
PENGEMBANGAN AUTOMATIC FEEDER PAKAN IKAN MENGGUNAKAN MICROCONTROLLER ESP8266

Very Kusuma Pela^{*1}, I Gede Juliana Eka Putra², I Gede Putu Krisna Juliharta³

^{1,2,3}Teknik Informatika, STMIK Primakara

E-mail: kusumavery019@gmail.com

Abstrak: Internet of Thing (IoT) merupakan perpaduan antara internet dan ubiquitous computing yang memiliki fungsi untuk mengendalikan berbagai macam barang elektronik dari jarak jauh dan bisa terhubung dengan berbagai macam sensor. Teknologi dapat dimanfaatkan untuk masalah pemberian pakan ikan secara otomatis yang bisa dimonitoring dari aplikasi blink berbasis android dan IOS. Pada Penelitian ini studi kasus pada ikan lele dengan menggunakan metode penelitian prototype mulai dari tahap pengumpulan kebutuhan sampai tahap penggunaan. Bahan yang digunakan dalam penelitian ini adalah dari besi dan prabotan rumah tangga sebagai konstruksi, Node Mcu Esp8266, Arduino, Loadcell, sensor ultrasonic, Motor Servo, serta Motor DC. Alat yang dibuat bekerja otomatis berjalan memberikan pakan pada jam yang telah ditentukan serta mengukur pemberian pakan berdasarkan berat yang ditentukan dengan jarak lontaran pakan mencapai 1 meter, dan dapat dimonitor dengan menggunakan aplikasi blynk melalui jaringan wifi local untuk mengetahui pakan telah ditebar dan jumlah ketersediaan pakan pada penampungan utama.

Kata Kunci: *Internet of Thing, arduino, Esp8266, pemberian pakan otomatis, ikan lele.*

Abstract: *The Internet of Thing (IoT) is a combination of the internet and ubiquitous computing which has the function of controlling various kinds of electronic goods remotely and can be connected to various kinds of sensors. Technology can be used for the problem of feeding fish automatically which can be monitored from the blink application based on Android and IOS. The materials used in this study are iron and household furniture as construction, Mcu Esp8266 Node, Arduino, Loadcell, ultrasonic sensors, Servo Motors, and DC Motors. Tools that are made to work automatically run provide feed at predetermined hours and measure feed based on weight determined with a distance of ejection of feed up to 1 meter, and can be monitored using the blynk application via a local wifi network to find out the feed has been stocked and the amount of feed available in the main shelter.*

Keywords: *Internet of Thing, arduino, Esp8266, pemberian pakan otomatis, ikan lele.*

1. PENDAHULUAN

Internet of Thing (IoT) merupakan perpaduan antara internet dan ubiquitous computing. Internet of Thing (IoT) melibatkan antar berbagai macam sensor, agregator, aktuator dan berbagai macam aplikasi serta berbagai macam domain. Pada dasarnya Internet of Thing terdiri dari dua komponen utama yaitu internet dan thing [1]. Internet of Thing (IoT) memiliki berbagai macam manfaat untuk mengendalikan perangkat elektronik sehingga dapat memudahkan manusia dalam untuk memantau dan memonitoring suatu alat atau benda melalui jaringan atau internet serta untuk mengefisienkan tenaga dan waktu. Konsep internet of thing sangat sederhana dengan cara kerja mengacu pada 3 elemen utama pada arsitektur IoT, yakni: Barang Fisik yang dilengkapi modul IoT, Perangkat Koneksi ke Internet seperti Modem dan Router Wireless [2].

Ikan lele merupakan salah satu ikan yang banyak dikonsumsi oleh masyarakat Indonesia dari data Warta Pasar Ikan, 2006, tercatat sekitar 5000 orang anggota pengusaha warung tenda pecel lele yang tersebar di daerah jabodetabek dengan kebutuhan ikan lele sekitar 40 ton/ hari [3]. Dengan data tersebut merupakan suatu potensi besar bagi petani ikan untuk industri penunjang kebutuhan baik di tingkat rumah tangga maupun sampai di jual ke supplier maupun restaurant, hal ini menjadi tantangan untuk petani ikan untuk memaksimalkan hasil produksi, biaya produksi dan waktu.

Berdasarkan hasil riset awal yang dilakukan dengan cara wawancara langsung pada pemilik kolam bioflokmina citta taman sari yang terletak di desa agantiga, kecamatan manggis, kabupaten karangasem – bali. Adapun kendala kendala dihadapi oleh petani ikan adalah pemberian pakan yang banyak menghabiskan waktu, karena banyak

dari usaha ini dijadikan sampingan serta yang fokus pada industri ini juga memiliki banyak kolam yang juga memerlukan waktu yang tidak sedikit.

Dari permasalahan tersebut maka peneliti telah berhasil membuat alat automatic feeder pakan ikan otomatis dengan menggunakan Node Mcu Esp8266 yang dapat dimonitoring melalui aplikasi blynk pada mobile Android dan IOS dengan jaringan wi-fi. Dalam penelitian ini peneliti menggunakan metodologi penelitian prototype yang dimulai tahap pengumpulan data sampai ke tahap penggunaan.

2. TINJAUAN PUSTAKA

Beberapa penelitian terdahulu yang penulis jadikan acuan antara lain penelitian dari Arief Wahyu Wicaksono, Dahniel Syaury dan Mochamad Hannats Hanafi Ichsan tahun 2019 yang berjudul "Sistem Embedded untuk Pengaturan Pakan Ikan Mas Koki berdasarkan Banyaknya Jumlah Ikan dan Ketinggian Air". Hasil dari penelitian ini adalah alat pemberian pakan ikan berdasarkan jumlah banyak ikan dan ketinggian air yang mempermudah pemberian makan ikan berdasarkan jumlah ikan yaitu secara otomatis sesuai dengan yang telah ditentukan.

Penelitian dari Kardiman, Dono martono, Rizal Hanifi, dan Eri Widiandot tahun 2019 yang berjudul "Pengembangan Sistem Kontrol Alat Penebar Pakan Ikan Otomatis Dengan Sumber Energi Matahari". Hasil dari penelitian ini adalah pengembangan alat penebar pakan ikan menggunakan panel surya atau energi matahari yang sebelumnya hanya menggunakan baterai.

Penelitian dari Heri Ngarianto dan Alexander tahun 2020 yang berjudul "Pengembangan Automatic Pet Feeder Menggunakan Platform Blynk Berbasis Mikrokontroler ESP8266". Hasil dari penelitian ini adalah alat pemberi makan ikan otomatis yang dapat diatur waktu pemberian pakannya dan takaran pemberian pakannya menggunakan Arduino Uno dan Arduino Mega 2560.

Beberapa penelitian diatas penulis jadikan acuan untuk membuat pengembangan automatic feeder pakan ikan menggunakan microcontroller ESP8266 yang

dapat bekerja otomatis berjalan memberikan pakan pada jam yang telah ditentukan

3. METODE

Pada penelitian ini menggunakan metode penelitian Prototype yang terbagai dalam 7 fase atau tahap penelitian yaitu:



Gambar 1. Metodologi Prototype.

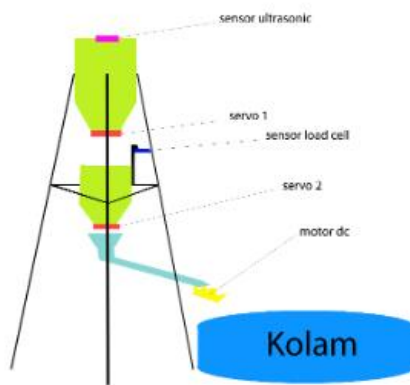
1. Pengumpulan kebutuhan, pada tahap ini Pelanggan dan pengembang bersama-sama mendefinisikan format dan kebutuhan perangkat lunak secara keseluruhan, mengidentifikasi semua kebutuhan, dan garis besar sistem yang akan dibuat.
2. Membangun prototyping, yang dilakukan pada tahap ini adalah membuat desain sementara yang berpusat pada penyajian kepada narasumber.
3. Evaluasi prototyping, jika prototype yang sudah dibangun sesuai dengan keinginan dari narasumber yang diwawancarai maka dilakukan evaluasi. Jika sudah sesuai akan dilanjutkan ke tahap pengkodean (coding) namun jika belum sesuai maka akan diperbaiki dengan mengulang kembali tahap 1, tahap 2 dan tahap 3.
4. Mengkodekan Sistem, tahap ini lakukan jika prototype yang dibuat sudah diterima dan disepakati yang dilanjutkan dengan pengkodean dengan bahasa pemrograman Arduino IDE,
5. Menguji Sistem, tahap ini dilakukan untuk mengetahui performance dari sistem yang dibangun. Pengujian sistem dilakukan dengan metode Black Box.
6. Evaluasi Sistem, tahap ini dilakukan untuk mengetahui apakah sistem yang dibangun telah sesuai dengan yang diharapkan. Jika sudah memenuhi dapat dilakukan tahap selanjutnya, namun jika tidak maka perlu dilakukan pengulangan pada tahap 4 dan tahap 5. Menggunakan Sistem, yaitu sistem yang dibangun siap

digunakan oleh pengusaha penggemukan lele.

Teknik pengumpulan data dalam penelitian ini dilakukan dengan studi literatur dan wawancara. Studi literatur dilakukan untuk memperoleh referensi untuk pengembangan automatic feeder untuk pakan ikan otomatis menggunakan Node Mcu ESP8266. Studi literatur yang digunakan adalah jurnal yang berkaitan Internet of Thing (IoT) dengan pemberian pakan otomatis.

4. HASIL DAN PEMBAHASAN

4.1. Perancangan Alat

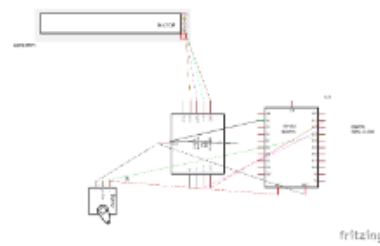


Gambar 2. Desain Alat Automatic Feeder

Dari gambar 2 diatas dapat dijelaskan desain perancangan hardware dan sensor sebagai berikut:

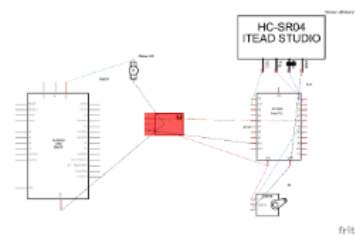
1. ultrasonic untuk mengukur ketersediaan pakan pada penampungan.
2. Servo motor 1 digunakan untuk membuka penampungan utama saat servo menerima perintah untuk membuka dari arduino sesuai dengan jam yang telah ditentukan dengan memutar 90°
3. Sensor load cell akan menghitung berat, saat jumlah satuan ukur sudah diterima sistem maka secara otomatis akan menutup servo 1, secara otomatis berganti servo 2 akan membuka dengan memutar 90°
4. Setelah servo 2 bergerak maka motor DC akan ikut bergerak untuk menebar pakan ikan lele ke kolam. Desain perancangan sistem terbagi menjadi.

2 skematik untuk menyesuaikan permasalahan yang dialami



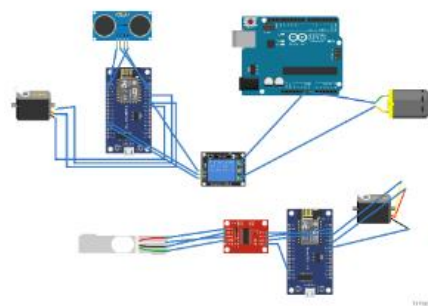
Gambar 3. Skematik Pertama

Skematik pertama digunakan untuk membuka servopenampungan pakan utama dan load cell digunakan untuk mengukur massa berat pada penampungan pakan ikan yang akan ditebar.



Gambar 4. Skematik kedua

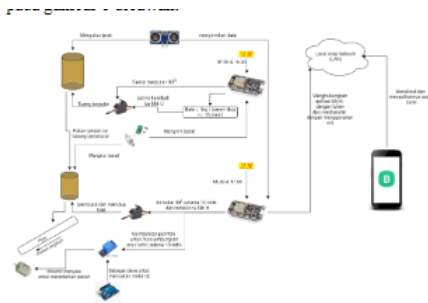
Skematik kedua digunakan untuk menyalakan servodan relay pada jam yang telah ditentukan serta untuk mengukur ketersediaan pakan pada tabung pakan ikan utama. Sedangkan anduino unosebagai sumber daya listrik untuk menyalakan Motor DC. Dari skematik pertama dan kedua menjadi rangkaian komponen hardware seperti pada gambar 5 dibawah :



Gambar 5. Desain rangkaian perancangan alat

Dari rangkaian perancangan alat pada gambar 5 diatas digambar arsitektur

cara kerja dari automatic feeder pakan ikan menggunakan Esp8266 seperti pada gambar 6 dibawah:



Gambar 6. Arsitektur alat

Dari gambar 6. diatas dapat dijelaskan cara kerja sistem dalam penelitian ini adalah sebagai berikut:

- 1 Pada pukul 07.00 Wita Node Mcu Esp 8266 mengirimkan perintah memutar servo 1 untuk berputar 900 untuk mengikir tabung penakaran.
- 2 Load cell secara terus menerus mengukur dan mengirimkan data berat yang di peroleh ke Node Mcu Esp 8266.
- 3 Jika Node Mcu Esp 8266 menerima data berat lebih sama dengan 1 kg maka otomatis Node Mcu Esp8266 akan mengirimkan perintah untuk memutar servo 1 kembali ke titik 0 atau menutup.
- 4 Jika Node Mcu Esp 8266 tidak menerima data berat lebih sama dengan 1 kg maka otomatis Node Mcu Esp8266 akan mengirimkan perintah untuk memutar servo 1 kembali ke titik 0 atau menutup.
- 5 Pada Pukul 08.00 wita Node Mcu Esp 8266 akan memerintahkan servo untuk berputar 900 dan relay untuk menyambungkan arus listrik agar motor dc dapat menyala.
- 6 Setelah 10 detik maka secara otomatis Node Mcu Esp8266 akan memerintahkan servo kembali ke titik 0 (menutup) dan memerintahkan relay untuk memutus arus listrik.
- 7 Setelah realtime sensor ultrasonic terus mengirimkan data jumlah jarak ketersediaan data
- 8 Saat proses nomer 6 berjalan sistem akan print data pada serial monitor pakan sedang ditabur dan setelah proses nomer 6 diatas selesai maka sistem

akan print data pada serial monitor berupa pesan pakan telah ditabur.

- 9 Data pada sistem ditransfer melalui jaringan local area network (LAN) menggunakan wifi.
- 10 Dalam sistem terdapat token untuk menghubungkan sistem dengan aplikasi blynk.
- 11 Secara otomatis jika proses diatas sudah berjalan maka user dapat melihat hasil dan jumlah tersediaan pakan melalui aplikasi blynk.

4.2. Implementasi Alat

Hasil dari implementasi pengembangan automatic feeder pakan ikan menggunakan Node Mcu Esp8266 dijelaskan sebagai berikut:



Gambar 7. Alat automatic feeder

Gambar 7 diatas merupakan alat yang berhasil dibuat untuk pemberian pakan otomatis menggunakan Node Mcu Esp 8266 yang diuji coba pada kolam bioflox



Gambar 7. Hardware electronic

Gambar 7 diatas merupakan rangkaian Node Mcu Esp8266 yang dioperasikan dengan menggunakan tenaga listrik dc untuk menyalakan sensor dan dll. Untuk memantau alat dari mobile android dan IOS menggunakan aplikasi blynk yang merupakan yang dikemabangkan untuk project-project IoT dimana pengguna dapat membuat dan memprogram aplikasi untuk menjalankan atau menjadi remote dari project Internet of Think (IoT).

Pada Gambar 8. diatas dapat dijelaskan pesan yang disampaikan dari Node Mcu ke Aplikasi Blynk yang dirancang untuk memonitoring atau memantau keadaan alat yang sedang bekerja.

4.3. Blackbox Testing

Pengujian dilakukan alat dilakukan dengan menggunakan metode blackbox testing dengan paparan sebagai berikut:

N o.	Yang Diuji	Yang Harapan	Hasil Pengamatan	Kesimpulan
1	Pemindahan Pakan kering ke tabung penaburan pakan ikan	Pakan ikan kering dapat jatuh bergetah ke tabung penampung an pakan yang akan ditabur	Pakan ikan yang kering dapat bergetah ke tabung penaburan pakan dengan cepat	Selesai
2	Pendataran pakan kering melalui pipa ke kolam	Pakan ikan kering dapat jatuh dengan cepat dengan jalur pipa ke kipas motor di dengan cepat	Pakan ikan kering dapat jatuh melalui pipa ke kipas motor di dengan cepat	Selesai
3	Pemutaran servo untuk membuka tabung pakan warna	Motor servo pada tabung penampung an pakan warna dapat memutar 90° agar	Motor servo pada tabung penampung an pakan warna dapat berputar 90° untuk membuka tuas	Selesai



Gambar 8. Aplikasi Blynk Feeder

		diatur membuka fase penutup tabung dengan tekanan beban pada penampung an utama	penutup dengan beban tekanan pakan dari penampung pakan utama	
4	Pemutaran servo untuk membuka tabung pemberian	Mesin servo pada tabung penampung an pakan yang diantar dapat bekerja 90° untuk membuka fase penutup an pakan penampung an pakan yang akan diantar	Mesin servo pada penampung an pakan penampung an pakan yang akan diantar dapat bekerja 90° membuka fase penutup an pakan penampung an pakan yang akan diantar	Sesuai
5	Pengukuran beban dengan sensor load cell	Dengan beban 1 1/4 kg dapat memutar servo kembali ke titik 0° untuk memutup fase tabung penampung an utama	Pengujian dilakukan dengan menggunakan alat pembobot konvensional dengan beban 1 kg, dapat yang diantar tidak dapat memutar fase servo kembali ke titik 0° tabung utama, sedangkan pada beban 1 1/4 kg sesuai dengan data yang diminta servo dapat bekerja dengan untuk menutup fase tabung penampung an utama	Sesuai
6	Melotarkan pakan dengan kipas Motor DC	Mesin dc dapat bekerja untuk melotarkan pakan ikan dapat memutar	Mesin dc dapat bekerja dengan melotarkan pakan ikan melotarkan ke kolam ikan.	sesuai
7	Pengukuran ketersediaan Pakan pada penampung an utama dengan menggunakan sensor ultrasonik	Pengukuran ketersediaan pakan pada tabung penampung an utama dengan sensor ultrasonik dengan rumus total	Setelah dilakukan pengujian beberapa kali terdapat 1-2 cm dari jarak pantulan yang dipancarkan dengan sensor ultrasonik. Artinya pengukuran	Kurang sesuai

		total tinggi penampung an dikurangi dengan jarak pantulan pakan.	bisa kurang sekitar 1-2 cm	
8	Akurasi jam menggunakan Zona (+8) pemberian pakan ikan pada pagi hari, pada pukul 08.00	Jam pada panel yang menggunakan zona (+8) dapat sama dengan library pada Arduino diaktifkan dengan pembukaan servo tabung penampung an pakan ikan yang akan tobar pada pukul 08.00	Servo pada penampung an pakan ikan yang akan diantar bekerja pada pukul 08.00 artinya ada perbedaan 1 menit dari waktu pada panel.	Kurang sesuai
9	Informasi ketersediaan pakan pada dashboard	Aplikasi bisa dapat menerima data dari Node Mcu ESP8266 untuk ditampilkan di dashboard aplikasi	Aplikasi bisa dapat menerima data dari Node MCU ESP8266 melalui jaringan wifi dan ditampilkan dalam aplikasi bisa.	Sesuai
10	Notifikasi ketersediaan pakan melalui Notifikasi pakan sudah diantar	Aplikasi bisa dapat memberikan notifikasi ketersediaan pakan melalui dan notifikasi pakan sudah diantar	Aplikasi bisa mengirimkan notifikasi ketersediaan pakan saat ketersediaan pakan melalui, serta aplikasi bisa mengirimkan notifikasi pakan sudah tobar.	Sesuai
11	Detegan koneksi wifi dapat terkoneksi dengan Node Mcu ESP8266	Node MCU ESP8266 dapat terkoneksi untuk jaringan wifi	Node MCU ESP8266 dapat terkoneksi dengan wifi diaktifkan dengan jaringan Router mobile yang terkoneksi	Sesuai

4.4. Persamaan Matematika

Pemberian pakan otomatis menggunakan Node Mcu Esp8266 dengan Real Time Clock (RTC) DS3231 membuat alat ini berjalan sesuai dengan waktu yang telah diatur sesuai dengan keinginan secara tertur dan dipantau melalui Lcd (Liquid Crystal Display) [4]. Pemberian pakan otomatis menggunakan dengan Node Mcu Esp8266 yang berbasis Internet of Thing (IoT) pada kolam ikan yang berukuran besar sehingga memerlukan pelontar menggunakan Motor AC untuk pelontar sehingga pelontaran pakan bisa jarak jauh, penakaran pakan menggunakan putaran servo untuk mencari berat pakan yang diinginkan dan ketersediaan pakan diukur dengan menggunakan sensor ultrasonik serta data dari proses tersebut dapat ditampilkan dalam

bentuk web[5]. Pemberi pakan ikan otomatis berbasis web Designing and developing a web-based interface of automatic fish feeding sistem dapat dipantau melalui web dengan menyajikan data mulai dari suhu, keadaan air kolam sampai pemberian pakan secara otomatis[6]. Pemberian Pakan Ikan Koki Otomatis Pada Aquarium Berbasis Mikrokontroler AT89S52 hasil dari penelitian ini memberikan pakan ikan otomatis pada aquarium dengan menggunakan keypad untuk mengatur waktunya[7]. Simulasi alat pemberi pakan dan pengendali kincir air yang berdasarkan suhu dan kadar oksigen pada kolam ikan gurami berbasis MCU AT89C51 merupakan penelitian simulasi untuk pemberian pakan gurami secara otomatis dan dapat menampilkan suhu serta kadar oksigen[8]. Perancangan dan implementasi alat pemberian pakan ikan lele otomatis pada fase pendederan berbasis arduino dan aplikasi blynk dengan hasil penelitian ini adalah memudahkan pada fase pendederan dalam pemberian pakan dengan dikontrol smartphone untuk mengetahui sisa pakan yang tersedia[9]. Implementasi sistem pakan ikan menggunakan buzzer dan aplikasi antarmuka berbasis mikrokontroler dengan hasil penelitian. Hasil penelitian ini berupa alat pemberian pakan ikan secara otomatis sesuai dengan penjadwalan yang telah ditentukan sebelumnya ataupun melalui aplikasi antarmuka dan sistem peringatan menggunakan buzzer[10]. sedang automatic feeder pemberian pakan ikan otomatis yang berhasil dibuat adalah pemberian pakan otomatis untuk kolam lele bioflox yang dapat telah diatur jam pemberian pakan secara otomatis, sistem dapat mengukur berat pakan yang telah ditentukan dengan menggunakan load cell dengan jarak lontaran pakan mencapai 1 Meter dengan menggunakan Motor DC, jumlah tersedian pakan yang diukur dengan menggunakan sensor ultrasonic dan dapat di pantau dari aplikasi blynk pada mobile android dan IOS.

```
#include <Servo.h>
#include <NTFClient.h>
#include <WiFiUdp.h>
#include <SimpleTimer.h>
#include "HX711.h"
#define DOUT D1
#define CLK D0

#include <Servo.h>

char auth[] = "99ffbf578ad66df1b3601628bfad6d2";
char ssid[] = "Phone";
char pass[] = "chancid";

//char auth[] = "99ffbf578ad66df1b3601628bfad6d2";
//char ssid[] = "L";
//char pass[] = "kompota@amta.com";

const long utcOffsetInSeconds = 25200;
const int ledPin = 13;
const int ledPinServo = 13;

HX711 scale(DOUT, CLK);
//float calibration_factor = 650;
float calibration_factor = 127.50;
//float calibration_factor = 390.5;
int GRAM;
//int relayOutput = 13; //DS NodeMCU
int statusServo;

Servo servo1;
Servo servo2;

SimpleTimer timer;

WiFiUDP udpUDP;
NTFClient ntfClient(udpUDP, "id.pool.org", utcOffsetInSeconds);

//AT89C52
void setup() {
  Serial.begin(9600);

  WiFi.begin(ssid, pass);

  while(WiFi.status() != WL_CONNECTED){
    Serial.print(".");
    delay(100);
  }

  timer.begin();

  Serial.println("Tekan a,s,d,f untuk menaikkan
  calibration_factor ke 10,100,1000,10000");
  Serial.println("Tekan z,x,c,v untuk menurunkan
  calibration_factor ke 10,100,1000,10000");
  Serial.println("Tekan T untuk Tare");
  scale.set_scale();
  scale.tare();
  long zero_factor = scale.read_average();
  Serial.println("Zero factor: ");
  Serial.println(zero_factor);

  servo1.attach(D0); //D0
  //servo2.attach(D4); //D4
  servo1.write(90); //D0
  // servo2.write(0);
  statusServo = 0; //servo tertutup
}
```

4.5. Kode Pemrograman

```
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
```

```

void loop() {
    int jam, menit, detik;

    timeClient.update();

    jam = timeClient.getHours();
    menit = timeClient.getMinutes();
    detik = timeClient.getSeconds();

    Serial.println("\n");
    Serial.println(jam);
    Serial.println("-");
    Serial.println(menit);
    Serial.println("-");
    Serial.println(detik);
    Serial.println("\n");

    scale.set_scale(calibration_factor);
    GRAM = scale.get_units(), 4;
    Serial.println("Reading: ");
    Serial.println(GRAM);
    Serial.println(" Gram");
    Serial.println(" calibration_factor: ");
    Serial.println(calibration_factor);
    Serial.println(" Servo Status: ");
    Serial.println(statusServo);
    Serial.println();

    if (GRAM == 1000) {
        if (jam + 1 == intUlangPagi || jam + 1 ==
        intUlangSore) {
            if (menit == 10) {
                if (statusServo == 1) {
                    statusServo = 1;
                    Serial.println("SERVO IS OPENING");
                }
                else {
                    servo1.write(0);
                    statusServo = 1;
                    Serial.println("SERVO IS OPENED");
                }
            }
            else {
                if (statusServo == 1) {
                    servo1.write(90);
                    statusServo = 0;
                    Serial.println("SERVO IS CLOSED");
                }
                else {
                    statusServo = 0;
                    Serial.println("SERVO IS CLOSING");
                }
            }
        }
        else {
            if (statusServo == 0) {
                statusServo = 0;
                Serial.println("SERVO IS CLOSING");
            }
            else {
                servo1.write(90);
                statusServo = 0;
                Serial.println("SERVO IS CLOSED");
            }
        }
        else {
            if (statusServo == 0) {
                statusServo = 0;
                Serial.println("SERVO IS CLOSING");
            }
        }
    }
}
    
```

```

else {
    servo1.write(90);
    statusServo = 0;
    Serial.println("SERVO IS CLOSED");
}
}

if (Serial.available()) {
    char temp = Serial.read();
    if (temp == '/' || temp == '\n')
        calibration_factor -= 0.1;
    else if (temp == 'x' || temp == 'X')
        calibration_factor += 0.1;
    else if (temp == '1')
        calibration_factor -= 10;
    else if (temp == 'h')
        calibration_factor += 10;
    else if (temp == '4')
        calibration_factor += 100;
    else if (temp == 'c')
        calibration_factor += 100;
    else if (temp == 'f')
        calibration_factor += 1000;
    else if (temp == 'y')
        calibration_factor += 1000;
    else if (temp == 'v')
        scale.tare();
}

delay(1000);
}
    
```

Tabel 1. Kodingan Skematik pertama

```

#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <Servo.h>
#include <NTPClient.h>
#include <WiFiUdp.h>
#include <SimpleTimer.h>

#define TRIGGERPIN D1
#define ECHO_PIN D2
#define relay D4

char auth[] =
    "00f5fb878e2446f18b5601638bfd4d3";
char ssid[] = "iPhone";
char pass[] = "chancid";

//char auth[] =
//    "00f5fb878e2446f18b5601638bfd4d3";
//char ssid[] = "L";
//char pass[] = "hotspotjaminjem";

const long utcOffsetInSeconds = -25200;
#define one_tint_makanPagi - 11;
#define one_tint_makanSore - 21;

int servoStatus;

//Servo servo1;
Servo servo2;

SimpleTimer timer;
    
```



```

WiFiUDP udpUDP,
NTPClient      timeClient(udpUDP,
"ntp.pool.org", utcOffsetInSeconds);

WidgetLCD led(V1);
WidgetLCD led2(V2);

void setup() {
  Serial.begin(9600);
  Blynk.begin(auth, ssid, pass);
  timeClient.begin();

  pinMode(TRIGGERPIN, OUTPUT);
  pinMode(ECHOPIN, INPUT);
  pinMode(relay, OUTPUT);

  digitalWrite(relay, HIGH);

  led.clear(); //Use it to clear the LCD Widget
  led2.clear(); //Use it to clear the LCD
  Widget
  led.print(0, 0, "Jarak dalam cm"); // use:
  (position X: 0-15, position Y: 0-1, "Message
  you want to print")
  led2.print(0, 0, "Pernaburan Pakan"); // use:
  (position X: 0-15, position Y: 0-1, "Message
  you want to print")

  //servo1.attach(D4); //D4
  servo2.attach(D3); //D3
  //servo1.write(0);
  servo2.write(70);
  servoStatus = 0;
}

void loop() {

  long duration;
  float distance;
  int jam, menit, detik;

  timeClient.update();

  Serial.println(timeClient.getFormattedTime());
  ;

  timer.run();
  led.clear();
  led2.clear();

  digitalWrite(TRIGGERPIN, LOW);
  delayMicroseconds(2);

  digitalWrite(TRIGGERPIN, HIGH);
  delayMicroseconds(2);

  jam = timeClient.getHours();
  menit = timeClient.getMinutes();
  detik = timeClient.getSeconds();

  digitalWrite(TRIGGERPIN, LOW);
  duration = pulseIn(ECHOPIN, HIGH);
  distance = (duration/2) / 29.1;
    
```

```

if(distance >= 30) {
  Serial.print(distance);
  Serial.println("cm");
  led.print(0, 0, "Pakan Menipis");
  led2.print(0, 1, "Segera Isi Ulang!");
}
else {
  Serial.print(distance);
  Serial.println("cm");
  led.print(0, 0, "Pakan Terpenuhi");
  led2.print(7, 1, distance);
}

if(jam + 1 == makanPagi || jam + 1 ==
makanSore) {
  if(menit < 1) {
    if(servoStatus == 1) {
      servoStatus = 1;
      led2.print(0, 0, "Saat ini pakan");
      led2.print(0, 1, "sedang ditabur");
    }
    else {
      Serial.print(jam);
      Serial.println(":");
      Serial.print(menit);
      Serial.println(":");
      Serial.print(detik);
      Serial.println("s");
      //servo1.write(0);
      servo2.write(0);
      delay(100);
      digitalWrite(relay, LOW);
      servoStatus = 1;
      led2.print(0, 0, "Saat ini pakan");
      led2.print(0, 1, "sedang ditabur");
    }
  }
  else {
    if(servoStatus == 1) {
      digitalWrite(relay, HIGH);
      Serial.print(jam);
      Serial.println(":");
      Serial.print(menit);
      Serial.println(":");
      Serial.print(detik);
      Serial.println("s");
      //servo1.write(90);
      servo2.write(75);
      delay(100);
      servoStatus = 0;
      if(jam + 1 >= makanPagi && jam + 1 <
makanSore) {
        led2.print(0, 0, "Telah ditabur");
        led2.print(0, 1, "Jam 11 siang");
      }
      else {
        led2.print(0, 0, "Telah ditabur");
        led2.print(0, 1, "Jam 9 malam");
      }
    }
    else {
      servoStatus = 0;
      if(jam + 1 >= makanPagi && jam + 1 <
makanSore) {
    
```


- Inf, no. Semester Genap2013/2014, 2014.
- [11]. M. Muksin, "Simulasi alat pemberi pakan dan pengendali kincir air yang berdasarkan suhu dan kadar oksigen pada kolam ikan gurami berbasis mcu at89c51," *Widya Tek.*, vol. 18, no. 1, pp. 40–43, 2010.
- [12]. Harifuzzumar, F. Arkan, and Ghiri Basuki Putra, "Perancangan Dan Impementasi Alat Pembe Rian Pakan Ikan Lele Otom Atis Pada Fase Pendederan Berbasis Arduino Dan Aplikasi Blynk," *Pros. Semin. Nas. Penelit. Pengabdi. pada Masy.*, pp. 67–71, 2018.
- [13]. Y. A. Sari, Kartika, Cucu Suhery, "Implementasi Sistem Pakan Ikan Menggunakan Buzzer Dan Aplikasi Antarmuka Berbasis Mikrokontroler," *J. Coding Sist. Komput. Untan*, vol. 3, no. 2, p. 113, 2015.