

BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1 HASIL IMPLEMENTASI

Setelah sistem dianalisis dan rancasngan secara rinci, maka akan menuju tahap implementasi alat. Implementasi alat merupakan tahap meletakkan sistem sehingga siap untuk dioperasikan. Implementasi bertujuan untuk mengkonfirmasi modul-modul perancangan, sehingga pengguna dapat melihat hasil dari alat yang dibuat.

Seperti yang telah dijelaskan dalam perancangan implementasi ini, alat yang dibuat adalah sebuah box. Box ini dibuat menggunakan bahan plastic jenis akrilik. Pada bagian dalam ruangan box digunakan untuk menanam tanaman seperti terlihat pada gambar 5.1 :



Gambar 5.1 Alat Perawat Tanaman Otomatis

Sensor kelembaban tanah ditancapkan pada tanah didekat tanaman, penampungan air penyiraman dan penampungan pupuk diletakkan dibelakang box dengan pompa air mini didalam setiap penampungan tersebut. Sedangkan sensor suhu diletakkan pada bagian dalam atas box bersama dengan kipas dan led uv.

5.2 PENGUJIAN SISTEM

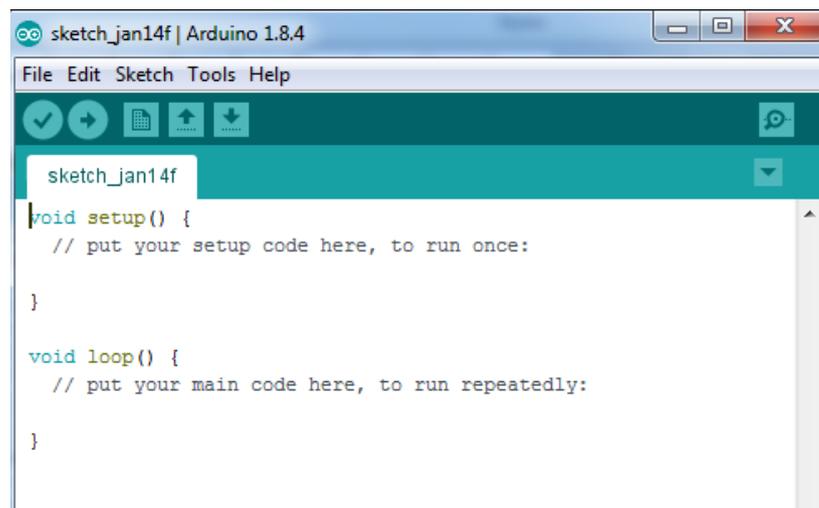
Pengujian sistem bertujuan untuk memastikan apakah semua fungsi sistem bekerja dengan baik dan mencari kesalahan yang mungkin terjadi. Dalam pengujian sistem meliputi pengujian perangkat lunak dan pengujian perangkat keras.

5.2.1 Pengujian Perangkat Lunak

5.2.1.1. Arduino IDE

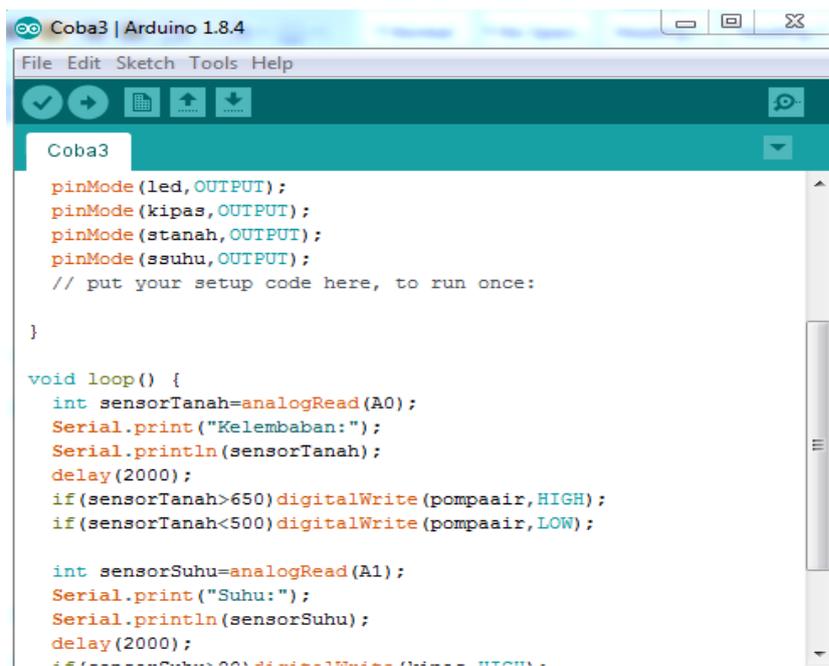
Dalam pembuatan alat ini penulis menggunakan Arduino IDE, sebab Arduino IDE adalah pemrograman yang dibuat untuk mikrokontroller arduino.

Untuk pengujian yaitu pembuatan sketch program baru, tekan file kemudian pilih New.



Gambar 5.2 Menu Sketch Baru

Setelah muncul jendela koding program yang berupa koding *default* seperti gambar diatas maka langkah selanjutnya mulai pengetikkan *sketch* program.



```

Coba3 | Arduino 1.8.4
File Edit Sketch Tools Help
Coba3
pinMode(led, OUTPUT);
pinMode(kipas, OUTPUT);
pinMode(stanah, OUTPUT);
pinMode(ssuhu, OUTPUT);
// put your setup code here, to run once:
}

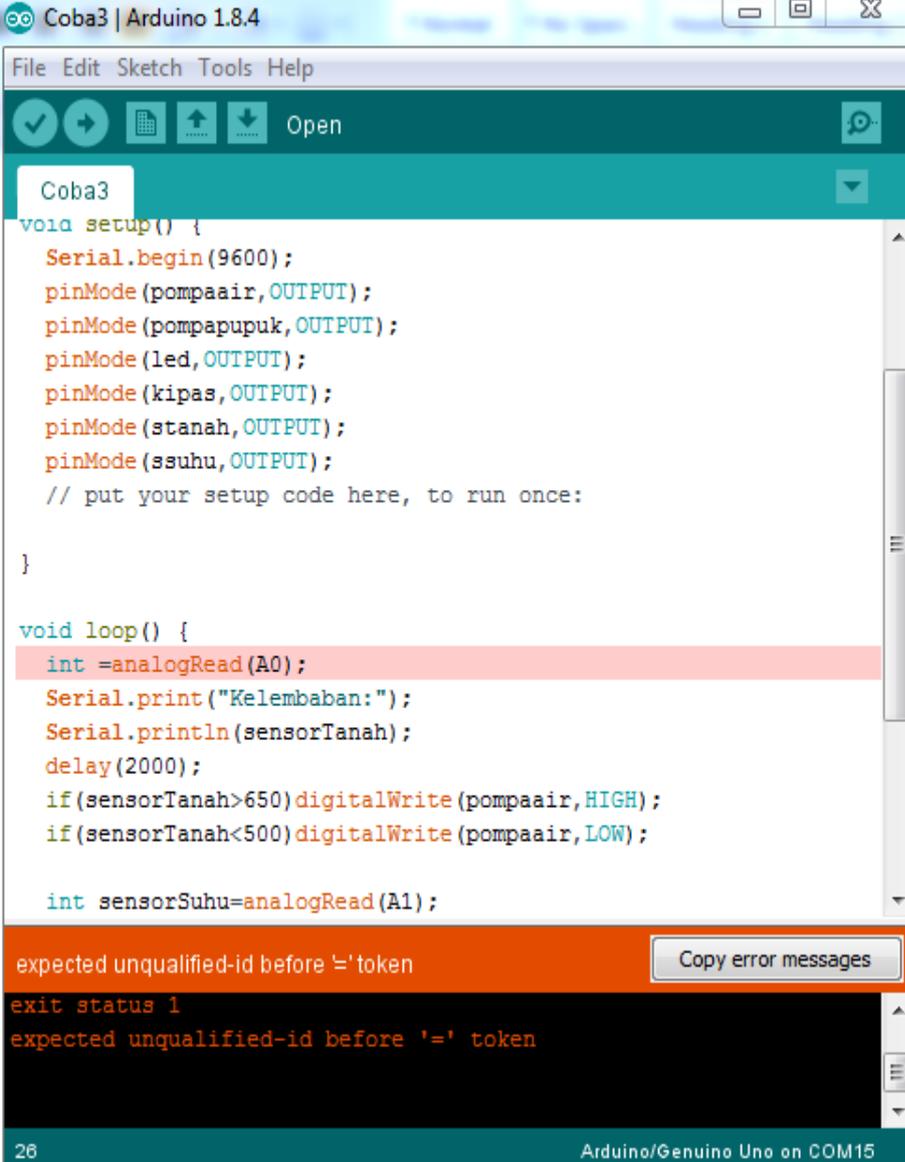
void loop() {
  int sensorTanah=analogRead(A0);
  Serial.print("Kelembaban:");
  Serial.println(sensorTanah);
  delay(2000);
  if(sensorTanah>650)digitalWrite(pompaair,HIGH);
  if(sensorTanah<500)digitalWrite(pompaair,LOW);

  int sensorSuhu=analogRead(A1);
  Serial.print("Suhu:");
  Serial.println(sensorSuhu);
  delay(2000);
  if(sensorSuhu>30)digitalWrite(kipas,HIGH);
}

```

Gambar 5.3 Sketch Program

Setelah *Sketch* program dibuat maka tahap berikutnya adalah menyimpan *sketch* yang telah dibuat dengan memilih menu File kemudian *Save* atau dengan menekan CTRL+S, setelah disimpan langkah selanjutnya *verify/compile* program yang dibuat atau menguji kebenaran koding - koding program yang kita buat dengan cara menekan CTRL + R dan apabila program yang kita buat salah maka akan muncul jendela informasi bahwa terdapat *error*, terdapat petunjuk dimana terjadi kesalahan tersebut, dan apabila program yang kita buat benar maka akan muncul informasi *No error*, maka arduino ide langsung meng*compile* program tersebut seperti tampak pada Gambar 5.4 dan 5.5 berikut :



The screenshot shows the Arduino IDE interface with a file named 'Coba3'. The code in the main editor is as follows:

```

void setup() {
  Serial.begin(9600);
  pinMode(pompaair, OUTPUT);
  pinMode(pompapupuk, OUTPUT);
  pinMode(led, OUTPUT);
  pinMode(kipas, OUTPUT);
  pinMode(stanah, OUTPUT);
  pinMode(ssuhu, OUTPUT);
  // put your setup code here, to run once:
}

void loop() {
  int =analogRead(A0);
  Serial.print("Kelembaban:");
  Serial.println(sensorTanah);
  delay(2000);
  if (sensorTanah>650) digitalWrite (pompaair, HIGH);
  if (sensorTanah<500) digitalWrite (pompaair, LOW);

  int sensorSuhu=analogRead (A1);
}

```

The error message displayed at the bottom of the IDE is:

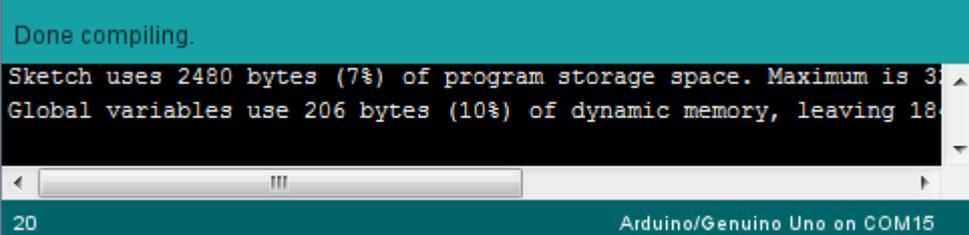
```

expected unqualified-id before '=' token
exit status 1
expected unqualified-id before '=' token

```

The error is highlighted in orange. A 'Copy error messages' button is visible to the right of the error text. The status bar at the bottom indicates '26 Arduino/Genuino Uno on COM15'.

Gambar 5.4 Tampilan Program Ketika Terjadi Error



The screenshot shows the Arduino IDE interface with a successful compilation. The output in the serial monitor is as follows:

```

Done compiling.
Sketch uses 2480 bytes (7%) of program storage space. Maximum is 32256 bytes.
Global variables use 206 bytes (10%) of dynamic memory, leaving 1814 bytes free.

```

The status bar at the bottom indicates '20 Arduino/Genuino Uno on COM15'.

Gambar 5.5 Tampilan Program Tidak Terdapat Error

5.3 PENGUJIAN ALAT

5.3.1 Pengujian Perangkat Keras

Pengujian perangkat keras ini dilakukan untuk mengetahui benar atau tidaknya sebuah rangkain listrik yang telah di rangkai. Pengujian dilakukan secara satu-persatu dari beberapa rangkaian yang telah selesai dibuat dan dengan alat bantu multimeter. Adapun tahapan yang dilakukan dalam pengujian perangkat keras ialah melakukan pengujian tegangan pada masing-masing rangkaian. Kemudian dilanjutkan dengan melakukan pengujian fungsi masing-masing rangkaian dengan demikian dapat diketahui apakah rangkaian dapat berjalan sesuai dengan yang diinginkan. Tahap terakhir ialah melakukan pengujian rangkian keseluruhan.

5.3.1.1 Pengujian Tegangan Sumber

Tahap pertama yang dilakukan adalah pengecekan catu daya. Catu daya yang digunakan memiliki keluaran sebesar 5 volt. Pengujian dilakukan dengan cara menggunakan multimeter. Hubungkan katup positif dari multimeter ke keluaran 5 volt dan hubungkan katup negatif multimeter ke ground pada catu daya.

Hasil pengujian tegangan catu daya 5 volt dapat dilihat pada tabel 5.1 :

Tabel 5.1 Tegangan Catu Daya 5 Volt

Sumber Arus	Tegangan Input	Tegangan Output
Catu Daya	5 Volt	4.5 Volt

5.3.1.2 Pengujian Tegangan Arduino

Arduino merupakan otak dari seluruh rangkaian. Semua rangkaian yang ada dikendalikan *input output*-nya oleh rangkaian mikrokontroler ini. Proses pengujian rangkaian ini adalah dengan menghubungkan pin VCC (+) dan GND (-) pada multimeter. Adapun hasil dari pengujian tegangan Arduino ini dapat dilihat pada tabel 5.2

Tabel 5.2 Pengujian Tegangan Arduino

Sumber Arus	Tegangan Input	Tegangan Output
ATmega	5 Volt	4.5 Volt

5.3.1.3 Pengujian Tegangan Pada Masing-masing Rangkaian

Pengujian tegangan yang dilakukan bertujuan untuk mengetahui seberapa besar tegangan pada setiap alat secara terpisah. Hal ini perlu diperhatikan karena beberapa komponen mempunyai tegangan yang berbeda disetiap rangkaian. Pengujian tegangan pada setiap rangkaian dapat dilihat pada tabel 5.3:

Tabel 5.3 Pengujian Tegangan

NO	Blok Rangkaian	Tegangan Yang diinginkan	Tegangan Sebenarnya
1	Regulator	5 volt	4.5 volt
2	Arduino	5 volt	4.5 Volt

5.3.1.4 Pengujian Sensor Kelembaban Tanah

Pengujian ini dilakukan untuk mengetahui apakah sensor kelembaban tanah bekerja sesuai dengan fungsinya. Teknis pengujian dilakukan dengan cara menghidupkan alat dengan menghubungkan konektor positif sensor kelembaban tanah dengan Vcc Arduino dan konektor negatif dengan GND Arduino, kemudian A0 sensor dihubungkan ke salah satu pin analog input pada arduino. Sensor kelembaban tanah diuji dengan cara menancapkan sensor kedalam tanah kemudian sensor akan mengirimkan nilai kelembaban tanah ke arduino yang menjadi acuan apakah tanah perlu disiram atau tidak. Untuk melakukan pengujian ini diperlukan program agar sensor ini mampu mendeteksi kelembaban tanah dengan baik. Tabel hasil pengujian sensor kelembaban tanah dapat dilihat dalam tabel 5.4 :

Tabel 5.4 Pengujian Sensor Kelembaban Tanah

No	Objek	Waktu (Detik)	Nilai Kelembaban(rh)
1	Tanah Kering	8 Detik	763
2	Pasir Basah	6 Detik	512
3	Air	3 Detik	216

Berikut adalah sketch program untuk pembacaan nilai kelembaban tanah:

```
#include <DallasTemperature.h>

void setup() {
  Serial.begin(9600);}

void loop() {
  int sensorTanah=analogRead(A0);
```

```

Serial.print("Kelembaban:");

Serial.println(sensorTanah);

delay(1000);}

```

5.3.1.5 Pengujian Sensor Suhu

Pengujian ini dilakukan untuk mengetahui apakah sensor suhu bekerja sesuai dengan fungsinya. Teknis pengujian dilakukan dengan cara menghidupkan alat dengan menghubungkan konektor positif sensor kelembaban tanah dengan Vcc Arduino dan konektor negatif dengan GND Arduino, kemudian A0 sensor dihubungkan ke salah satu pin analog input pada arduino. Sensor suhu diuji dengan cara meletakkan sensor pada suatu tempat atau ruangan untuk mengetahui suhu pada tempat tersebut kemudian sensor akan mengirimkan nilai suhu ke arduino yang menjadi acuan apakah suhu terlalu panas dan tidak baik untuk tanaman atau tidak. Untuk melakukan pengujian ini juga diperlukan program agar sensor ini mampu mendeteksi suhu dengan baik dan mengirimkannya pada arduino. Tabel hasil pengujian sensor suhu dapat dilihat dalam tabel 5.5:

Tabel 5.5 Pengujian Sensor Suhu

No	Tempat	Waktu (Detik)	Nilai Suhu
1	Dalam Ruangan	5 Detik	24 ⁰ c
2	Diluar Ruangan	7 Detik	27 ⁰ c

Berikut sketch program untuk pembacaan sensor suhu:

```
#include <OneWire.h>

#include <DallasTemperature.h>

#define ONE_WIRE_BUS 2

OneWire oneWire(ONE_WIRE_BUS);

DallasTemperature sensorsuhu(&oneWire);

float suhusekarang;

void setup() {

  Serial.begin(9600);

  sensorsuhu.begin();}

void loop() {

  suhusekarang=ambilsuhu();

  Serial.print("Suhu:");

  Serial.println(suhusekarang);

  delay(1000);}

float ambilsuhu(){

  sensorsuhu.requestTemperatures();
```

```
float suhu=sensorsuhu.getTempCByIndex(0);
```

```
return suhu;}
```

5.3.1.6 Pengujian Pompa air mini, kipas dan led

Pada tahap pengujian pompa air mini, kipas dan led yang dilakukan hanyalah memberi tegangan positif dan negative ke alat. Hasil pengujian dapat dilihat dalam tabel 5.6 :

Tabel 5.6 Pengujian pompa air, LED dan kipas

No	Sumber Tegangan	LED	Pompa air mini	Kipas
	5 V Arduino	Menyala Terang	Hidup	Berputar Normal
	3.3 V Arduino	Menyala Redup	Hidup	Berputar Pelan

Dari tabel hasil pengujian 5.6 dapat disimpulkan bahwa pompa air mini, kipas dan led dapat beroperasi dengan baik.

5.3.1.7 Pengujian Sensor Kelembaban tanah dengan Pompa air

Pengujian ini dilakukan untuk mengetahui apakah pompa air akan menyala atau tidak berdasarkan nilai dari sensor kelembaban tanah tentunya pengujian ini membutuhkan program untuk menjalankannya, sebelum menjalankannya program telah dibuat sesuai dengan karakteristik tanaman seledri yaitu pompa air akan menyala saat kelembaban tanah diatas 600 rh dan akan mati saat kelembaban tanah diatas 500 rh . Hasil Pengujian dapat dilihat dalam tabel 5.7:

Tabel 5.7 Sensor Kelembaban Tanah dan Pompa air

No	Objek	Waktu (Detik)	Nilai Kelembaban	Pompa air
1	Tanah Kering	6 Detik	763	Menyala
2	Pasir Basah	5 Detik	549	Mati
3	Air	3 Detik	240	Mati

Dari hasil pengujian ini dapat disimpulkan bahwa sensor kelembaban tanah dan pompa air mini dapat berfungsi dengan baik untuk menyiram tanaman secara otomatis. Berikut sketch program sensor kelembaban tanah dan pompa air:

```
#include <OneWire.h>

#include <DallasTemperature.h>

int pompaair=2;

void setup() {

  Serial.begin(9600);

  pinMode(pompaair,OUTPUT);}

void loop() {

  int sensorTanah=analogRead(A0);

  Serial.print("Kelembaban:");

  Serial.println(sensorTanah);

  delay(1000);

  if(sensorTanah>650)digitalWrite(pompaair,HIGH);

  if(sensorTanah<500)digitalWrite(pompaair,LOW); }
```

5.3.1.8 Pengujian Sensor Suhu dan Kipas

Pengujian ini dilakukan untuk mengetahui apakah kipas akan menyala atau tidak berdasarkan nilai dari sensor suhu tentunya pengujian ini membutuhkan program untuk menjalankannya, sebelum menjalankannya program telah dibuat dengan ketentuan kipas akan menyala sesuai dengan kebutuhan tanaman seledri saat suhu didalam box diatas 29°C dan akan mati saat suhu didalam box sudah dibawah 27°C . Hasil Pengujian dapat dilihat dalam tabel 5.8:

Tabel 5.8 Sensor Suhu dan Kipas

No	Objek	Waktu (Detik)	Nilai Suhu	Kipas
1	Dalam Ruangan	4 Detik	27°c	Tidak menyala
2	Luar Ruangan	7 Detik	29°c	Menyala

Dari hasil pengujian ini dapat disimpulkan bahwa sensor suhu dan kipas dapat berfungsi dengan baik untuk memberikan suhu yang baik untuk tanaman.

Dibawah ini sketch program sensor suhu dan kipas:

```
#include <OneWire.h>

#include <DallasTemperature.h>

#define ONE_WIRE_BUS 2

OneWire oneWire(ONE_WIRE_BUS);

DallasTemperature sensorsuhu(&oneWire);

float suhusekarang;

int kipas=8;

void setup() {
```

```

Serial.begin(9600);

sensorsuhu.begin();

pinMode(kipas,OUTPUT);}

void loop() {

suhusekarang=ambilsuhu();

Serial.print("Suhu:");

Serial.println(suhusekarang);

delay(1000);

if(suhusekarang>30)digitalWrite(kipas,HIGH);

if(suhusekarang<28)digitalWrite(kipas,LOW); }

float ambilsuhu(){

sensorsuhu.requestTemperatures();

float suhu=sensorsuhu.getTempCByIndex(0);

return suhu;}

```

5.3.1.9 Pengujian Keseluruhan

Tabel 5.9 Pengujian Keseluruhan

NO.	Sumber Tegangan	Arduino	RTC	Objek		Pompa air	Pompa pupuk	Kipas	LED
				Sensor suhu	Sensor tanah				
1	5V	1	8.00	28 ⁰ C	340 rh	0	1	0	1
2	5V	1	9.00	28 ⁰ C	1022 rh	1	0	0	0
3	5V	1	10.00	30 ⁰ C	450 rh	0	0	1	1
4	5V	1	11.00	27 ⁰ C	754 rh	1	1	0	0
5	5V	1	12.00	27 ⁰ C	337 rh	0	0	0	1
6	5V	1	13.00	31 ⁰ C	592 rh	0	0	1	0
7	5V	1	14.00	29 ⁰ C	563 rh	0	1	0	1

8	5V	1	15.00	27 ⁰ C	466 rh	0	0	0	0
9	5V	1	16.00	28 ⁰ C	268 rh	0	0	0	1
10	5V	1	17.00	30 ⁰ C	387 rh	0	1	1	0

Sebelum melakukan pengujian ini alat sudah diprogram dengan ketentuan pompa air akan menyala apabila nilai kelembaban berada diatas 600rh, pompa pupuk akan menyala 3 jam sekali, kipas akan menyala saat suhu berada di atas 29⁰C dan LED akan menyala saat RTC menunjukkan waktu genap. Keterangan : 0 = mati dan 1 = menyala.

5.4 ANALISA SISTEM SECARA KESELURUHAN

Untuk mendeteksi apabila terjadi kesalahan setelah uji coba, maka perlu dilakukan analisa sistem secara keseluruhan. Dari seluruh proses yang telah dilakukan, baik pengujian perangkat keras maupun perangkat lunak, dapat dikatakan bahwa alat ini dapat berfungsi sebagaimana yang penulis inginkan. Proses pembacaan kelembaban tanah dan suhu pun tidak terjadi kesalahan pembacaan data, pompa air, led dan kipas dapat berkerja sesuai dengan input yang diberikan.

Cara kerjanya terdapat 4 tahap yaitu :

1. Penyiraman otomatis dengan cara sensor kelembaban tanah membaca kelembaban tanah disekitar tanaman kemudian mengirimkannya ke arduino, kelembaban tanah tentu juga dipengaruhi oleh suhu, semakin panas suhu maka kelembaban tanah akan semakin rendah. Setelah pembacaan kelembaban tanah selanjutnya sensor akan mengirimkan

nilai kelembaban tanah kepada arduino apakah perlu penyiraman atau tidak jika perlu penyiraman maka arduino akan menyalakan pompa air.

2. Pengatur suhu otomatis yaitu membaca suhu didalam box tanaman dan mengirimkan nilai suhu ke arduino kemudian arduino akan memutuskan apakah suhu baik atau tidak baik untuk tanaman dengan cara menyalakan kipas.
3. Pemberian pupuk otomatis, ini bekerja dengan mengatur waktu pada arduino jika sudah sampai pada waktu yang ditentukan maka pompa pupuk akan menyala.
4. Penyinaran otomatis, penyinaran otomatis ini juga berdasarkan pengaturan waktu yang telah dibuat pada program arduino dalam hal ini led akan menyala pada waktu dan durasi yang telah ditentukan berdasarkan kebutuhan tanaman.