pada proyek ini kami membuat sistem Pendeteksi hama menggunakan sensor Pir. Dalam implementasinya, kami menggunakan Raspberry Pi sebagai perangkat mikrokontroler. Sensor Pir akan bekerja dengan mengeluarkan bunyi yang dimana bunyi tersebut akan berbunyi yang tergantung pada gerak hama .

1. **Hasil dan Implementasi**
2. **Komponen**

Ada beberapa komponen yang diperlukan untuk membangun sistem “Lampu Otomatis”, sebagai berikut:

* Breadboard
* Raspberry pi 4
* Sensor Pir
* Kabel Jumper

1. **Code program pada raspberry pi 4**

import RPi.GPIO as GPIO

import json

import time

from paho.mqtt import client as mqtt\_client

broker = 'raspberrypi'

port = 1883

topic\_publish = 'motion\_detection\_publish'

topic\_subscribe = 'motion\_detection\_subscribe'

client\_id = 'emqx'

username = 'user1'

password = 'user1'

pin\_pir\_sensor = 17 # GPIO pin for PIR sensor

pin\_buzzer = 18 # GPIO pin for buzzer

GPIO.setwarnings(False)

GPIO.setmode(GPIO.BCM)

GPIO.setup(pin\_pir\_sensor, GPIO.IN)

GPIO.setup(pin\_buzzer, GPIO.OUT)

def get\_motion\_status():

if GPIO.input(pin\_pir\_sensor):

return 1 # Motion detected

else:

return 0 # No motion detected

def control\_buzzer(motion\_status):

if motion\_status == 1:

GPIO.output(pin\_buzzer, GPIO.HIGH)

else:

GPIO.output(pin\_buzzer, GPIO.LOW)

def connect\_mqtt():

def on\_connect(client, userdata, flags, rc):

if rc == 0:

print("Connected to MQTT Broker!")

else:

print("Failed to connect, return code %d\n" % rc)

client = mqtt\_client.Client(client\_id)

client.username\_pw\_set(username, password)

client.on\_connect = on\_connect

client.connect(broker, port)

return client

def publish\_motion\_status(client, motion\_status):

msg = json.dumps({

"motion\_status": motion\_status

})

result = client.publish(topic\_publish, msg)

status = result.rc

if status == 0:

print(f"Motion Status: {motion\_status}, published to topic {topic\_publish}")

else:

print(f"Failed to send message to topic {topic\_publish}")

def on\_message(client, userdata, msg):

print(f"Received {str(msg.payload)} from topic {msg.topic}")

def publish\_subscribe(client):

client.on\_message = on\_message

client.subscribe(topic\_subscribe)

while True:

motion\_status = get\_motion\_status()

control\_buzzer(motion\_status)

print(f"Motion Status: {motion\_status}")

publish\_motion\_status(client, motion\_status)

time.sleep(2)

def run():

client = connect\_mqtt()

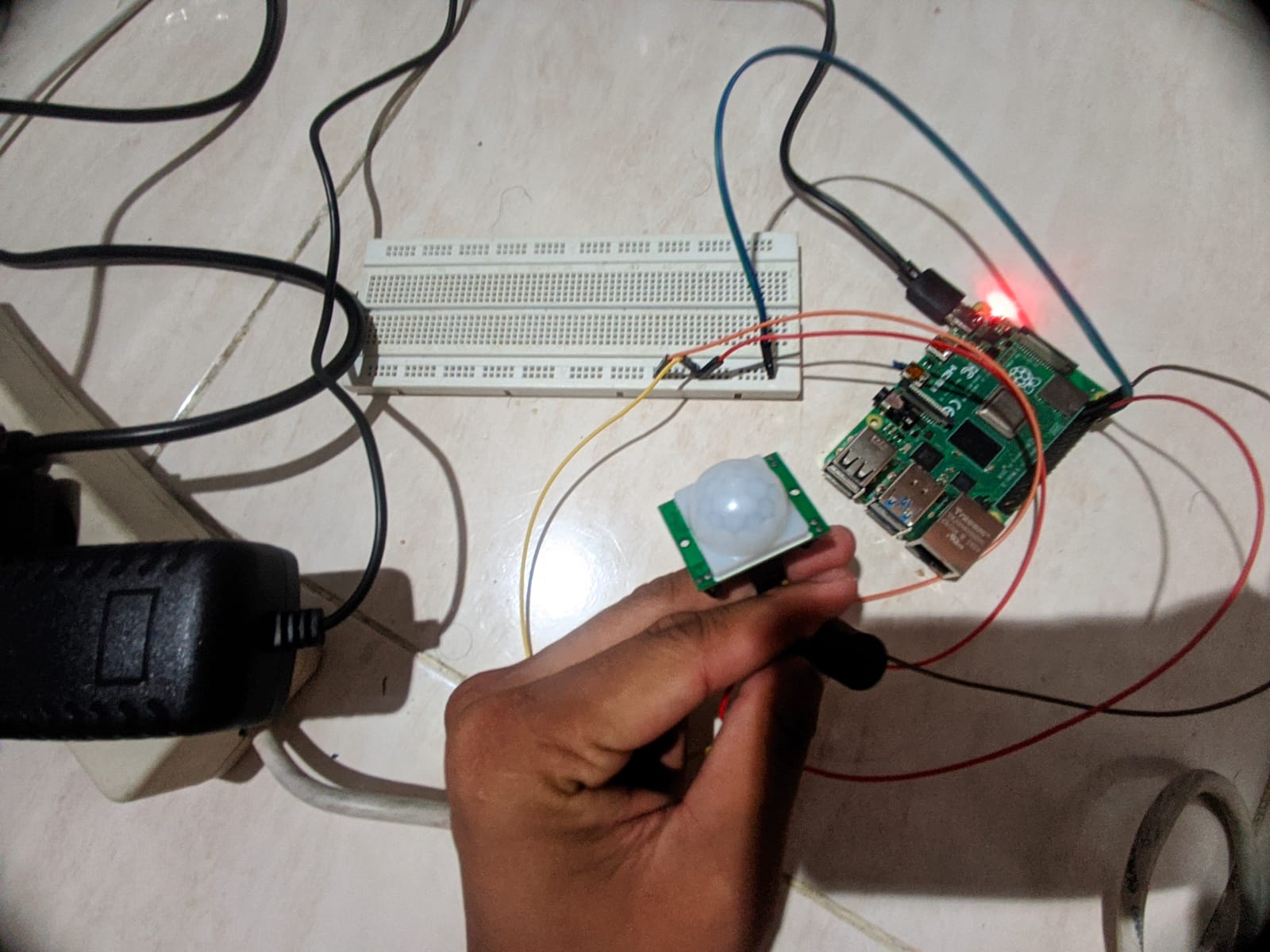
client.loop\_start()

publish\_subscribe(client)

if \_\_name\_\_ == '\_\_main\_\_':

run()

1. **Rangkaian**



**Instalasi EMQX**

Dalam proyek pendeteksi hama dengan sensor PIR, peran utama EMQX sebagai broker MQTT sangat penting. EMQ X memastikan komunikasi efisien antara sensor PIR dan perangkat lain, memungkinkan respons yang cepat terhadap keberadaan hama. Keunggulan EMQ X meliputi integrasi yang lancar, kemampuan pertumbuhan horisontal, dan keamanan data yang solid. Dengan infrastruktur yang handal, EMQ X mendukung keberhasilan proyek dalam menciptakan ekosistem yang efisien dan responsif terhadap tantangan pertanian.







**Instalasi Paho MQTT**

Instalasi Paho MQTT pada proyek pendeteksi hama menggunakan sensor PIR memberikan kemampuan untuk komunikasi efisien antar perangkat dengan menggunakan protokol MQTT. Fungsionalitas Paho MQTT, seperti model publikasi-langganan, manajemen pesan, dan dukungan untuk berbagai bahasa pemrograman, memungkinkan pertukaran informasi deteksi hama yang cepat dan terpercaya. Dengan sifatnya yang sederhana dan mudah digunakan, Paho MQTT untuk pengembangan sistem yang responsif dan terintegrasi dalam proyek ini.









**Instalasi Node red**

Instalasi Node-RED pada proyek pendeteksi hama menggunakan sensor PIR membawa manfaat signifikan. Node-RED memberikan antarmuka visual yang intuitif untuk mengelola aliran data dari sensor PIR, memfasilitasi integrasi dengan broker MQTT, dan memungkinkan pengaturan aksi otomatisasi. Dengan antarmuka yang mudah digunakan dan alat monitoring yang efisien, Node-RED menjadi alat yang berharga untuk merancang solusi yang responsif dan efisien dalam proyek ini.





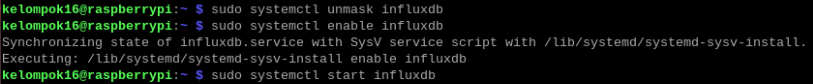
**Instalasi Influxdb**

Instalasi InfluxDB pada proyek pendeteksi hama dengan sensor PIR berperan kunci dalam menyimpan, mengelola, dan menganalisis data waktu nyata. Fungsi utamanya mencakup penyimpanan data historis, kemampuan query fleksibel, visualisasi data, integrasi dengan Node-RED, skalabilitas tinggi, dan pemantauan data. Dengan InfluxDB, proyek dapat efisien menyimpan dan menganalisis data deteksi hama, memberikan wawasan yang diperlukan untuk pengambilan keputusan dan tindakan pencegahan yang efektif.

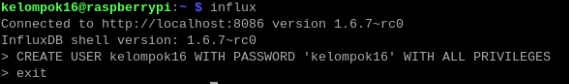


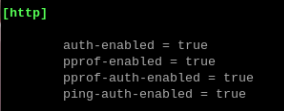


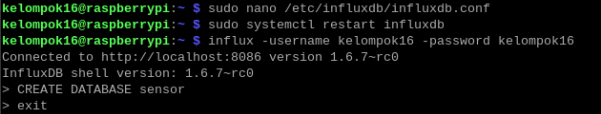












**Instalasi Grafana**

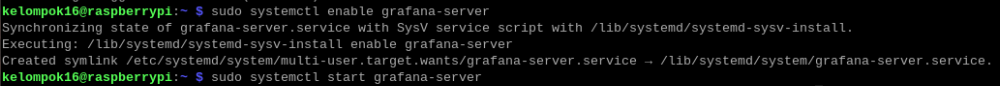
Instalasi Grafana pada proyek pendeteksi hama dengan sensor PIR berfungsi untuk memberikan visualisasi data yang interaktif dan informatif. Grafana memungkinkan pemantauan real-time, dan integrasi yang baik dengan InfluxDB dan Node-RED. Dengan fitur customization, notifikasi, dan peringatan, Grafana meningkatkan pemahaman terhadap keberadaan hama, mendukung pengambilan keputusan yang lebih efektif dalam proyek ini.





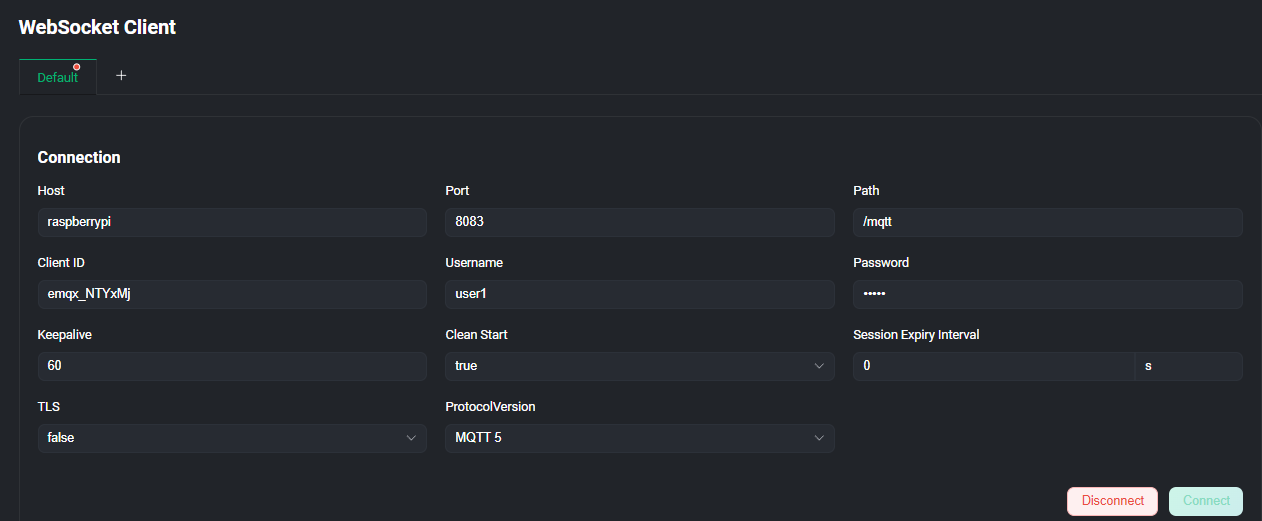


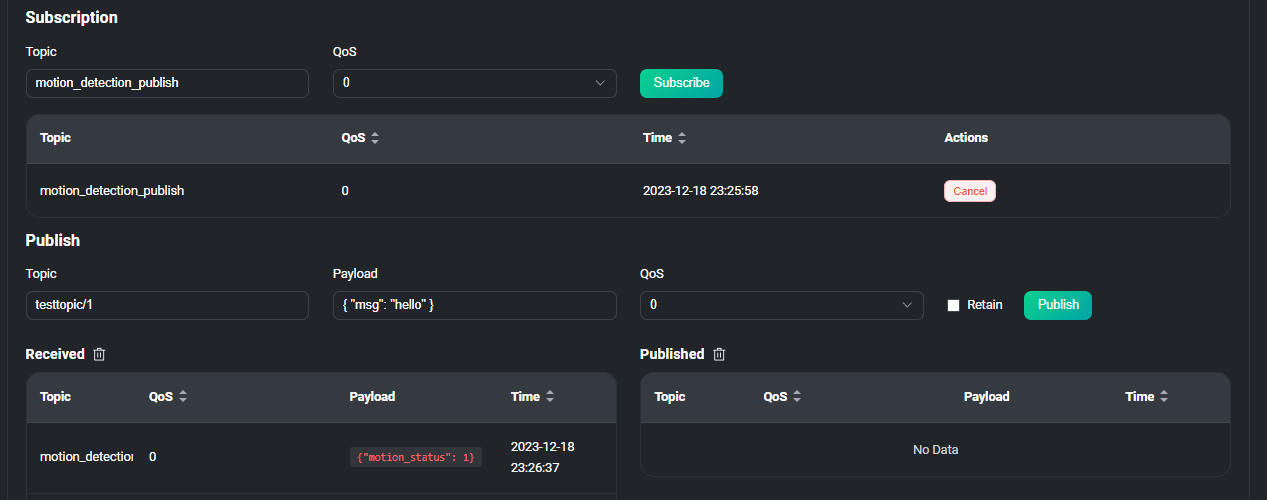




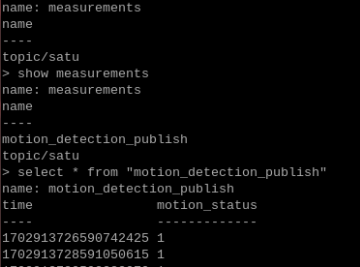
**Keberhasilan Proyek**

Emqx

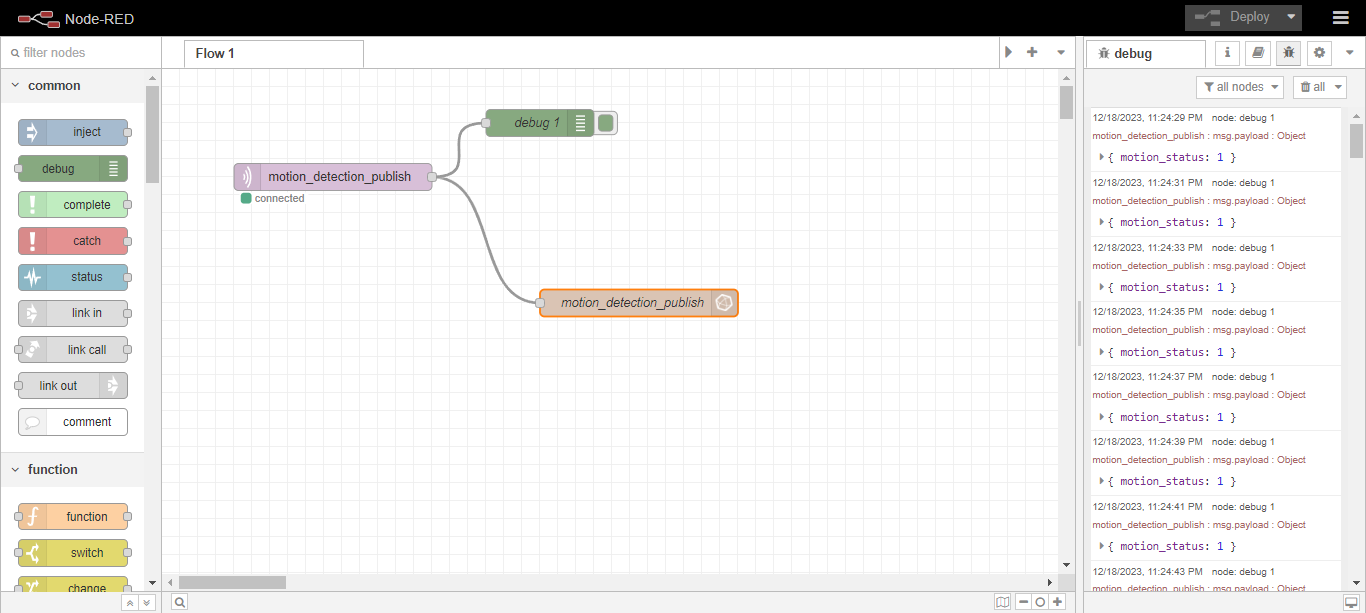




Influxdb



Node Red



Grafana

