

SISTEM LAMPU OTOMATIS BERBASIS IOT DENGAN SENSOR GERAK TUBUH

Mohammad Baharudin Yusuf¹, Fakhri Rifqi Fahreza², M. Subhakti A. Maulana³ Susilawati⁴

Fakultas Ilmu Komputer, Universitas Singaperbangsa Karawang, Kabupaten Karawang^{1,2,3,4}

Email: yusufbaharudin33@gmail.com¹, fakhrirezafahreza@gmail.com²,

soebhaktimoelana03@gmail.com³ susilawati.sobur@staff.unsika.ac.id⁴

Informasi	Abstract
Volume : 2 Nomor : 11 Bulan : November Tahun : 2025 E-ISSN : 3062-9624	<i>The development of Internet of Things (IoT) technology enables electronic devices to be controlled automatically and remotely. One of its applications is an automatic lighting system that regulates lamp activation based on the presence of an object. In this study, an IoT-based automatic lamp system was designed using an infrared sensor (IR Obstacle Avoidance) as a motion detector and an ESP32 as the main controller connected to an MQTT broker. Sensor data are sent to the HiveMQ broker and displayed through the MQTT Dash application on a smartphone. The test results show that the system can automatically turn on the lamp when the sensor detects an object and provide real-time feedback through MQTT. The implementation of this system can save energy and improve lighting control efficiency in home and office environments.</i> Keyword: IoT, ESP32, Infrared Sensor, MQTT, Relay, Smart Home.

Abstrak

Perkembangan teknologi Internet of Things (IoT) memungkinkan pengendalian perangkat elektronik secara otomatis dan jarak jauh. Salah satu penerapannya adalah sistem lampu otomatis yang berfungsi mengatur nyala lampu berdasarkan keberadaan objek. Pada penelitian ini dirancang sistem lampu otomatis berbasis IoT menggunakan sensor infrared (IR Obstacle Avoidance) sebagai pendeteksi gerakan dan ESP32 sebagai pengendali utama yang terhubung ke broker MQTT. Data sensor dikirim ke broker HiveMQ dan ditampilkan melalui aplikasi MQTT Dash di smartphone. Hasil pengujian menunjukkan sistem mampu menyalakan lampu secara otomatis ketika sensor mendeteksi objek, serta memberikan umpan balik real-time melalui MQTT. Penerapan sistem ini dapat menghemat energi dan meningkatkan efisiensi kontrol lampu di lingkungan rumah maupun kantor.

Kata Kunci: IoT, ESP32, Sensor Infrared, MQTT, Relay, Smart Home.

A. PENDAHULUAN

Kebutuhan efisiensi energi listrik semakin meningkat seiring dengan berkembangnya konsep **smart home**. Salah satu masalah utama dalam konsumsi listrik adalah pemborosan akibat lampu yang dibiarkan menyala tanpa kehadiran pengguna. Untuk mengatasi hal tersebut, diperlukan sistem otomatisasi yang mampu mengatur nyala lampu berdasarkan keberadaan manusia atau objek di sekitar ruangan.

Internet of Things (IoT) memungkinkan pengendalian lampu secara cerdas dan terhubung melalui jaringan internet. Dengan protokol **MQTT (Message Queuing Telemetry**

Transport), sistem dapat mentransmisikan data sensor ke server secara ringan dan cepat. Dalam penelitian ini, digunakan **ESP32**, **sensor infrared MH-Sensor**, dan **modul relay** untuk mengontrol nyala lampu. Tujuan penelitian ini adalah mengimplementasikan sistem lampu otomatis berbasis IoT menggunakan MQTT agar dapat menghemat energi dan memberikan kenyamanan pengguna.

B. METODE PENELITIAN

Studi Literatur

Kajian teori dilakukan terkait konsep IoT, arsitektur MQTT, serta penggunaan ESP32 dan sensor IR sebagai sistem deteksi otomatis. Referensi diperoleh dari jurnal ilmiah, dokumentasi resmi ESP32, dan sumber daring seperti HiveMQ serta Arduino.

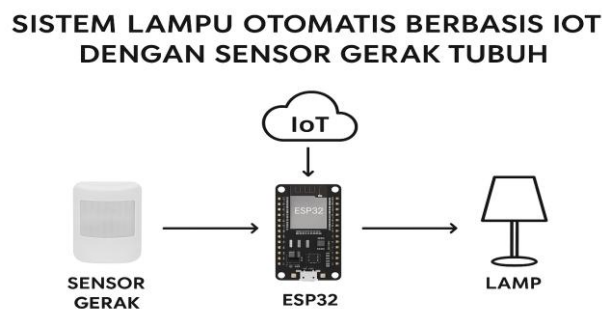
Perancangan Sistem

Alur kerja sistem dapat dijelaskan sebagai berikut:

Sensor Infrared → ESP32 → Relay → Lampu → Broker MQTT → Aplikasi Smartphone (MQTT Dash)

Saat sensor mendeteksi objek, ESP32 mengirimkan data ke broker MQTT dan menyalakan lampu melalui relay. Jika tidak ada deteksi, lampu akan mati otomatis setelah waktu tertentu.

Diagram Sistem:



Gambar 1. Diagram Sistem.

Implementasi Hardware

Perangkat keras yang digunakan meliputi:

- **ESP32 Dev Board** sebagai mikrokontroler dan modul Wi-Fi



Gambar 2. ESP32 Dev Board.

- **Sensor Infrared MH-Sensor** untuk deteksi gerakan



Gambar 3. Sensor Infrared MH-Sensor.

- **Modul Relay 1 Channel** sebagai saklar elektronik lampu



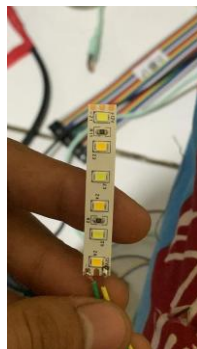
Gambar 4. Modul Relay 1 Channel.

- **Adaptor 12V 2A** untuk suplai daya



Gambar 5. Adaptor 12V 2A.

- **Lampu LED Strip 12V** untuk penerapan sistem lampu otomatis



Gambar 5. Lampu LED.

- **Kabel Jumper** untuk koneksi rangkaian



Gambar 6. Kabel Jumper.

● **Tangga Kardus** untuk media lampu



Gambar 7. Tangga Kardus.

Pemrograman dan Koneksi MQTT

Pemrograman dilakukan menggunakan **Arduino IDE** dengan library WiFi.h dan PubSubClient.h.

ESP32 akan:

1. Terhubung ke Wi-Fi.
2. Mengirim data pembacaan sensor ke broker broker.hivemq.com.
3. Menerima umpan balik untuk mematikan atau menyalakan lampu melalui topik MQTT.

Potongan kode utama:

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

// =====
// === KODE
// =====

#define SENSOR_PIN 4 // Pin sensor IR

// Pin relay
#define RELAY1 18
#define RELAY2 19
#define RELAY3 26
#define RELAY4 22
#define RELAY5 23
```

```
#define WAKTU_NYALA 5000 // Lama lampu menyala (ms)
#define JEDA_NYALA 800 // Jeda antar relay saat NYALA (ms)
#define JEDA_MATI 300 // Jeda antar relay saat MATI (ms)
```

```
// Relay aktif LOW
#define AKTIF_HIGH false
```

```
bool lampuNyala = false;
unsigned long waktuNyalaMulai = 0;
```

```
// --- Fungsi bantu ---
void nyalakanRelay(int pin) {
digitalWrite(pin, AKTIF_HIGH ? HIGH : LOW);
}
```

```
void matikanRelay(int pin) {
digitalWrite(pin, AKTIF_HIGH ? LOW : HIGH);
}
```

```
// =====
// === MQTT & WIFI TAMBAHAN ===
// =====
```

```
// Masukan Koneksi WiFi
const char* ssid = "Indihome";
const char* password = "@indihome123";
```

```
// MQTT Broker
const char* mqtt_server = "broker.emqx.io";
const int mqtt_port = 1883;
const char* mqtt_user = "smarthome";
const char* mqtt_password = "smarthome1234";
```

```
// Topics MQTT
const char* mqtt_topic_pub = "tangga/status";
const char* mqtt_topic_sub = "tangga/control";

WiFiClient espClient;
PubSubClient client(espClient);

// --- Forward declarations ---
void publishStatus(const char* status);
void mqttHandleOnCommand();
void mqttHandleOffCommand();

// --- Koneksi WiFi ---
void setup_wifi() {
  Serial.print("Menghubungkan ke WiFi: ");
  Serial.println(ssid);
  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("\nWiFi Terhubung");
  Serial.print("IP Address: ");
  Serial.println(WiFi.localIP());
}

// --- Callback MQTT ---
void callback(char* topic, byte* payload, unsigned int length) {
  String msg = "";
  for (unsigned int i = 0; i < length; i++) msg += (char)payload[i];
  msg.trim();
```

```
Serial.print("MQTT diterima [");
Serial.print(topic);
Serial.print("]: ");
Serial.println(msg);

if (msg.equalsIgnoreCase("ON")) mqttHandleOnCommand();
else if (msg.equalsIgnoreCase("OFF")) mqttHandleOffCommand();
}

// --- Reconnect MQTT ---
void reconnect() {
while (!client.connected()) {
Serial.print("Menghubungkan ke MQTT...");
if (client.connect("ESP32Tangga", mqtt_user, mqtt_password)) {
Serial.println("Terhubung!");
client.subscribe(mqtt_topic_sub);
} else {
Serial.print("Gagal, rc=");
Serial.print(client.state());
Serial.println(" coba ulang 3 detik...");
delay(3000);
}
}
}

// Publish
void publishStatus(const char* status) {
if (client.connected()) {
client.publish(mqtt_topic_pub, status);
Serial.print("Publish MQTT: ");
Serial.print(mqtt_topic_pub);
Serial.print(" -> ");
Serial.println(status);
}
```



```
    }  
    }  
  
    // Jika ON manual dari MQTT  
    void mqttHandleOnCommand() {  
        if (!lampuNyala) {  
            Serial.println("MQTT -> ON: menyalakan lampu");  
            nyalakanRelay(RELAY1); delay(JEDA_NYALA); client.loop();  
            nyalakanRelay(RELAY2); delay(JEDA_NYALA); client.loop();  
            nyalakanRelay(RELAY3); delay(JEDA_NYALA); client.loop();  
            nyalakanRelay(RELAY4); delay(JEDA_NYALA); client.loop();  
            nyalakanRelay(RELAY5); client.loop();  
  
            lampuNyala = true;  
            waktuNyalaMulai = millis();  
        }  
        publishStatus("ON");  
    }  
  
    // Jika OFF manual dari MQTT  
    void mqttHandleOffCommand() {  
        if (lampuNyala) {  
            Serial.println("MQTT -> OFF: mematikan lampu");  
            matikanRelay(RELAY5); delay(JEDA_MATI); client.loop();  
            matikanRelay(RELAY4); delay(JEDA_MATI); client.loop();  
            matikanRelay(RELAY3); delay(JEDA_MATI); client.loop();  
            matikanRelay(RELAY2); delay(JEDA_MATI); client.loop();  
            matikanRelay(RELAY1); client.loop();  
  
            lampuNyala = false;  
        }  
        publishStatus("OFF");  
    }
```

```
// =====  
// === SETUP ESP32 =====  
// =====  
  
    void setup() {  
        Serial.begin(115200);  
        pinMode(SENSOR_PIN, INPUT);  
  
        pinMode(RELAY1, OUTPUT_OPEN_DRAIN);  
        digitalWrite(RELAY1, HIGH);  
  
        pinMode(RELAY2, OUTPUT);  
        pinMode(RELAY3, OUTPUT);  
        pinMode(RELAY4, OUTPUT);  
        pinMode(RELAY5, OUTPUT);  
  
        matikanRelay(RELAY1);  
        matikanRelay(RELAY2);  
        matikanRelay(RELAY3);  
        matikanRelay(RELAY4);  
        matikanRelay(RELAY5);  
  
        Serial.println("Sistem Lampu Tangga Aktif");  
  
        setup_wifi();  
        client.setServer(mqtt_server, mqtt_port);  
        client.setCallback(callback);  
    }  
  
// =====  
// === LOOP ESP32 =====  
// =====  
  
    void loop() {
```

```
        if (!client.connected()) reconnect();
        client.loop();

    int sensorState = digitalRead(SENSOR_PIN);

    if (sensorState == LOW && !lampuNyala) {
        Serial.println("Gerakan terdeteksi! Lampu nyala otomatis");

        nyalakanRelay(RELAY1); delay(JEDA_NYALA); client.loop();
        nyalakanRelay(RELAY2); delay(JEDA_NYALA); client.loop();
        nyalakanRelay(RELAY3); delay(JEDA_NYALA); client.loop();
        nyalakanRelay(RELAY4); delay(JEDA_NYALA); client.loop();
        nyalakanRelay(RELAY5); client.loop();

        lampuNyala = true;
        waktuNyalaMulai = millis();
        publishStatus("ON");
    }

    if (lampuNyala && (millis() - waktuNyalaMulai >= WAKTU_NYALA)) {
        Serial.println("Timer habis, mematikan lampu otomatis");

        matikanRelay(RELAY5); delay(JEDA_MATI); client.loop();
        matikanRelay(RELAY4); delay(JEDA_MATI); client.loop();
        matikanRelay(RELAY3); delay(JEDA_MATI); client.loop();
        matikanRelay(RELAY2); delay(JEDA_MATI); client.loop();
        matikanRelay(RELAY1); client.loop();

        lampuNyala = false;
        publishStatus("OFF");
    }

    delay(100);
```

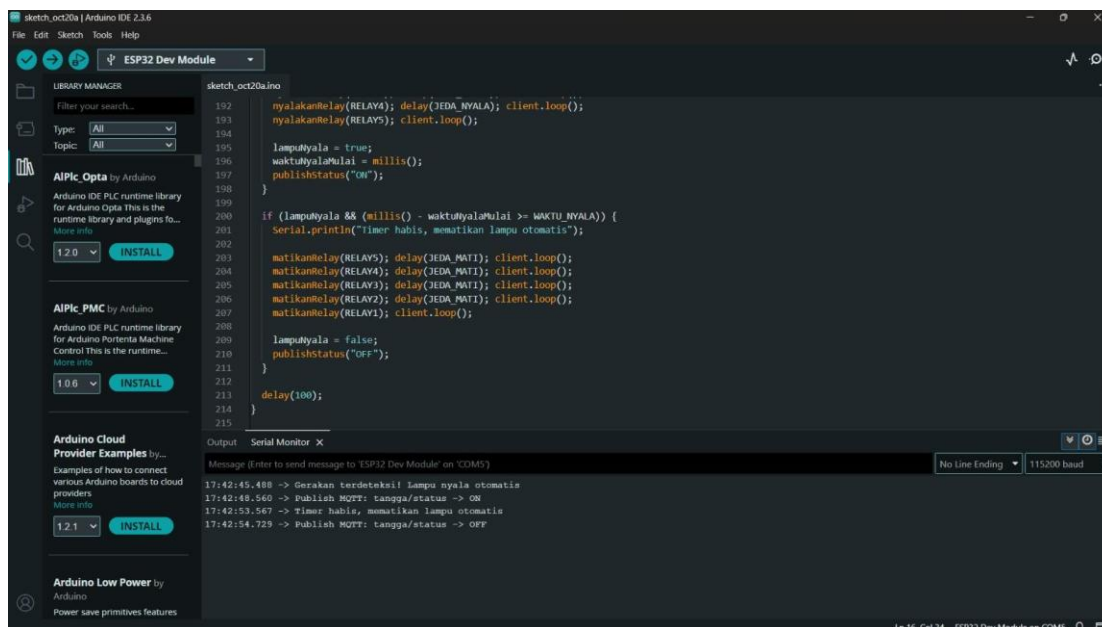
}

C. HASIL DAN PEMBAHASAN

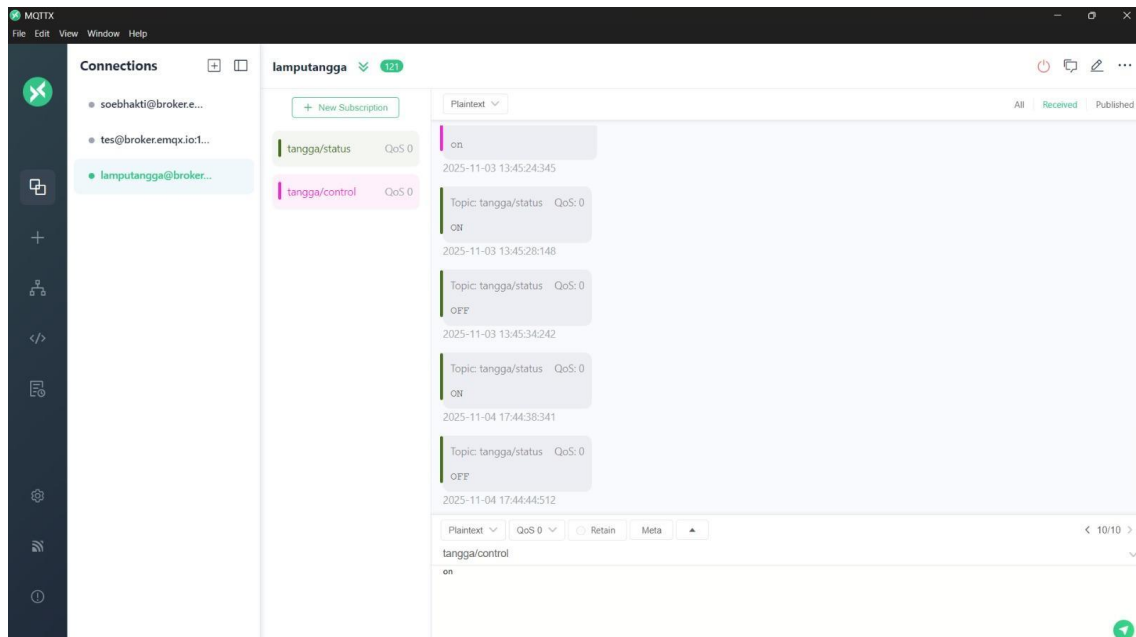
Hasil pengujian menunjukkan sistem dapat bekerja dengan baik:

- Saat sensor IR mendeteksi gerakan atau objek, relay aktif dan lampu menyala.
- Saat tidak ada gerakan, relay nonaktif dan lampu mati otomatis.
- Data status lampu berhasil dikirim ke **broker MQTT HiveMQ** dan tampil di aplikasi **MQTT Dash** secara real-time.

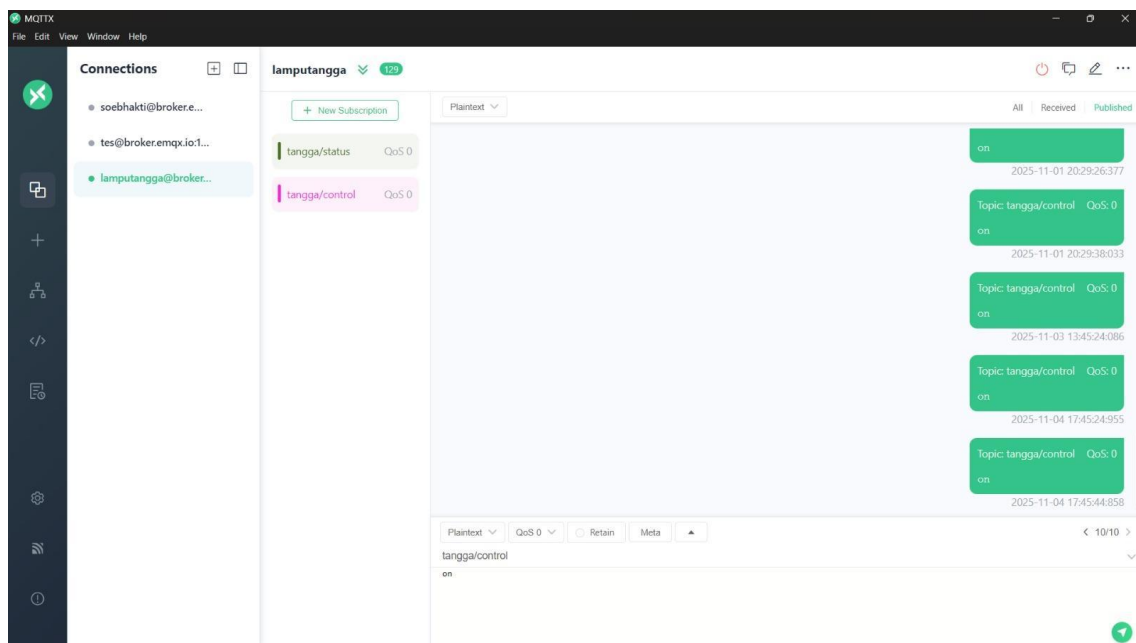
Dari pengujian yang dilakukan, waktu respon sistem rata-rata **<1 detik**, dan stabilitas koneksi MQTT tetap terjaga.



Gambar 8. Output pada serial monitor



Gambar 9. Tampilan pesan di topik: tangga/status pada MQTT



Gambar 10. Tampilan pesan “on” di topik: tangga/control pada MQTT



Gambar 11. Hardware Sistem Lampu Otomatis Berbasis Iot Dengan Sensor Gerak Tubuh



Gambar 12. Lampu menyala saat sensor menangkap gerakan

D. KESIMPULAN

Penelitian ini berhasil mengimplementasikan sistem lampu otomatis berbasis IoT dengan sensor infrared dan koneksi MQTT. Sistem bekerja secara efisien, mampu menyalakan lampu berdasarkan deteksi objek, serta memungkinkan pemantauan dan kontrol jarak jauh melalui smartphone. Teknologi ini terbukti dapat menghemat energi listrik dan mendukung pengembangan smart home berbasis IoT.

E. DAFTAR PUSTAKA

- Susilawati, M. Internet of Things: Menggunakan ESP32 dan Sensor Kelembaban Tanah. Karawang: UNSIKA, 2023.
- I. Marzuki, "Perancangan dan Pembuatan Sistem Penyalan Lampu Otomatis Dalam Ruangan Berbasis Arduino Menggunakan Sensor Gerak dan Sensor Cahaya," Jurnal Penelitian Ilmu Teknik dan Terapan, 2020.
- HiveMQ Documentation. MQTT Essentials. [Online] Available: <https://www.hivemq.com/mqtt-essentials/>
- Espressif Systems. ESP32 Technical Reference Manual. [Online] Available: <https://www.espressif.com>