

## LAMPIRAN

### LAMPIRAN A Kode Program Sistem

```
#include <Wire.h>
#include <RTClib.h> //Inisialisasi RTC
#include <BH1750.h> //Inisialisasi Sensor Intensitas Cahaya
#include <DHT.h> //Inisialisasi Sensor Suhu
#include <Adafruit_NeoPixel.h> // Inisialisasi LED
#include <CTBot.h> //Inisialisasi CTBot
#include <WiFi.h> // Inisialisasi Wifi

#define LED_PIN 27 // Pin yang terhubung ke LED
#define NUM_LEDS 298 // Jumlah Neopixel yang digunakan
#define LED_TYPE NEO_GRB + NEO_KHZ800 // Jenis LED NeoPixel
#define RTC_ADDRESS 0x68 // Alamat I2C untuk RTC
#define BH1750_ADDRESS 0x23 // Alamat I2C untuk sensor cahaya BH1750
#define DHT_PIN 33 // Pin untuk sensor DHT11
#define RELAY_PIN 5 // Pin untuk relay

RTC_DS3231 rtc; // Inisialisasi objek RTC
BH1750 lightMeter(BH1750_ADDRESS); // Inisialisasi objek sensor
cahaya
Adafruit_NeoPixel strip(NUM_LEDS, LED_PIN, LED_TYPE); // Inisialisasi
Jumlah, pin dan type LED
DHT dht(DHT_PIN, DHT11); // Inisialisasi objek sensor suhu dan kelembaban
DHT
CTBot myBot; // Inisialisasi objek CTBot

// ChatID
int64_t chatID = 1130520162; // Ganti dengan ChatID Anda

// Metode regresi linier sensor PH
```

```

int ph1 = 35;
float dph1;
float calibration_value1 = 21.34 + 3; // Kalibrasi sensor PH
unsigned long int avgval;
int buffer_arr[15], temp;
// Configuratioin sensor PH

// WiFi credentials
const char* ssid = "Yurike";
const char* password = "pujakerangajaib";

void setup() {
// Inisialisasi komunikasi I2C
Wire.begin();

// Setup Serial Monitor
Serial.begin(9600);
Serial.println("RTC, Sensor Cahaya BH1750, DHT11, dan Relay");

// Setup RTC
if (!rtc.begin()) {
Serial.println("Tidak dapat menemukan RTC");
while (1);
}

if (rtc.lostPower()) {
Serial.println("RTC berhenti; setting waktu baru...");
rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
}

// Setup sensor cahaya
lightMeter.begin();

```

```
// Setup sensor suhu dan kelembaban DHT11
dht.begin();

// Setup pin untuk relay
pinMode(RELAY_PIN, OUTPUT);

// Setup Neopixel
strip.begin();
strip.show(); // Initialize all pixels to 'off'

// Setup koneksi ke Telegram
myBot.wifiConnect(ssid, password);

myBot.setTelegramToken("7018059660:AAFkS7VdITLHhThypRrRt3ybXbsNn
Q9IXOE");
}

void loop() {
    // Periksa dan sambungkan kembali WiFi jika terputus
    reconnectWiFi();

    // Baca waktu dari RTC
    DateTime now = rtc.now();
    String currentTime = String(now.hour()) + ":" + String(now.minute()) + ":" +
String(now.second());
    Serial.print("Waktu: ");
    Serial.println(currentTime);

    // Baca nilai cahaya dari sensor BH1750
    float lux = lightMeter.readLightLevel();
    Serial.print("Nilai Cahaya: ");
```

```

Serial.print(lux);
Serial.println(" lx");

// Intensitas cahaya
if (lux > 108) {
  // Matikan LED
  strip.clear();
  strip.show();
  Serial.println("Nilai cahaya melebihi 108 lx, LED dimatikan.");
} else if (lux < 32) {
  strip.setBrightness(153);
} else {
  // Setel ke tingkat kecerahan normal
  strip.setBrightness(153);

  // Mode lampu
  for (int i = 0; i < NUM_LEDS; i++) {
    if (i % 2 == 0) {
      strip.setPixelColor(i, strip.Color(255, 0, 0)); // Atur warna merah untuk LED
genap
    } else {
      strip.setPixelColor(i, strip.Color(0, 0, 255)); // Atur warna biru untuk LED
ganjil
    }
  }
  strip.show(); // Tampilkan perubahan
}

// Baca suhu dari sensor DHT11
float temperature = dht.readTemperature();
Serial.print("Suhu: ");
Serial.print(temperature);

```

```

Serial.println(" °C");

// Baca pH air
rph1();
Serial.print("ph1 = ");
Serial.println(dph1);

// Kirim pesan ke Telegram
String message = "Waktu saat ini: " + currentTime + "\nSuhu saat ini: " +
String(temperature) + " °C\nIntensitas Cahaya: " + String(lux) + " lx\nNilai PH Air:
" + String(dph1);

// Logika Temperature Suhu
if (temperature > 30) {
    message += "\nSuhu melebihi 30°C, kipas dihidupkan.";
    myBot.sendMessage(chatID, message);
    digitalWrite(RELAY_PIN, HIGH); // Hidupkan relay
    Serial.println("Suhu melebihi 30°C, Kipas dihidupkan.");
} else {
    message += "\nSuhu normal, kipas dimatikan.";
    myBot.sendMessage(chatID, message);
    digitalWrite(RELAY_PIN, LOW); // Matikan relay
    Serial.println("Suhu normal, kipas dimatikan.");
}

delay(5000); // Tunggu 5 detik sebelum mengirim pesan lagi
}

// Rumus kalibrasi pH air
void rph1() {
    for (int i = 0; i < 10; i++) {
        buffer_arr[i] = analogRead(ph1);
    }
}

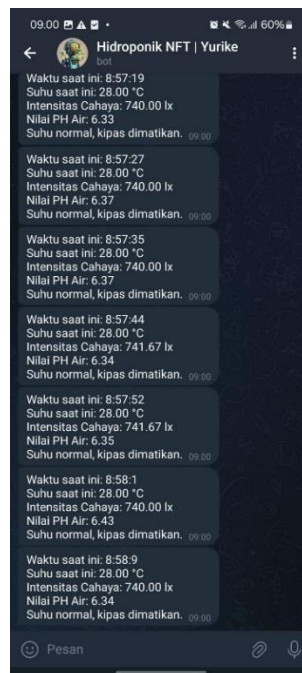
```

```
    delay(30);
}
avgval = 0;
for (int i = 0; i < 10; i++) {
    avgval += buffer_arr[i];
}
float volt = (float)avgval * 3.3 / 4095.0 / 10; // Rumus penggunaan sensor PH
untuk ESP32
dph1 = (-5.70 * volt) + calibration_value1;
}
```

```
void reconnectWiFi() {
if (WiFi.status() != WL_CONNECTED) {
    Serial.println("Terputus dari WiFi, mencoba menyambung kembali...");
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println("\nTersambung ke WiFi");
    myBot.wifiConnect(ssid, password); // Reconnect CTBot to the WiFi
}
}
```

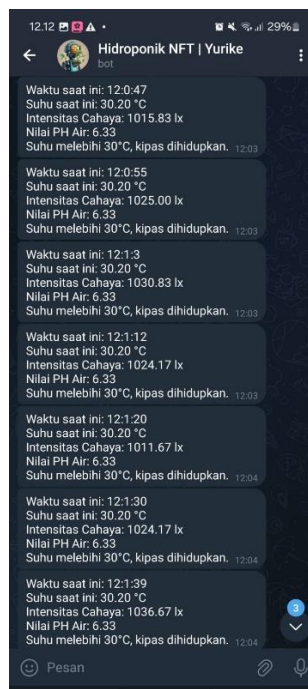
## LAMPIRAN B Tampilan notifikasi telegram bot dari pengujian sistem

### a) Pagi hari



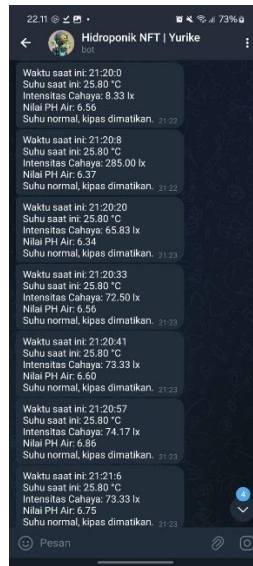
Gambar lampiran notifikasi telegram pengujian keseluruhan sistem pada pagi hari

### b) Siang hari



Gambar lampiran notifikasi telegram pengujian keseluruhan sistem pada siang hari

c) Malam hari



**Gambar lampiran notifikasi telegram pengujian keseluruhan sistem pada malam hari**

**LAMPIRAN C Hidroponik NFT**

a. Siang Hari



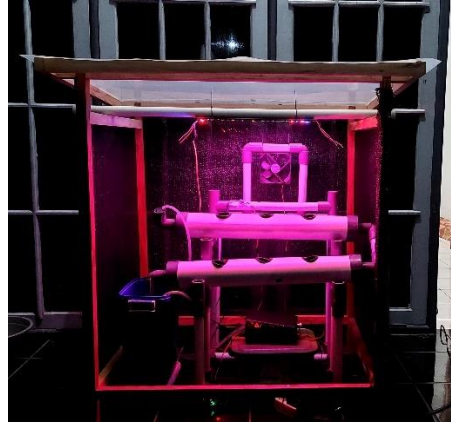
**Gambar lampiran hidroponik *nutrient film technique* saat dibuka**



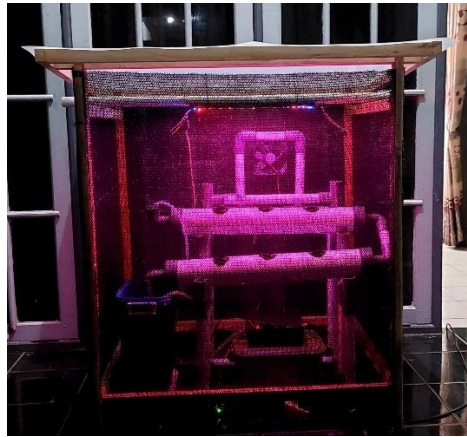
**Gambar lampiran hidroponik *nutrient film technique* saat ditutup**



**b. Malam Hari**

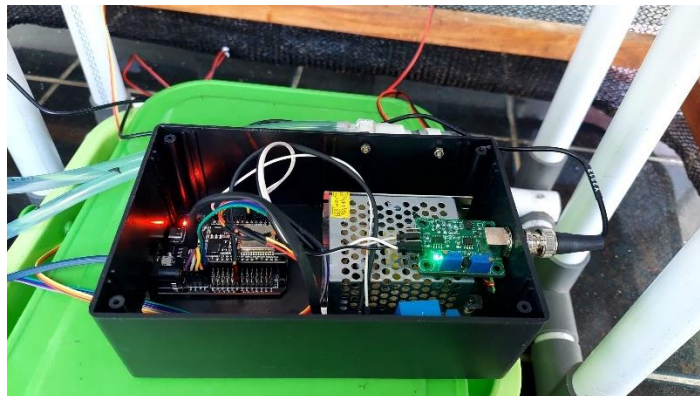


**Gambar lampiran hidropoik *nutrient film technique* saat dibuka**



**Gambar lampiran hidropoik *nutrient film technique* saat ditutup**

**LAMPIRAN D Alat yang dibuat**



**Gambar lampiran alat yang dibuat**