

Lampiran 3 Source Code :

```
#define BLYNK_TEMPLATE_ID "TMPL6A3P5RWmk"
#define BLYNK_TEMPLATE_NAME "ProjekSkripsiIkan"
#define BLYNK_AUTH_TOKEN "0HHZN8vK-60-Ix0lg5ecyWdCnwceoF5m"

#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
#include <Wire.h>

//Notif Di Blynk
String sisapakan_event = "sisapakan";
String tdssensor_event = "tdssensor";

//RTC
#include "RTClib.h"
RTC_DS3231 rtc;
char daysOfTheWeek[7][12] = {"Minggu", "Senin", "Selasa",
"Rabu", "Kamis", "Jum'at", "Sabtu"};
int jam, menit, detik;
int tanggal, bulan, tahun;
String hari;
int StatusPakan;

const char* ssid = "AliefWifi";
const char* pass = "Alief090";

// Blynk auth token from the Blynk app
char auth[] = BLYNK_AUTH_TOKEN;

// Definisikan pin
int buzzerPin = 18;
const int servoPin = 13;
const int trigPin = 14;
const int echoPin = 12;
#define TdsSensorPin 34 // Ubah ke pin ADC yang sesuai,
misalnya 34

//SERVO
#include <ESP32Servo.h>
Servo mekanik;

// Fungsi untuk menghitung jarak menggunakan HC-SR04
long getDistance() {
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);
```

```

    long duration = pulseIn(echoPin, HIGH);
    long distance = (duration / 2) / 29.1;
    return distance;
}

float readTDS() {
    int sensorValue = analogRead(TdsSensorPin);
    float voltage = sensorValue * (3.3 / 4096.0); // Menghitung
    tegangan sensor dari nilai ADC
    float tdsValue = (133.42 * voltage * voltage * voltage -
    255.86 * voltage * voltage + 857.39 * voltage) * 0.5; //
    Menghitung nilai TDS dalam ppm
    return tdsValue;
}

void setup() {
    Serial.begin(115200);

    // Connecting to WiFi
    Serial.print("Connecting to WiFi");
    WiFi.begin(ssid, pass);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println();
    Serial.println("WiFi connected");
    Serial.print("IP address: ");
    Serial.println(WiFi.localIP());

    // Connecting to Blynk
    Blynk.begin(auth, ssid, pass);
    while (Blynk.connect() == false) {
        // Wait until connected to Blynk
    }
    Serial.println("Blynk Terkoneksi");

    // Inisialisasi pin
    pinMode(trigPin, OUTPUT);
    pinMode(echoPin, INPUT);
    pinMode(buzzerPin, OUTPUT);

    //RTC
    if (!rtc.begin()) {
        Serial.println("Couldn't find RTC");
        Serial.flush();
        while (1) delay(10);
    }
    //rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
    // rtc.adjust(DateTime(2014, 1, 21, 3, 0, 0));
}

```



```

//SERVO
mekanik.attach(servoPin);
mekanik.write(0);

while (!Serial);
}

void loop() {
  Blynk.run();

  //RTC
  DateTime now = rtc.now();
  jam = now.hour();
  menit = now.minute();
  detik = now.second();
  tanggal = now.day();
  bulan = now.month();
  tahun = now.year();
  hari = daysOfTheWeek[now.dayOfTheWeek()];
  Serial.println(String() + hari + ", " + tanggal + "-" + bulan
+ "-" + tahun);
  Serial.println(String() + jam + ":" + menit + ":" + detik);
  //delay(1000);

  // Send RTC data to Blynk
  Blynk.virtualWrite(V4, jam); // Send hours to V4
  Blynk.virtualWrite(V5, menit); // Send minutes to V5
  Blynk.virtualWrite(V6, detik); // Send seconds to V6

  //KONDISI
  if ((jam == 12) && (menit == 48) && (detik == 0)) {
    kasih_pakan(1);
  }
  if ((jam == 12) && (menit == 48) && (detik == 15)) {
    kasih_pakan(1);
  }
  if ((jam == 12) && (menit == 48) && (detik == 30)) {
    kasih_pakan(1);
  }

  // Cek kondisi pakan setiap detik
  long distance = getDistance();
  Serial.print("Distance: ");
  Serial.print(distance);
  Serial.println(" cm");

  // Jika jarak lebih dari 10 cm (asumsikan pakan habis)
  if (distance > 15) {
    String text = "Pakan Telah Habis,Silahkan Isi Kembali";
    Serial.println(text);
  }
}

```

```

    Blynk.logEvent(sisapakan_event, text);
    Serial.println("Pakan habis!");
    tone(buzzerPin, 440);
    digitalWrite(buzzerPin, HIGH); // Nyalakan buzzer
    Blynk.virtualWrite(V3, 440);
    delay(3000);
} else {
    digitalWrite(buzzerPin, LOW); // Matikan buzzer
    noTone(buzzerPin);
    Blynk.virtualWrite(V3, 0);
    //delay(1000);
}
Blynk.virtualWrite(V0, distance);

// Pembacaan nilai TDS
float tdsValue = readTDS();
Serial.print("Nilai TDS: ");
Serial.print(tdsValue);
Serial.println(" ppm");

// Kondisi untuk mengecek kualitas air kolam lele
if (tdsValue <= 300) {
    Serial.println("Kualitas air baik untuk kolam lele.");
} else {
    Serial.println("Kualitas air buruk untuk kolam lele.");
    String text = "Kualitas air kolam buruk,Mohon dikuras";
    Serial.println(text);
    Blynk.logEvent(tdssensor_event, text);
}
Blynk.virtualWrite(V2, tdsValue);
delay(1000); // Tambahkan delay untuk membaca nilai TDS
secara periodik
}

void kasih_pakan(int jumlah) {
    // Buka tempat pakan
    Serial.println("Feeding time!");
    for (int i = 1; i <= jumlah; i++) {
        mekanik.write(150);
        delay(3000);
        mekanik.write(0);
        delay(3000);
    }
}

// Function to control servo from Blynk using a switch button
BLYNK_WRITE(V1) {
    int switchState = param.asInt(); // Get switch state from
Blynk app
    if (switchState == 1) {
        mekanik.write(150); // Open position

```



```
Serial.println("Servo Terbuka");  
} else {  
mekanik.write(0); // Close position  
Serial.println("Servo Tertutup");  
}  
}
```

