

DAFTAR PUSTAKA

Hidayat, R., & Wibowo, A. R. (2014). ANALISIS PENGENDALIAN NYALA LAMPU RUMAH DENGAN MENGGUNAKAN ANDROID SMARTPHONE BERBASIS MIKROKONTROLER ATMEGA 8535. *Jurnal Online Sekolah Tinggi Teknologi Mandala*, 8(1), 35-42.

Hilal, A., & Manan, S. (2015). Pemanfaatan Motor Servo Sebagai Penggerak Cctv Untuk Melihat Alat-Alat Monitor Dan Kondisi Pasien Di Ruang Icu. *Gema Teknologi*, 17(2).

Hildayanti, A., & Machrizzandi, M. S. R. (2020). THE APPLICATION OF IOT (INTERNET OF THINGS) FOR SMART HOUSING ENVIRONMENTS AND INTEGRATED ECOSYSTEMS. *Nature: National Academic Journal of Architecture*, 7(1), 80-88.

Himawan, F., Perdana, P., & Surya, Y. A. (2021). Rancang Bangun Purwarupa Smart Garden Menggunakan Kamera, Sensor Suhu Dan Kelembaban Tanah Berbasis Internet Of Things (IOT) Dengan ESP8266. *Jurnal JEETech*, 2(2), 78-83.

Junaidi, A. (2015). Internet of things, sejarah, teknologi dan penerapannya. *Jurnal Ilmiah Teknologi Infomasi Terapan*, 1(3).

Lutfiyanto, A. H. (2018). *Rancang Bangun Pintu Wahana Otomatis Menggunakan Sensor Ultrasonik HC-SR04 sebagai Pengukur Tinggi Badan dan Sensor Load Cell Dilengkapi dengan Hx711 Sebagai Pengukur Berat Badan berbasis Arduino Mega 2560* (Doctoral dissertation, undip).

Metcalf & Eddy, (2003), “ Wastewater Engineering – Treatment and Reuse “, Fourth Edition, (Mc Graw Hill, New York)

R. Setiawan, “Rancang Bangun Alat Pemberi Pakan Ikan Menggunakan Mikrokontroler,” J. ICTEE, vol. 1, no. 1, pp. 51–54, 2020, doi: 10.33365/jictee.v1i1.698.



S., & Burlian, A. (2019). Sistem penjadwalan pompa air otomatis pada k menggunakan mikrokontrol Arduino UNO R3. In *Seminar Nasional i* (Vol. 1, No. 2, pp. 187-187).

Setiawan, Yoyok. 2017. Rancang Bangun Pemantauan Dan Penjadwalan Alat Pemberi Pakan Ikan Otomatis Secara Jarak Jauh. Skripsi. Institut Bisnis dan Informatika Stikom: Surabaya.

Sholeh, N., Joni, K., & Ulum, M. (2020). Sistem Monitoring Kondisi Kendaraan Motor Injeksi Berbasis Mikrokontroler. *Jurnal JEETech*, 1(1), 37-42.

Wang, C., Daneshmand, M., Dohler, M., Mao, X., Hu, R. Q., & Wang, H. (2013). Guest Editorial-Special issue on internet of things(IoT): Architecture, protocols and services. *IEEE Sensors Journal*, 13(10), 3505–3508.

Widharma, I. G. S., & Wiranata, L. F. *Mikrokontroler dan Aplikasi*. wawasan Ilmu.

Widodo B, Firmansyah R. 2008. Elektronika digital + mikroprosesor. Yogyakarta: Andi.

Wijanarko, D., Widiastuti, I., & Widya, A. (2017). Gelombang Ultrasonik Sebagai Alat Pengusir Tikus Menggunakan Mikrokontroler Atmega 8. *Jurnal Teknologi Informasi dan Terapan*, 4(1), 65-70.



LAMPIRAN

```
#include <virtuabotixRTC.h> //|

#include "CTBot.h"

#include <SoftwareSerial.h>

// software serial #1: RX = digital pin D5, TX = digital pin D6

SoftwareSerial portOne(14, 12); // Arduino to ESP (5nano --> D6esp & 4nano -->
D5esp)

char baca ;

bool parsing2 = false;

bool parsing3 = false;

bool kirimberat = false;

String rData, data[3];

String berat1 = "";

String berat2 = "";

String berat3 = "";

CTBot myBot;

String ssid = "Indihome";

String pass = "23456789";

String token = "5627244583:AAF1T1UCjV3TuMxSAIJwYyN-8fCrcGjW7U0";

id = 1388511839; //937692071 //1388511839
```



```

// Creation of the Real Time Clock Object

virtuabotixRTC myRTC(5, 4, 2); //sck(D1),data(D2),rst(D4)

void RTC1();

bool RTC = false;

void Kirim1();

bool kirim = false;

//+++++
+++++//

void setup() {

  Serial.begin(9600);

  portOne.begin(9600);

  Serial.println("Starting TelegramBot...");

  myBot.wifiConnect(ssid, pass);

  myBot.setTelegramToken(token);

  if (myBot.testConnection()) {

    Serial.println("Koneksi Bagus");

    // myBot.sendMessage( id, "SELAMAT DATANG");

    Serial.println("Terakhir");

    Serial.println("Koneksi Jelek");
  }
}

```



```
}
```

```
// Set the current date, and time in the following format:
```

```
// seconds, minutes, hours, day of the week, day of the month, month, year
```

```
//myRTC.setDS1302Time(25, 56, 13, 6, 5, 5, 2023);
```

```
RTC = true;
```

```
myBot.sendMessage( id, "SELAMAT DATANG");
```

```
Serial.println("Terkirim");
```

```
//portOne.print("2");
```

```
}
```

```
//Terima Pesan dari Telegram
```

```
if (CTBotMessageText == myBot.getNewMessage(msg)) {
```

```
    kirimberat = false;
```

```
    String message = msg.text;
```

```
    Serial.println(message);
```

```
    int index = 0;                                //Urutan
```

```
    arrData[0] = "";                               //Reset Variabel
```

```
    arrData[1] = "";
```

```
    arrData[2] = "";
```

```
    arrData[3] = "";
```



```
        print("ini pesan ");
```

```
        println( message);
```

```

for (int i = 0; i < message.length(); i++) {

    char delimiter = '#';                //Karakter(')

    if (message[i] != delimiter) {

        if ( i = 2

            arrData[index] += message[i];        //Gabungkan isi data array dengan
            karakter yg sedang dibaca

            Serial.print("ini index");

            Serial.println(index);

            Serial.print("aardata");

            Serial.println( arrData[index]);

        }

    else {

        index = index + 1;

        delay(10);

    }

}

if (index == 3 ) {                        //Parameter lengkap

    jam1    = arrData[0].toInt();          //Ambil Jam1, Menit, Jam2, Menit2

    menit1   = arrData[1].toInt();

    jam2     = arrData[2].toInt();

    menit2   = arrData[3].toInt();

```



```
    I2C.write(0, jam1);
```

```
    //Simpan data ke EEPROM
```

```
    I2C.write(1, menit1);
```

```

EEPROM.write(2, jam2);

EEPROM.write(3, menit2);

EEPROM.commit();


Serial.println("Atur Jam Makan....");          //Setting Jam Makan

Serial.println(String() + "jam makan 1 = " + jam1 + "menit makan1 = " +
menit1);

myBot.sendMessage(id, "Pembaharuan Berhasil");

}

//delay(500);

//Serial.println("Tersimpan");

//myBot.sendMessage( id, "Berhasil hahahha");

//Serial.println(id);

//portOne.print("3");

//myBot.sendMessage(id, "Silakan masukkan jam & menit \nFormat :
Jam1#Menit1#Jam2#Menit2");


Serial.print("jam = ");

Serial.println(jam1);

if (message == "/Y") {

    RTC = false;

    //delay(1000);

    .println("Servo Nyala");

    t.sendMessage( id, "Berhasil Memberi Makan & Minum");

```



```

    portOne.print("1");

    RTC = true;

}

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+++++//

void loop() {

    TBMessage msg;

    int jam  = myRTC.hours;

    int menit = myRTC.minutes;

    int detik = myRTC.seconds;

    //RTC1();

    if (RTC) {

        RTC1();

    }

    //Atur Jadwal Waktu Pakan dan Minum

    if (jam == 14 && menit == 05 && detik == 12 || jam == 19 && menit == 34 &&
    detik == 15) {

        delay(500);

        myBot.sendMessage( id, "Waktunya Memberi Makan dan Minum\n/Y Iya \n/N

```



```
println("Terkirim1");
```



```

}

//Kirim Info Berat ke Telegram

if (kirimberat) {

    berat3 = "";

    berat3 += "Sisa Pakan = ";

    berat3 += berat1;

    berat3 += " gr dan Sisa Minuman = ";

    berat3 += berat2;

    berat3 += " mililiter";

    myBot.sendMessage( id, berat3);

    Serial.print(berat3);

    kirimberat = false;

}

//Terima Pesan dari Telegram

if (CTBotMessageText == myBot.getNewMessage(msg)) {

    // myBot.sendMessage(msg.sender.id, msg.text);

    String message = msg.text;

    Serial.println(message);

```



```

sage == "/Y") {

    = false;

```

```
//delay(1000);

Serial.println("Servo Nyala");

myBot.sendMessage( id, "Berhasil Memberi Makan & Minum");

portOne.print("1");

RTC = true;

}
```

```
if (message == "/command1") {

    RTC = false;

    delay(1000);

    Serial.println("Cek Pakan");

    portOne.print("2");

    RTC = true;

}
```

```
if (message == "/command2") {

    RTC = false;

    //delay(500);

    Serial.println("Atur Bukaannya");
```

```
    myBot.sendMessage( id, "Silahkan Atur Jumlah Pakan dan Minum yang
dikeluarkan\n /1 Pakan 40 gr dan Minum 50 ml\n /2 Pakan 80 gr dan Minum 100
ml");
```

```
    = true;
```



```
if (message == "/1") {  
  
    RTC = false;  
  
    //delay(500);  
  
    Serial.println("Tersimpan");  
  
    myBot.sendMessage( id, "Data Tersimpan");  
  
    portOne.print("3");  
  
    RTC = true;  
  
}
```

```
if (message == "/2") {  
  
    RTC = false;  
  
    //delay(500);  
  
    Serial.println("Tersimpan");  
  
    myBot.sendMessage( id, "Data Tersimpan");  
  
    portOne.print("4");  
  
    RTC = true;  
  
}
```

```
}
```



```
at();
```

```

}

//=====
=====//

```

```

void RTC1() {

    myRTC.updateTime();                //|

    // Start printing elements as individuals                //|

    //Serial.print("Current Date / Time: ");                //|

    Serial.print(myRTC.dayofmonth);                //|

    Serial.print("/");                //|

    Serial.print(myRTC.month);                //|

    Serial.print("/");                //|

    Serial.print(myRTC.year);                //|

    Serial.print(" ");                //|

    Serial.print(myRTC.hours);                //|

    Serial.print(":");                //|

    Serial.print(myRTC.minutes);                //|

    Serial.print(":");                //|

    Serial.println(myRTC.seconds);

```



so the program doesn't print non-stop

```
100);
```

```

void bacaberat() {

    while (portOne.available()) {

        baca = portOne.read();

        //Serial.println(baca);

        delay(15);

        rData += baca;

        //Serial.print(rData);

        //-----Baca Data Timbangan-----//

        if (baca == '&') {

            parsing2 = true;

        }

        if (parsing2) {

            int a = 0;

            for (int b = 0; b < rData.length(); b++) {

                if (rData[b] == '/') {

                    a++;

                    data[a] = "";

                }

                {

                    a[a] += rData[b];

```



```

    }

}

if (a == 3) {

    berat1 = data[1];

    berat2 = data[2];

    //Serial.println (berat1);

    kirimberat = true;

}

//Serial.println (data[a]);

parsing2 = false;

rData = "";

}

//-----Baca Data Timbangan1-----//

if (baca == 'x'){

    Serial.println("Pakan & Minum Habis");

    myBot.sendMessage( id, "Alert !!!\nPakan & Minum Habis !!!");

}

}

```



//ARDUINO NANO

#include <EEPROM.h>

#include "HX711.h"

#include <Servo.h>

#include <Wire.h> //library I2C

#include <LiquidCrystal_I2C.h> //library LCD

LiquidCrystal_I2C lcd(0x27, 16, 2); // set address I2C dan besar karakter untuk lcd
16×2

#include <SoftwareSerial.h>

// software serial #1: RX = digital pin 5, TX = digital pin 4

SoftwareSerial portOne(5, 4); // Arduino to ESP (5nano --> D6esp & 4nano -->
D5esp)

char baca ;

int Tpakan, Tpakan1, Tminum, Tminum1;

bool lcd2 = false;

//Timbangan

HX711 scale1;

HX711 scale2;

float calibration_factor1 = 60.60;



ration_factor2 = 90.40;

t1;

```

int weight2;

String DataTimbangan = "";

Servo myservo; // create servo object to control a servo

// twelve servo objects can be created on most boards

#define IN3 9 // deklarasi pin IN3

#define IN4 8 // deklarasi pin IN4

#define ENB 3 // deklarasi pin ENB

int pos = 0; // variable to store the servo position

bool servo = false;

bool motor = false;

void servo1();

void motor1();

void setup() {

  Serial.begin(9600);

  portOne.begin(9600);

  //EEPROM

  EEPROM.begin();

  = EEPROM.read(0);           //Baca EEPROM

  1 = EEPROM.read(1);

```




```

delay(100);

//LCD

lcd.begin();

lcd.backlight();

lcd.print("Robot Pakan");

delay(3000);

lcd.clear();

lcd2 = true;

//Timbangan

scale1.begin(12, 13); //Pin untuk Load cell 1

scale2.begin(6, 7); //Pin untuk Load cell 2

scale1.set_scale(calibration_factor1);

scale1.tare();

scale2.set_scale(calibration_factor1);

scale2.tare();

//Servo

myservo.attach(11); // pin servo 11 di nano

```



```

    .write(135);

```

```

    000);

```

```
servo = true;
```

```
//Motor
```

```
pinMode(IN3, OUTPUT);
```

```
pinMode(IN4, OUTPUT);
```

```
pinMode(ENB, OUTPUT);
```

```
}
```

```
void loop() {
```

```
  //scale1.set_scale(calibration_factor1);
```

```
  weight1 = scale1.get_units(), 4;
```

```
  //scale2.set_scale(calibration_factor2);
```

```
  weight2 = scale2.get_units(), 4;
```

```
  DataTimbangan = "";
```

```
  DataTimbangan += "/";
```

```
  DataTimbangan += weight1;
```

```
  DataTimbangan += "/";
```

```
  1bangan += weight2;
```

```
  1bangan += "/";
```



```

DataTimbangan += "&";

Serial.println(DataTimbangan);

delay(1000);

if (lcd2) {
    lcd1();
}

//Atur indikator pakan dan minum yang habis
if (weight1 <= 5 && weight2 <= 5) {
    portOne.print("x");
    delay(2000);
}

while (portOne.available()) {
    baca = portOne.read();
    Serial.println(baca);
    delay(500);
    if (baca == '1') {
        Serial.println("Servo dan Motor");
        1);
        (1000);

```



```

    motor1();

}

if (baca == '2') {

    Serial.println(DataTimbangan);

    portOne.print(DataTimbangan);

}


if (baca == '3') {

    Tpakan = 165;

    Tminum = 100;

    EEPROM.write(0, Tpakan);                //Simpan data ke EEPROM

    EEPROM.write(1, Tminum);

    Serial.println(Tpakan);

    Serial.println(Tminum);

}


if (baca == '4') {

    Tpakan = 180;

    Tminum = 200;

    EEPROM.write(0, Tpakan);                //Simpan data ke EEPROM

    EEPROM.write(1, Tminum);

```



```

        .println(Tpakan);

        .println(Tminum);

```

```

    }

}

}

void servo1() {

    lcd2 = false;

    lcd.clear();

    myservo.write(Tpakan);          // tell servo to go to position in variable 'pos'

    lcd.print("Memberi Pakan");

    Serial.print("Sudut Servo = ");

    Serial.println(Tpakan);

    delay(20000);                  // waits 15 ms for the servo to reach the position

    myservo.write(135);            // tell servo to go to position in variable 'pos'

    lcd2 = true;

}

```

```

void motor1() {

    lcd2 = false;

    lcd.clear();

    lcd.print("Memberi Minum");

    digitalWrite(IN3, HIGH);

```



```

    digitalWrite(IN4, LOW);

```

```

    digitalWrite(ENB, Tminum);

```

```
Serial.print("Kecepatan motor = ");  
  
Serial.println(Tminum);  
  
delay(10000); // Jeda 10 detik  
  
digitalWrite(IN3, LOW);  
  
digitalWrite(IN4, LOW);  
  
analogWrite(ENB, 0);  
  
lcd2 = true;  
  
}
```

```
void lcd1() {  
  
  lcd.clear();  
  
  lcd.setCursor(0, 0);  
  
  lcd.print("Pakan = " );  
  
  lcd.print(weight1);  
  
  lcd.setCursor(12, 0);  
  
  lcd.print("gr");  
  
  lcd.setCursor(0, 1);  
  
  lcd.print("Minum = ");  
  
  lcd.print(weight2);  
  
  lcd.setCursor(12, 1);  
  
  lcd.print("ml");  
  
}
```



//KALIBRASI LOAD CELL

```
#include "HX711.h"
```

```
#define DOUT 6
```

```
#define CLK 7
```

```
HX711 scale(DOUT, CLK);
```

```
float calibration_factor = 90.40;
```

```
int GRAM;
```

```
void setup() {
```

```
    Serial.begin(9600);
```

```
    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.print("Zero factor: ");
```

```
    Serial.println(zero_factor);
```

```
    }00);
```



```

void loop() {

    scale.set_scale(calibration_factor);

    GRAM = scale.get_units(), 4;

    Serial.print("Reading: ");

    Serial.print(GRAM);

    Serial.print(" Gram");

    Serial.print(" calibration_factor: ");

    Serial.print(calibration_factor);

    Serial.println();

    if (Serial.available()) {

        char temp = Serial.read();

        if (temp == '+' || temp == 'a')

            calibration_factor += 0.1;

        else if (temp == '-' || temp == 'z')

            calibration_factor -= 0.1;

        else if (temp == 's')

            calibration_factor += 10;

        else if (temp == 'x')

            calibration_factor -= 10;

        (temp == 'd')

        ation_factor += 100;

```




```

else if (temp == 'c')

    calibration_factor -= 100;

else if (temp == 'f')

    calibration_factor += 1000;

else if (temp == 'v')

    calibration_factor -= 1000;

else if (temp == 't')

    scale.tare();

}

}

//MOTOR DRIVER WATERPUMP

#define IN3 9 // deklarasi pin IN3

#define IN4 8 // deklarasi pin IN4

#define ENB 3 // deklarasi pin ENB

void setup() {

    // put your setup code here, to run once:

    pinMode(IN3, OUTPUT);

    pinMode(IN4, OUTPUT);

    pinMode(ENB, OUTPUT);

}

```



```
() {
```

```
// put your main code here, to run repeatedly:
```

```
digitalWrite(IN3, HIGH);
```

```
digitalWrite(IN4, LOW);
```

```
analogWrite(ENB, 100);
```

```
delay(5000); // Jeda 10 detik
```

```
digitalWrite(IN3, LOW);
```

```
digitalWrite(IN4, LOW);
```

```
analogWrite(ENB, 0);
```

```
delay(10000);
```

```
}
```

```
//RTC
```

```
#include <virtuabotixRTC.h>
```

```
//|
```

```
#include "CTBot.h"
```

```
#include <Servo.h>
```

```
#include "HX711.h"
```

```
Servo myservo; // variable untuk menyimpan posisi data
```



```
yBot;
```

```

String ssid = "Reader";

String pass = "12345678";

String token = "5627244583:AAF1T1UCjV3TuMxSAIJwYyN-8fCrcGjW7U0";

const int id = 937692071; //937692071 //1388511839


// Creation of the Real Time Clock Object

virtuabotixRTC myRTC(5, 4, 2); //sck(D1),data(D2),rst(D4)

// uint8_t hari = myRTC.dayofmonth;

// uint8_t bulan = myRTC.month;

// uint8_t tahun = myRTC.year;


HX711 scale1; //load cell


#define IN3 12 // deklarasi pin D6

#define IN4 13 // deklarasi pin D7

#define ENB 14 // deklarasi pin D7


void setup() {

  Serial.begin(9600);

  // pinMode(IN3, OUTPUT);

  // pinMode(IN4, OUTPUT);

  de(ENB, OUTPUT);

```



```

Serial.println("Starting TelegramBot...");

myBot.wifiConnect(ssid, pass);

myBot.setTelegramToken(token);

if (myBot.testConnection()) {

    Serial.println("Koneksi Bagus");

    myBot.sendMessage( id, "SELAMAT DATANG");

} else {

    Serial.println("Koneksi Jelek");

}

// myservo.attach(15); //D8

//scale1.begin(0, 15); // timbangan

// Set the current date, and time in the following format:

// seconds, minutes, hours, day of the week, day of the month, month, year

//myRTC.setDS1302Time(00, 21, 17, 6, 28, 1, 2023);

}

```



```

+++++
+++++//

```

```

() {

```

```

myRTC.updateTime();                                //|

// Start printing elements as individuals           //|

//Serial.print("Current Date / Time: ");           //|

Serial.print(myRTC.dayofmonth);                     //|

Serial.print("/");                                  //|

Serial.print(myRTC.month);                          //|

Serial.print("/");                                  //|

Serial.print(myRTC.year);                          //|

Serial.print(" ");                                  //|

Serial.print(myRTC.hours);                         //|

Serial.print(":");                                  //|

Serial.print(myRTC.minutes);                       //|

Serial.print(":");                                  //|

Serial.println(myRTC.seconds);

// Delay so the program doesn't print non-stop

delay(1000);

TBMessage msg;

uint8_t jam  = myRTC.hours;

uint8_t menit = myRTC.minutes;

detik = myRTC.seconds;

```



```

// long weight1 = scale1.read();

// Serial.print("Weight 1: ");

// Serial.println(weight1);


// if (myBot.getNewMessage(msg)) {

//   //myBot.sendMessage(msg.sender.id, msg.text);

//   Serial.println(msg.text);

//   String message = msg.text;

//

//   if (message == "/Y") {

//     Serial.println("Servo Nyala");

//     myBot.sendMessage( id, "Makan");

//     myservo.write(180);

//     delay(5000);

//     myservo.write(90);

//     digitalWrite(IN3, HIGH);

//     digitalWrite(IN4, LOW);

//     //digitalWrite(ENB, 50);

ay(2000);

igitalWrite(IN3, LOW);

```



```

//    digitalWrite(IN4, LOW);

//digitalWrite(ENB, 0);

//}

//}

//|

if (jam == 0 && menit == 12) {

    Serial.println("Waktunya Makan");

    myBot.sendMessage(id, "Waktunya Makan dan Minum \n/Y Berikan Makan &
Minum \n/N Tidak Makan dan Minum");

    //delay(1000);

}

}                                     //|

//|

//=====

=====//|

//SERVO

#include <Servo.h>

Servo myservo;

p() {

```



```
// put your setup code here, to run once:
```

```
myservo.attach(11);
```

```
myservo.write(135);
```

```
delay(1000);
```

```
}
```

```
void loop() {
```

```
// put your main code here, to run repeatedly:
```

```
myservo.write(180);
```

```
delay(1000);
```

```
myservo.write(135);
```

```
delay(1000);
```

```
}
```



**LISTING HARGA BIAYA PEMBUATAN ALAT PEMBERI
PAKAN DAN MINUM KUCING OTOMATIS BERBASIS
NODE MCU ESP8266**

No.	Komponen	Jumlah	Harga Satuan	Harga total
1.	Motor Servo	1	Rp.14.000	Rp.14.000
2.	RTC (Real Time Clock) DS1307	1	Rp.8.000	Rp.8.000
3.	LCD 16x2 i2c	1	Rp.27.000	Rp.27.000
4.	Node MCU ESP8266	1	Rp.38.000	Rp.38.000
5.	PCB	1	Rp.7000	Rp.7000
6.	Load-cell sensor	2	Rp.30.000	Rp.60.000
7.	Arduino Nano	1	Rp.46.000	Rp.46.000
8.	Mini Diapgram Water Pump 12V	1	Rp.40.000	Rp.40.000
9	LM2596	1	Rp.7000	Rp.7000
10.	Kabel	Secukupnya	Rp.8000	Rp.8000
11.	Adaptor 12V	1	Rp.48.000	Rp.48.000
12.	Tripleks	1	Rp. 50.000	Rp.50.000
13.	Selang Karet		Rp.10.000	Rp.10.000
14.	Wadah mika	2	Rp.5000	Rp.10.000
TOTAL				Rp.373.000

