

BAB IV

PEMBAHASAN

A. Analisis Hasil

1. Perencanaan (*Planning*)

a. Mengidentifikasi Masalah

- 1) Pengeringan gabah secara konvensional yang masih tergantung pada sinar matahari.
- 2) Kualitas beras yang kurang maksimal.

b. Mengidentifikasi Penyelesaian Masalah

- 1) Pengeringan gabah yang masih konvensional. Maka dari itu perlu dilakukan pemanfaatan mesin pengering mekanik sehingga pada saat pengeringan tidak bergantung pada kondisi cuaca.
- 2) Menggunakan sensor suhu DHT11 untuk mengontrol suhu ruang pemanas sehingga kadar air dalam gabah tidak terlalu kering dan tidak terlalu basah.

2. Analisis Kebutuhan Sistem

a. Kebutuhan Pengguna

- 1) Kebutuhan Masukan (*Input*)
 - a) Input sensor suhu DHT11.
 - b) Input sensor kapasitif.
- 2) Kebutuhan Proses

- a) Proses pengeringan kadar air pada gabah.
- b) Proses menampilkan karakter pada LCD.
- c) Proses komunikasi serial Arduino Mega dengan Dinamo.
- d) Proses Komunikasi komponen IoT.
- e) Proses pembacaan data dari sensor suhu.

3) Kebutuhan keluaran (*Output*)

- a) Suhu panas yang dihasilkan oleh *heater*.
- b) Penampilan karakter pada LCD
- c) Penampilan suhu yang masuk ke sensor pada LCD.
- d) Penampilan kadar air yang masuk ke sensor pada LCD.
- e) Indikator nyala LED.
- f) Penampilan laporan hasil kerja prototipe di aplikasi *smartphone*.

4) Kebutuhan Antarmuka

- a) Antarmuka tampilan suhu pada LCD.
- b) Antarmuka tampilan status prototipe pada aplikasi.
- c) Antarmuka tampilan laporan hasil pengeringan pada LCD.
- d) Antarmuka kondisi nyala LED.

b. Kebutuhan Perangkat Lunak

- 1) Windows 10
- 2) Android 8.0 Oreo
- 3) Arduino IDE 1.8.10
- 4) Fritzing version 0.9.3 beta

5) Blynk version 2.27.10

c. Kebutuhan Perangkat Keras

1) Arduino Mega

2) *Heater*

3) Motor AC

4) Sensor Suhu DHT11

5) NodeMCU ESP8266

6) LCD 16x2

7) Modul I2C

8) LED

9) Kipas angin DC (fan)

10) Relay

11) DC Step Down

12) Push Button

13) Resistor

14) Kapasitor

15) Dioda

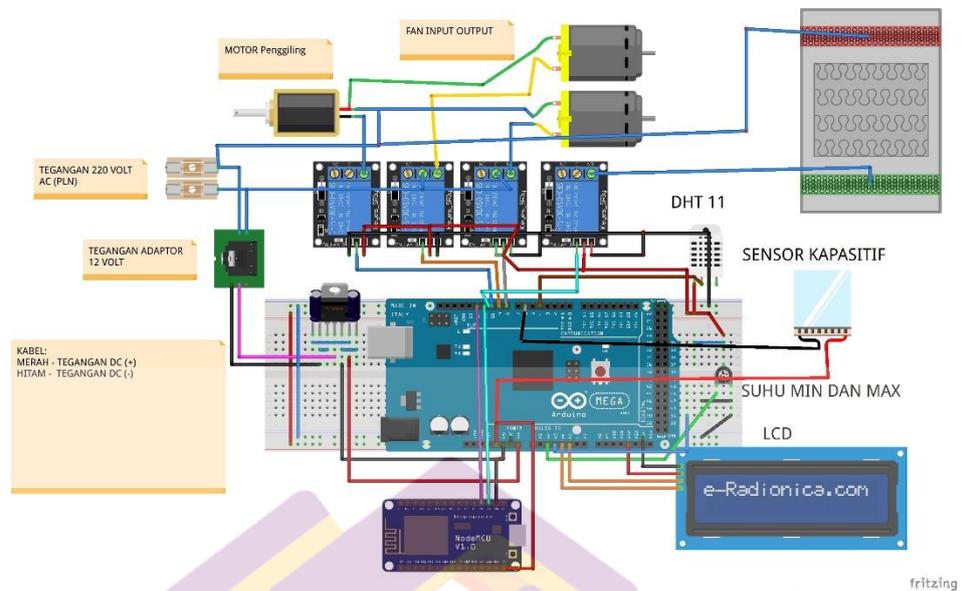
16) Papan PCB

17) Kabel

B. Pra Perancangan Skema

Proses pembuatan skema rangkaian ini di buat menggunakan *software*

Fritzing, adapun gambar skema rangkaian prototipe sebagai berikut:



Gambar 4.2 Skema Rangkaian Prototipe

Gambar 4.2 merupakan skema rangkaian yang sudah dibuat menggunakan software fritzing. Gambar skema ini nantinya akan dijadikan sebagai bahan layout untuk proses pencetakan pada PCB. Gambar rangkaian skema ini akan menentukan alur kerja pada system yang akan dibuat.

C. Komponen

1. Arduino Mega

Arduino Mega adalah komponen utama yang bertindak sebagai otak atau pengendali utama dari prototipe ini. Semua komponen berjalan dengan perintah yang telah terprogram di Arduino Mega. Tanpa alat ini prototipe tidak bisa dijalankan Arduino adalah komponen paling utama.

2. Penghantar Panas (*Heater*)

Komponen ini dipasang didalam ruang pengering dari prototipe karena komponen ini sebagai alat konversi dari energi listrik menjadi energi panas. Komponen ini akan menciptakan panas yang nantinya akan dialirkan melalui kawat kedalam ruang pengering.

3. Sensor Suhu DHT11

Komponen ini juga ditempatkan didalam ruang pengering yang mana sensor akan mengontrol suhu di dalam ruang pengering sehingga panas yang ada di dalam ruang pengering bisa terkontrol dengan baik.

4. Motor Sinkron (AC)

Komponen ini ditempatkan di sisi prototipe yang terpasang dengan sumbu as yang tersambung kedalam ruang pengering. Motor AC akan memutar as yang mana putaran dari as ini akan menggerakkan padi yang ada di dalam ruang pengering sehingga panas yang dialirkan bisa merata.

5. NodeMCU ESP8266

NodeMCU ESP8266 adalah komponen yang bertugas sebagai jembatan atau perantara antara prototipe dengan aplikasi *smarrthphone* yang dihubungkan dengan internet atau fitur ini sering disebut sebagai *Internet of Things (Iot)*. Semua hasil kerja dari prototipe nantinya bisa di tampilkan di aplikasi smartphome untuk memudahkan monitoring dari jarak jauh.

6. LED

Komponen ini digunakan sebagai indikator sistem. Ketika LED menyala itu adalah tanda sistem dalam keadaan hidup (*on*) dan berjalan. Begitupun sebaliknya, ketika LED mati (*off*) itu adalah tanda prototipe dalam keadaan tidak berjalan atau mati (*off*). LED ini juga bisa sebagai indikator adanya kerusakan pada komponen jika prototipe sudah dihidupkan tetapi LED masih dalam keadaan mati (*off*).

7. LCD 16x2

Komponen ini ditempatkan di tiang pada prototipe di atas modul I2C yang berguna untuk menampilkan beberapa hasil kerja dari prototipe seperti penampilan berapa isi daya dari baterai, tegangan yang masuk ke baterai, ataupun penampilan karakter lainnya untuk mempermudah dalam monitoring secara manual.

8. Relay

Komponen ini digunakan untuk memutus dan arus listrik pada heater jika suhu di dalam ruang pengering sudah terlalu panas (*overhead*). Jika suhu sudah terlalu rendah maka relay heater akan (*on*) dan kembali menghantarkan panas ke dalam ruang pengering.

9. Kipas angin DC (*fan*)

Komponen ini terletak di kedua sisi alat pengering digunakan sebagai suplai udara pada ruang pengering dan membuang uap panas jika suhu pada ruang pengering sudah terlalu panas.

10. Resistor

Komponen ini digunakan untuk membatasi jumlah arus yang mengalir ke komponen LED agar LED tidak dialiri arus listrik yang terlalu besar untuk menghindari korsleting LED.

11. Kapasitor

Komponen ini digunakan untuk menyimpan muatan listrik sementara. Saat tegangan yang akan mengalir ke suatu komponen lebih besar dari kebutuhannya, maka kapasitor akan menyimpan lebih dari listrik yang mengalir. Begitupun sebaliknya jika tegangan yang akan mengalir ke suatu komponen kurang dari kebutuhan komponen yang akan dialiri listrik, maka kapasitor akan membantu menambahkan tegangan listrik dari muatan yang telah disimpannya sehingga aliran listrik yang dialirkan ke komponen tersebut akan sesuai kebutuhan dari komponen itu sendiri.

12. DC Step Down

Komponen ini dipasang untuk menurunkan level tegangan listrik yang akan dialirkan dari adaptor ke Arduino Mega, NodeMCU ESP8266 agar menerima tegangan listrik dengan level tegangan listrik yang stabil agar komponen tersebut berjalan dengan baik.

13. Dioda

Komponen ini digunakan sebagai penyearah tegangan dan juga pengaman tegangan induksi.

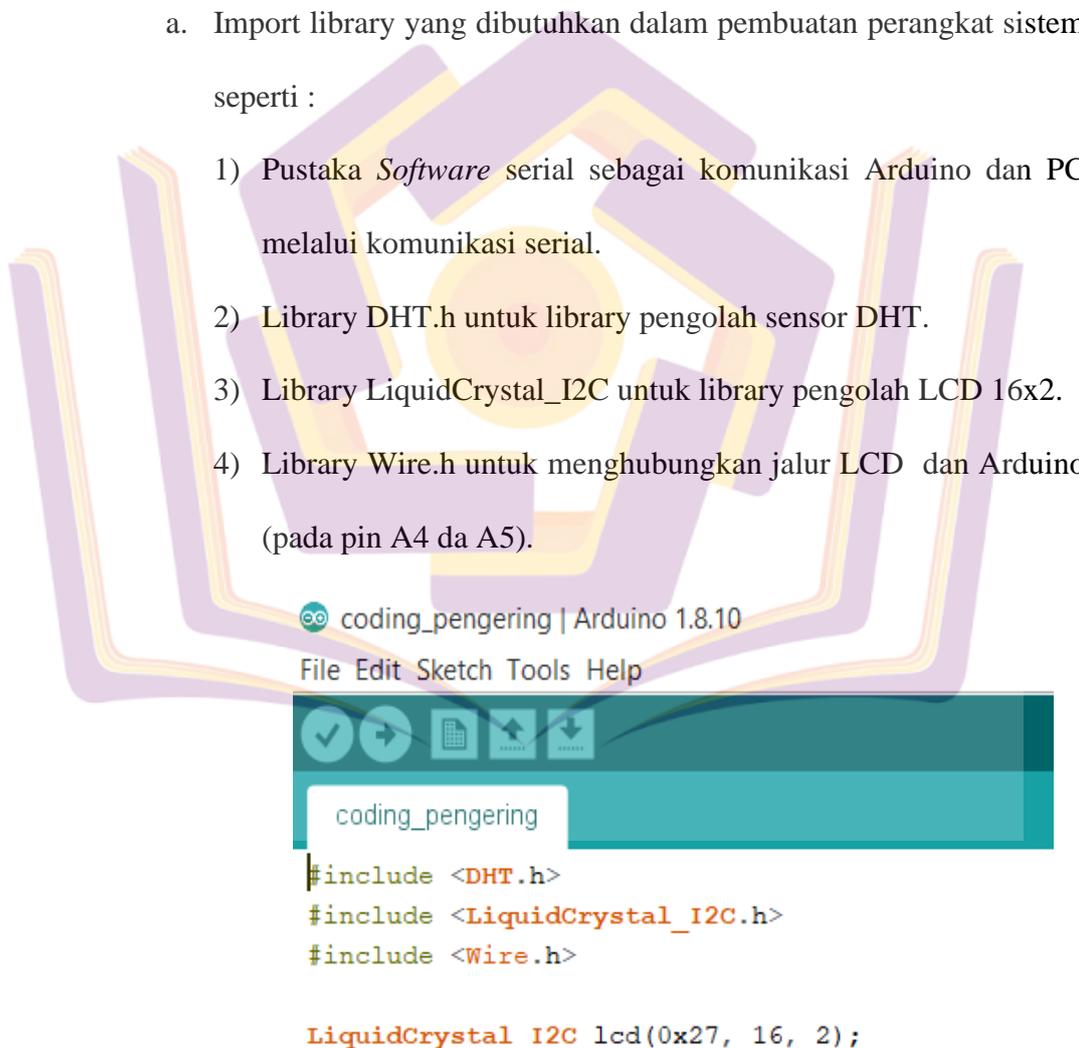
D. Pembuatan Program (*Coding*)

1. Pembuatan Program di Arduino IDE

Pembuatan atau penulisan kode program dilakukan di software Arduino IDE, yang nantinya program tersebut akan di upload atau di transfer ke perangkat keras Arduino Mega untuk mengendalikan input dan output sistem. Adapun langkah-langkahnya sebagai berikut:

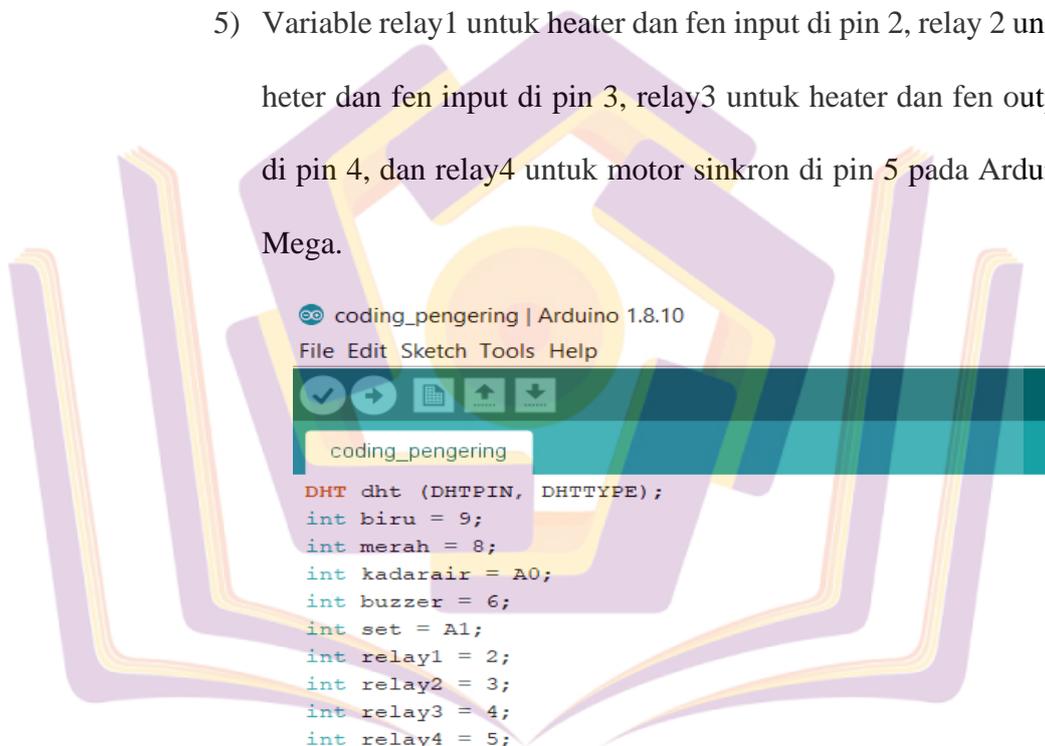
a. Import library yang dibutuhkan dalam pembuatan perangkat sistem seperti :

- 1) Pustaka *Software* serial sebagai komunikasi Arduino dan PC melalui komunikasi serial.
- 2) Library DHT.h untuk library pengolah sensor DHT.
- 3) Library LiquidCrystal_I2C untuk library pengolah LCD 16x2.
- 4) Library Wire.h untuk menghubungkan jalur LCD dan Arduino (pada pin A4 dan A5).



Gambar 4.3 *Library* Program

- b. Pemberian variable dan juga pin pada Arduino Mega antara lain:
- 1) Variabel biru untuk LED biru di pin 9 pada Arduino Mega.
 - 2) Variabel merah untuk LED merah di pin 8 Arduino Mega
 - 3) Variabel kadarair untuk sensor kapasitif di pin A0 pada Arduino Mega.
 - 4) Variabel set untuk batas suhu di pin A1 pada Arduino Mega.
 - 5) Variable relay1 untuk heater dan fen input di pin 2, relay 2 untuk heter dan fen input di pin 3, relay3 untuk heater dan fen output di pin 4, dan relay4 untuk motor sinkron di pin 5 pada Arduino Mega.



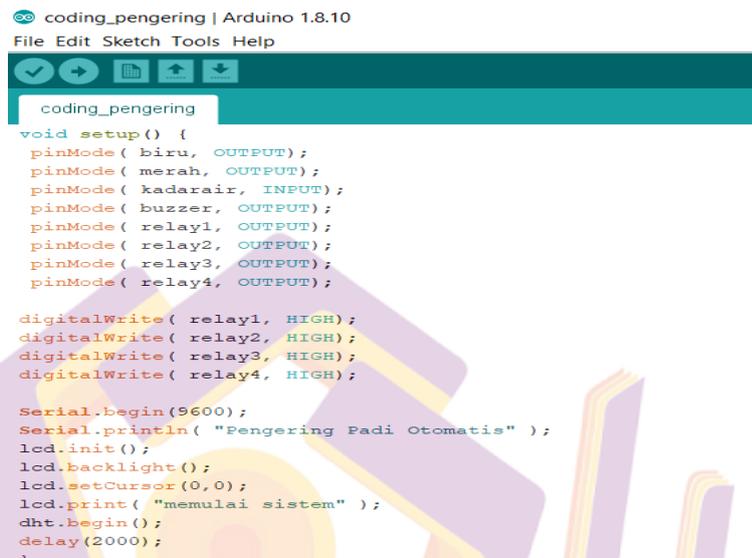
Gambar 4.4 Pembuatan variable program

c. Void setup

Dalam fungsi ini program akan melakukan program awal diantaranya:

- 1) Inisialisasi mode pin (variable pin) sebagai input atau output

- 2) Mengkondisikan relay1 sampai relay4 dalam kondisi off (HIGH) saat program pertama kali dihidupkan.
- 3) Menyiapkan kode (program) pembacaan sensor DHT.



```

coding_pengering | Arduino 1.8.10
File Edit Sketch Tools Help
coding_pengering
void setup() {
  pinMode( biru, OUTPUT);
  pinMode( merah, OUTPUT);
  pinMode( kadarair, INPUT);
  pinMode( buzzer, OUTPUT);
  pinMode( relay1, OUTPUT);
  pinMode( relay2, OUTPUT);
  pinMode( relay3, OUTPUT);
  pinMode( relay4, OUTPUT);

  digitalWrite( relay1, HIGH);
  digitalWrite( relay2, HIGH);
  digitalWrite( relay3, HIGH);
  digitalWrite( relay4, HIGH);

  Serial.begin(9600);
  Serial.println( "Pengering Padi Otomatis" );
  lcd.init();
  lcd.backlight();
  lcd.setCursor(0,0);
  lcd.print( "memulai sistem" );
  dht.begin();
  delay(2000);
}

```

Gambar 4.5 Inisialisasi Program

d. Void loop

Dalam fungsi ini program akan dijalankan berulang, diantaranya adalah :

- 1) Pembacaan data dari sensor DHT.
- 2) If jika data kosong maka ulangi membaca.
- 3) Lcd. menampilkan data pembacaan sensor pada tampilan LCD.
- 4) Pengkondisian heater, motor sinkron (pengaduk), fen *input*, dan fen *output*.
 - a) If 1 pengkondisian heater dan fen input jika suhu $< 60^{\circ}\text{C}$.
 - b) If 2 pengkondisian heater dan fen input jika suhu $> 60^{\circ}\text{C}$.
 - c) If 3 pengkondisian jika gabah telah kering

- d) If 4 pengkondisian jika belum kering maka motor pengaduk bekerja.



```

coding_pengering | Arduino 1.8.10
File Edit Sketch Tools Help
coding_pengering $
  lcd.setCursor(13,0);
  lcd.print( "k1=" );
  lcd.setCursor(13,1);
  lcd.print(h);

  if( t < 60) {
    digitalWrite(biru , HIGH);
    digitalWrite(relay1, LOW);
    digitalWrite(relay2, LOW);
  }
  if( t > 65) {
    digitalWrite(merah , HIGH);
    digitalWrite(relay1, HIGH);
    digitalWrite(relay2, LOW);
    digitalWrite(relay3, LOW);
  }
  if( t < 14) {
    digitalWrite(merah , HIGH);
    digitalWrite(biru , HIGH);
    digitalWrite(relay1, HIGH);
    digitalWrite(relay2, HIGH);
    digitalWrite(relay3, HIGH);
    digitalWrite(relay4, HIGH);
  }
  if( t > 14) {
    digitalWrite(relay4, LOW);
  }

```

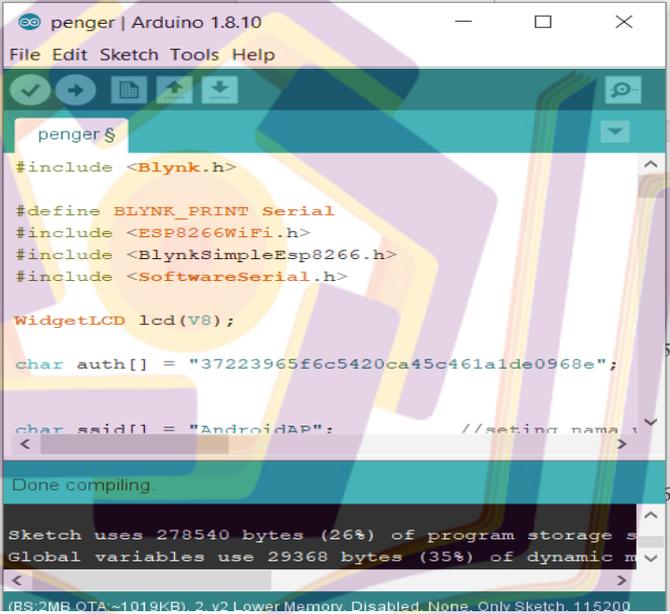
Gambar 4.6 Program *Looping*

e. Import library program Blynk

Dalam fungsi ini program akan melempar data ke aplikasi Blynk, diantaranya adalah:

- 1) Include Blynk.h untuk memanggil library Blynk
- 2) BLYNK_PRINT Serial untuk memanggil variabel blynk-print untuk komunikasi serial nodemcu.
- 3) Include ESP8266WiFi.h untuk memanggil library/pustaka wifi esp8266.

- 4) Include `BlynkSimpleEsp8266.h` untuk memanggil library/pustaka server blynk dengan perangkat nodemcu esp8266.
- 5) Include `SoftwareSerial.h` untuk memanggil library/pustaka `softwareserial` untuk jalur komunikasi nodemcu dengan PC (serial monitor).
- 6) `WidgetLCD` untuk membuat variabel lcd pada pin virtual 8 (V8) untuk menyimpan data pembacaan dari Arduino.



```

Arduino IDE Window: pengger | Arduino 1.8.10
File Edit Sketch Tools Help

pengger $
#include <Blynk.h>
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <SoftwareSerial.h>

WidgetLCD lcd(V8);

char auth[] = "37223965f6c5420ca45c461a1de0968e";

char ssid[] = "AndroidAP"; //setting nama
<
Done compiling.
Sketch uses 278540 bytes (26%) of program storage space. Maximum is 1024000 bytes.
Global variables use 29368 bytes (35%) of dynamic memory, leaving 40632 bytes free. Maximum is 65536 bytes.
(BS:2MB OTA~1019kB), 2.2V 1 Lower Memory, Disabled, None, Only Sketch, 115200

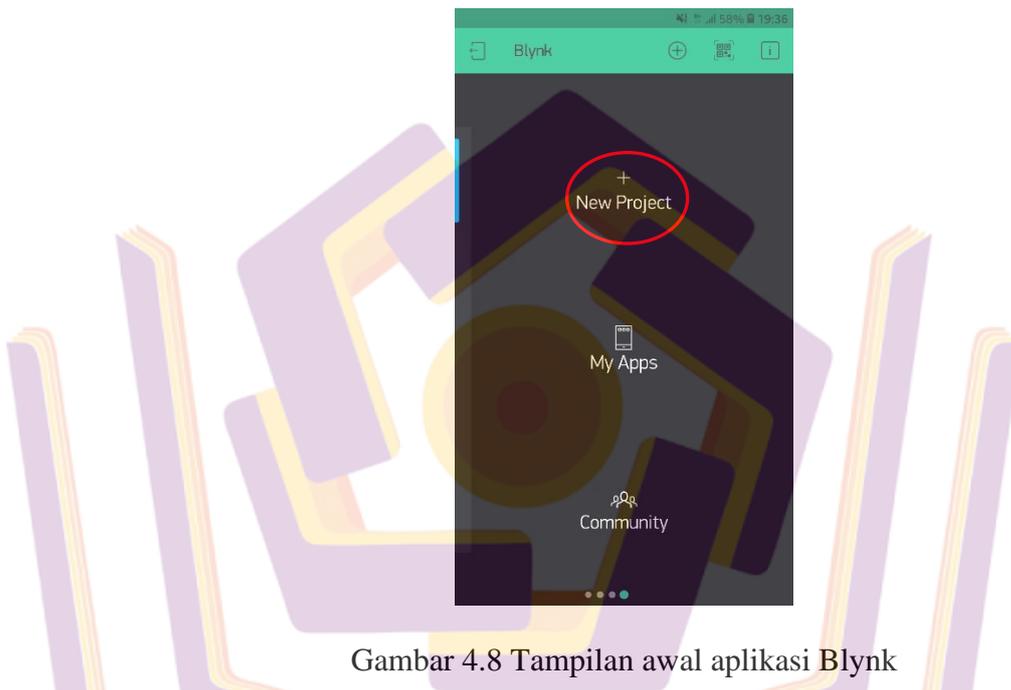
```

Gambar 4.7 Program untuk Blynk

2. Program *Software* Blynk

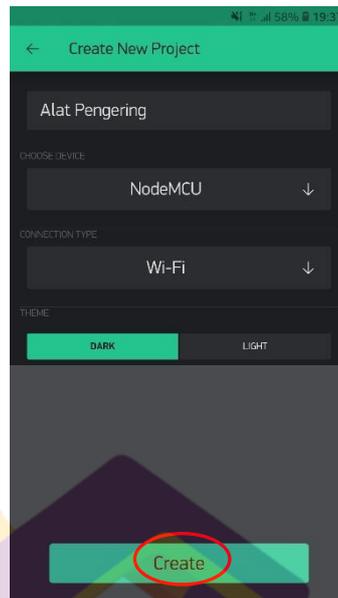
Pembuatan program untuk aplikasi di smartphone dilakukan di software Blynk, yang nantinya program tersebut akan dihubungkan dengan hardware di prototipe dengan fitur IoT (Internet of Things) yang ditanamkan di Hardware NodeMCU ESP8266. Adapun langkah-langkah pembuatan programnya sebagai berikut:

Yang pertama dilakukan adalah membuka aplikasi Blynk. Pembuatan program untuk aplikasi di smartphone dilakukan di software Blynk, yang nantinya program tersebut akan dihubungkan dengan hardware di prototipe dengan fitur IoT (Internet of Things) yang ditanamkan di Hardware NodeMCU ESP8266.



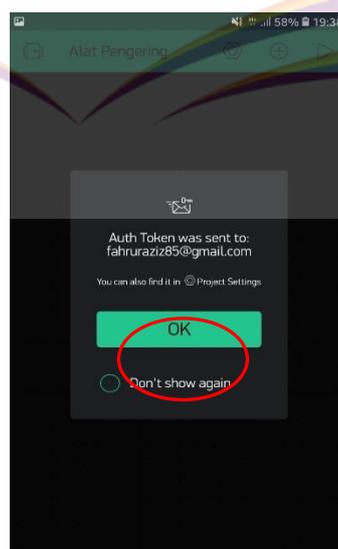
Gambar 4.8 Tampilan awal aplikasi Blynk

Setelah aplikasi Blynk dibuka, langkah selanjutnya adalah klik button “*New Project*” untuk membuat projek baru lalu beri nama projek, pilih perangkat dan tipe koneksi yang akan digunakan lalu klik button “*Create*”.

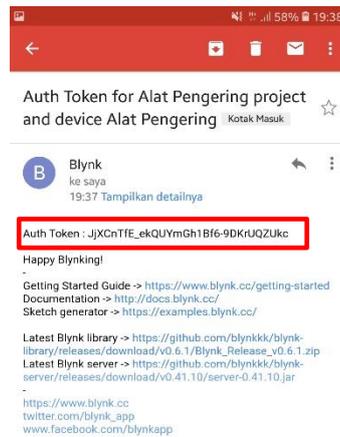


Gambar 4.9 Tampilan menu *New Project*

Setelah menekan button “*Create*” maka aplikasi akan memberikan akses token ke alamat *e-mail* dimana token tersebut adalah kode yang nantinya dimasukkan kedalam coding program di *software* Arduino IDE agar program aplikasi Blynk yang dibuat bisa terkoneksi ke perangkat NodeMCU ESP8266 melalui fitur IoT.



Gambar 4.10 Tampilan Pengiriman token autentikasi

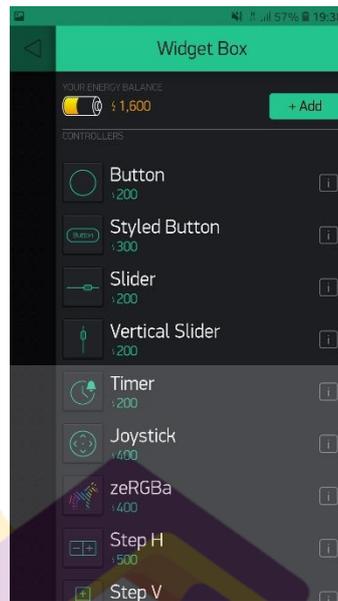


Gambar 4.11 *E-mail* token autentikasi

Setelah itu akan muncul tampilan awal project yang baru dibuat. Langkah selanjutnya yaitu menambahkan komponen widget yang dibutuhkan dengan menekan button “widget box” disebelah kanan atas aplikasi.

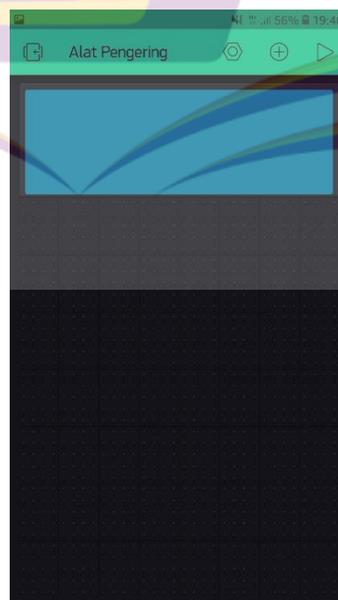


Gambar 4.12 Tampilan awal *project* yang dibuat

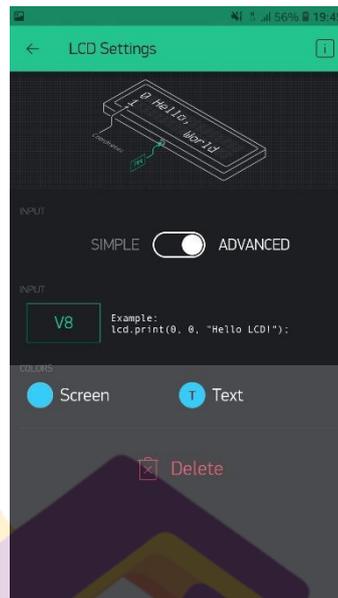


Gambar 4.13 Tampilan *Widget Box*

Pada menu “*wiget box*” ada banyak pilihan komponen namun hanya tersedia 2000 energi jika ingin menambahkan energi maka perlu biaya tambahan. Setelah masuk menu “*wiget box*”, *scroll* ke bawah dan pilih LCD.



Gambar 4.14 Tampilan *Widget LCD*



Gambar 4.15 Tampilan “LCD Setting”

Setelah widget muncul, lalu klik pada widget LCD untuk melakukan setting widget dan juga setting virtual input yang nantinya akan dimasukkan di coding program software Arduino IDE agar tampilan LCD di prototipe sama dengan tampilan LCD di program Blynk.

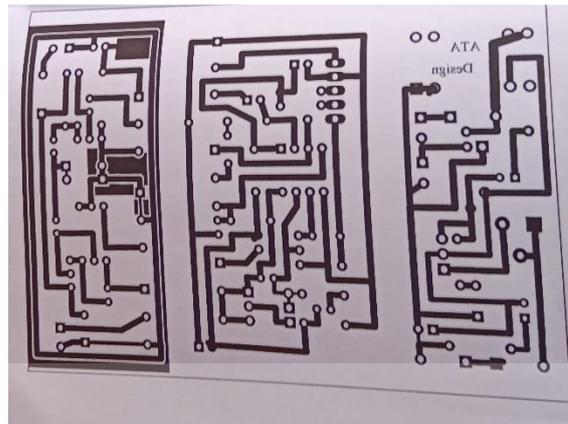
E. Perakitan Komponen Dan Transfer Program

1. Perakitan Komponen

Perakitan komponen bertujuan untuk menghubungkan komponen yang satu dengan yang lain, sehingga prototipe akan bekerja dengan baik. Adapun langkah-langkahnya sebagai berikut:

a. Pencetakan

Pada proses ini gambar desain layout rangkaian PCB yang telah dibuat di software fritzing dicetak menggunakan kertas HVS polos berwarna putih.



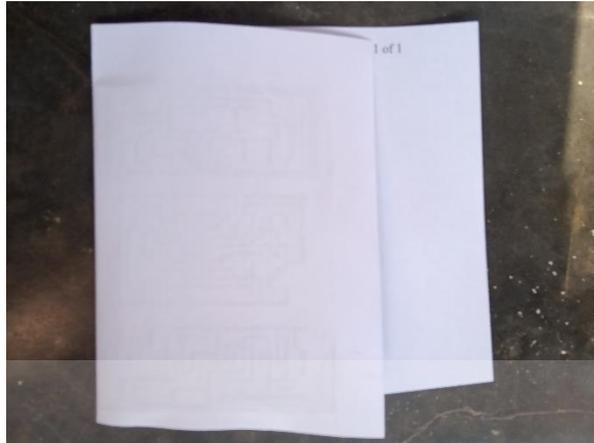
Gambar 4.16 Desain *layout* PCB

Setelah desain layout PCB dicetak, langkah selanjutnya adalah memotong board PCB polos sesuai dengan ukuran desain layout PCB yang telah dicetak.



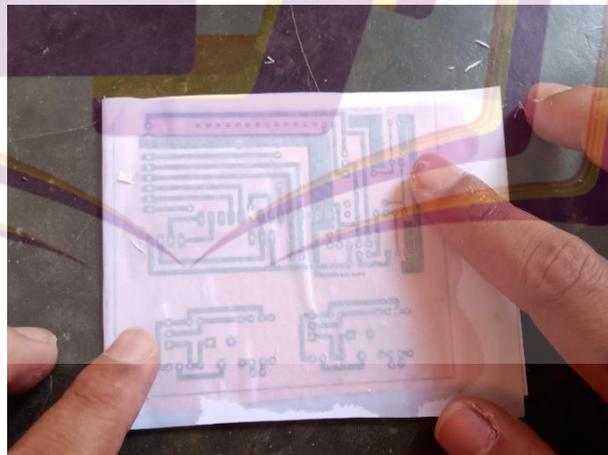
Gambar 4.17 Papan PCB Polos

Setelah itu letakkan kertas cetakan layout PCB diatas papan PCB yang telah dipotong dengan posisi kertas cetakan layout PCB yang terdapat tintanya berada di bawah atau di tempelkan ke papan PCB.



Gambar 4.18 Kertas desain layout PCB diatas papan PCB

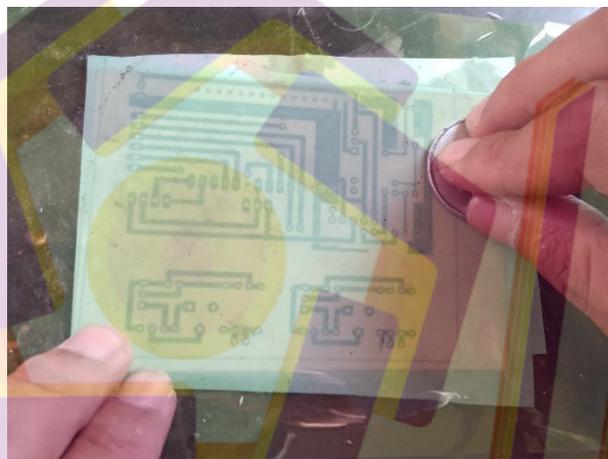
Siapkan gelas kecil dan masukkan lotion autan dan air putih secukupnya, campurkan dan aduk sampai merata. Cairan ini digunakan untuk membasahi kertas cetakan layout PCB yang ditempelkan diatas papan PCB agar tinta cetakan layout PCB luntur dan menempel di papan PCB.



Gambar 4.19 Kertas desain layout PCB yang telah dibasahi

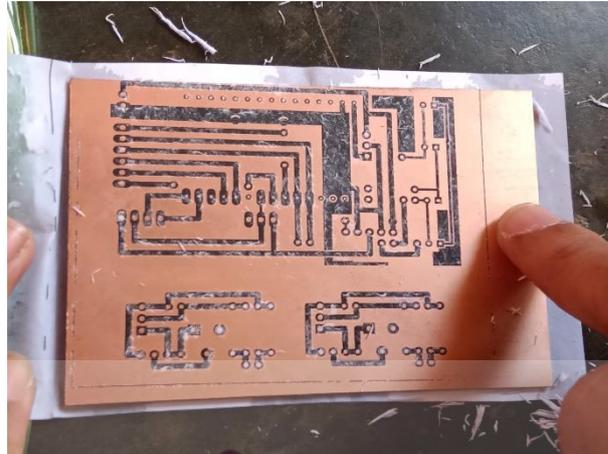
Setelah kertas cetakan layout PCB telah di basahi diatas papan PCB, langkah selanjutnya adalah menyiapkan plastik mika dan uang logam untuk proses penggosokan agar tinta dikertas cetakan desain

layout PCB menempel di papan PCB. Plastik mika diletakkan di atas kertas cetakan desain layout PCB yang tadi sudah dibasahi dengan larutan autan yang dicampur air. Digunakannya plastik mika ini bertujuan agar proses penggosokan lebih mudah dan tidak merobek kertas desain layout PCB. Untuk penggosokan ini memerlukan waktu kurang lebih 5 menit atau sampai proses penggosokan dirasa sudah merata.



Gambar 4.20 Proses Penggosokan desain layout PCB

Setelah proses penggosokan selesai, buka plastik mika dan kertas cetakan desain layout PCB secara perlahan. Bersihkan sisa-sisa kertas yang menempel di papan PCB secara perlahan agar papan PCB lebih bersih dan juga untuk menghindari tinta yang sudah menempel di papan PCB mengelupas.



Gambar 4.21 Papan PCB yang telah digosok

b. Pelarutan lapisan tembaga pada papan PCB

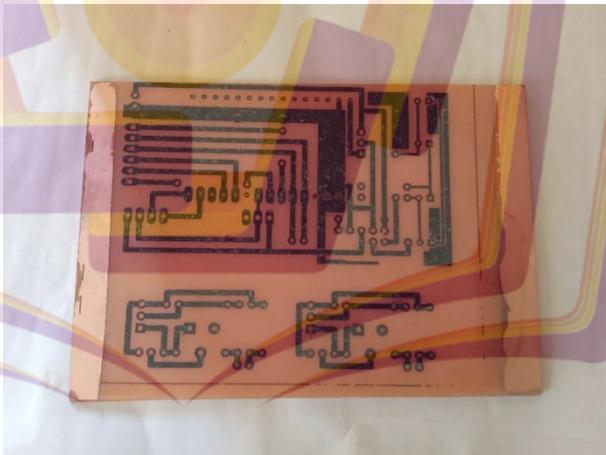
Proses ini bertujuan untuk menghilangkan atau melarutkan lapisan tembaga yang tidak terpakai atau tidak tertutup dengan tinta hitam hasil pencetakan menggunakan cairan lotion autan yang sebelumnya dilakukan sehingga nantinya akan menghasilkan jalur-jalur penghantar yang diperlukan untuk merangkai komponen elektronika.

Langkah pertama yang dilakukan adalah menyiapkan wadah untuk menampung air yang akan dicampurkan dengan cairan kimia FHCL, lalu masukkan papan PCB kedalam wadah tersebut. Goyangkan wadah sampai lapisan tembaga di papan PCB yang tidak tertutup tinta mengalami pengikisan.



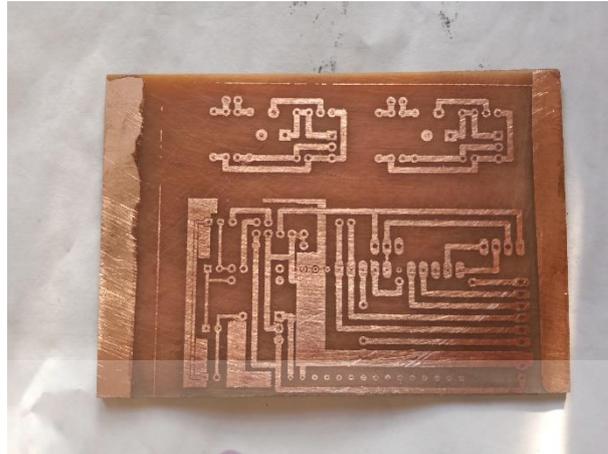
Gambar 4.22 Pelarutan papan PCB dengan larutan FeCl_3

Setelah proses pelarutan selesai, angkat papan PCB dan bersihkan papan PCB menggunakan air mengalir kemudian keringkan papan PCB sampai benar-benar kering.



Gambar 4.23 Papan PCB yang telah di keringkan

Proses selanjutnya adalah membersihkan atau menghilangkan tinta yang menempel di papan PCB sampai terlihat jalur-jalur tembaga yang nantinya akan menjadi jalur-jalur penghantar yang diperlukan untuk merangkai komponen.

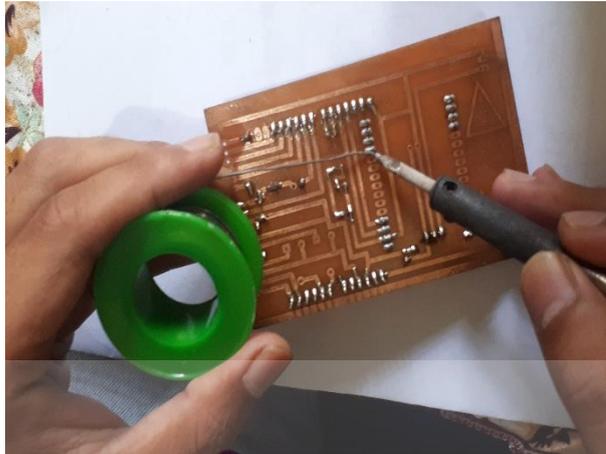


Gambar 4.24 Papan PCB yang telah dibersihkan

Setelah semua proses pelarutan selesai, langkah selanjutnya adalah pengeboran papan PCB untuk membuat lubang-lubang kecil yang digunakan sebagai penghubung antar komponen atau untuk penanaman kaki-kaki yang terdapat pada komponen.

c. Penyolderan Papan PCB

Pada proses ini yang pertama dilakukan adalah menyiapkan alat dan bahan yang dibutuhkan untuk proses penyolderan seperti solder, tinol timah, amplas halus, dan komponen elektronika lainnya. Setelah komponen yang dibutuhkan telah tersedia semua, selanjutnya masuk pada proses pengamplasan permukaan PCB yang akan disolder untuk menghilangkan lapisan yang membuat timah tidak menempel sempurna, lalu masukkan pangkal kaki komponen pada lubang di papan PCB. Setelah solder panas, arahkan ujung solder keujung tinol timah dan arahkan ke pangkal kaki komponen yang sudah menempel di papan PCB kemudian solder semua pangkal kaki komponen sampai tertutup timah dengan sempurna.



Gambar 4.25 Penyolderan Papan PCB

d. Rangkaian alat keseluruhan

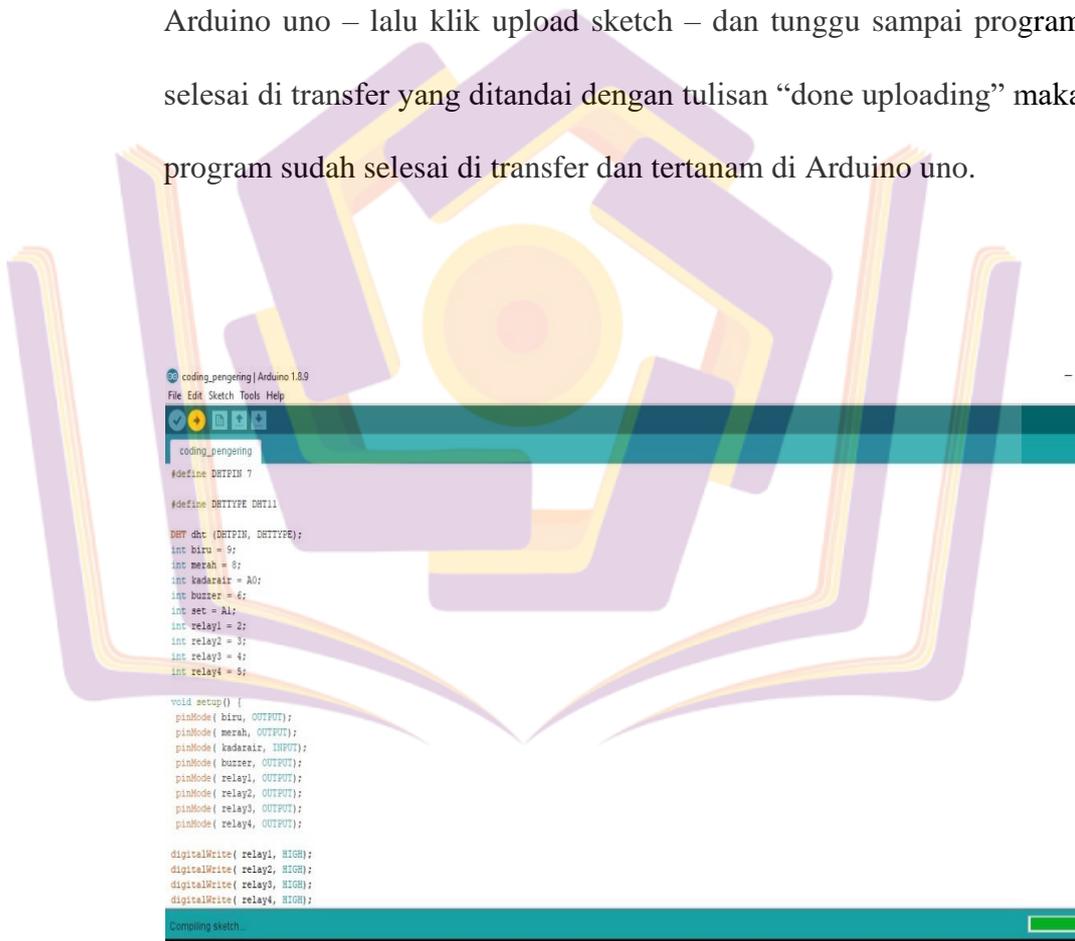
Rangkaian ini merupakan prototipe alat pengering gabah berbasis Arduino Mega. Pada rangkaian alat ini terdiri dari beberapa komponen yaitu : Arduino Mega, Heater, Sensor Suhu, Sensor Kapasitif, Kipas angin DC (fan), motor sinkron (AC), NodeMCU ESP8266, Relay, IC regulator LM7809, LCD 16x2, LED, Relay, Kapasitor, Dioda, Papan PCB, Push Button, dan Kabel.



Gambar 2.26 Rangkaian alat keseluruhan

2. Proses Transfer Program

Proses ini bertujuan untuk mentransfer program yang sebelumnya sudah di buat menggunakan software Arduino IDE yang kemudian di compile atau di transfer ke Arduino uno menggunakan kabel USB mikro yang dihubungkan ke port USB Arduino uno. Cara transfer program ke Arduino uno yaitu pertama klik tool pada software - klik board – pilih Arduino uno – lalu klik upload sketch – dan tunggu sampai program selesai di transfer yang ditandai dengan tulisan “done uploading” maka program sudah selesai di transfer dan tertanam di Arduino uno.



Gambar 4.25 Proses *Transfer* Program

F. Pengujian Sistem

Pengujian sistem ini bertujuan untuk mengetahui bahwa setiap komponen berfungsi dengan baik, sehingga alat dapat bekerja secara

maksimal sesuai dengan yang diharapkan dan digunakan untuk mengukur sistem itu sendiri. Adapun beberapa pengujian system antara lain:

1. Pengujian sensor suhu dan *heater*

Pengujian pada sensor suhu dilakukan untuk mengetahui apakah sensor suhu dapat membaca suhu ruang pengering dengan baik. Sensor suhu ini menggunakan sensor suhu DHT11 yang ditanam didalam ruang pengering. Pengujian sensor suhu dan heater dilakukan dengan cara menghubungkan heater dengan tegangan PLN (220 volt AC). Tahapan ini bertujuan untuk mengetahui besarnya suhu maksimal yang dihasilkan oleh komponen *heater*, suhu awal sebelum *heater* dinyalakan yaitu sebesar 29°C.

Tabel 4.1 Pengujian sensor DHT11 dan *Heater* ruang tanpa gabah

Waktu (menit)	suhu
0	29°C
10	42°C
20	54°C
30	56°C
40	58°C
50	59°C
60	59°C

Pada tabel 4.1 pengujian pertama dilakukan dalam waktu 1 jam (60 menit) *heater* dapat menghasilkan suhu sebesar 59°C pada kondisi ruang pemanas tanpa gabah.

Tabel 4.2 Pengujian sensor DHT11 dan *heater* ruang berisi gabah

Waktu (menit)	suhu
0	29°C
10	40°C
20	49°C
30	52°C
40	55°C
50	57°C
60	58°C

Tabel 4.2 pada pengujian kedua diambil data per sepuluh menit dalam waktu satu jam (60 menit) *heater* dapat menghasilkan suhu panas sebesar 58°C kondisi ruang berisi gabah.

Pada pengujian pertama dan kedua terdapat selisih kenaikan suhu, dimana pada 10 menit pertama terjadi selisih kenaikan sebesar 2°C selisih yang tipis ini dikarenakan *heater* yang baru dinyalakan sehingga belum maksimal untuk menghantarkan panas. Dimenit ke 20 dan 30 *heater* sudah mulai bekerja secara maksimal sehingga selisih kenaikan suhu sebesar 5°C dan 4°C. Sedangkan dimenit 40, 50, dan 60 terjadi penurunan selisih dari menit-menit sebelumnya dikarenakan *heater* berdaya 25 watt hanya dapat menghasilkan suhu panas sebesar 59°C.

2. Pengujian sensor kadar air

Untuk pengujian kadar air menggunakan sensor kapasitif, pengujian ini dilakukan dengan cara meletakkan sensor kedalam wadah yang telah

berisi gabah untuk mendeteksi kadar air dalam gabah. Pengujian dilakukan dengan meletakkan gabah dalam empat wadah dengan kadar air awal yang berbeda. Kemudian akan dimasukan dalam ruang pengering selama 60 menit maka akan keluar hasil akhir dengan rumus

$$\left(\frac{\text{data analog}}{\text{jarak air dan udara}} \times 100 = \text{kadar air akhir} \right).$$

Data analog merupakan hasil yang di inputkan oleh sensor DHT11 ke Arduino mega kemudian ditampilkan sebagai output data analog. Jarak air dan udara merupakan hasil pengujian sensor kapasitif dengan kadar air mutlak 10-800 kemudian ditentukan jarak sebesar 790. Sedangkan untuk angka 100 dihasilkan dari mapping arduino mega. Setelah semua dijumlahkan dengan rumus yang telah dibuat maka akan keluar persentase hasil kadar air akhir

Tabel 4.3 Pengujian waktu kadar air

Wadah	Kadar air awal	Waktu	Kadar air akhir
A	30%	60 menit	16%
B	22%	60 menit	14%
C	17%	60 menit	14%
D	16%	60 menit	14%

3. Pengujian tampilan LCD 16x2.

Pada pengujian tampilan LCD digunakan sebuah LCD berkarakter 16 x 2. LCD disini berfungsi untuk menampilkan data input suhu ruang pemanas oleh sensor DHT11 dan nata input kadar air dalam gabah oleh sensor kapasitif .



Gambar 4.26 menampilkan informasi saat sistem aktif

Gambar 4.26 merupakan gambar tampilan pada LCD pada saat menampilkan informasi sistem mulai mengecek komponen yang bekerja.

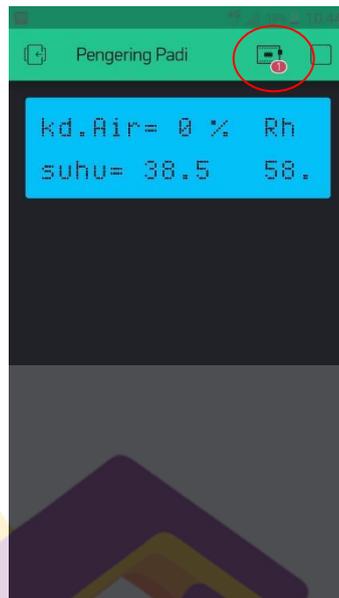


Gambar 4.27 menampilkan informasi kadar air dan suhu

Gambar 4.27 merupakan gambar informasi inputan dari sensor suhu DHT11 dan sensor kapasitif kemudian ditampilkan melalui LCD.

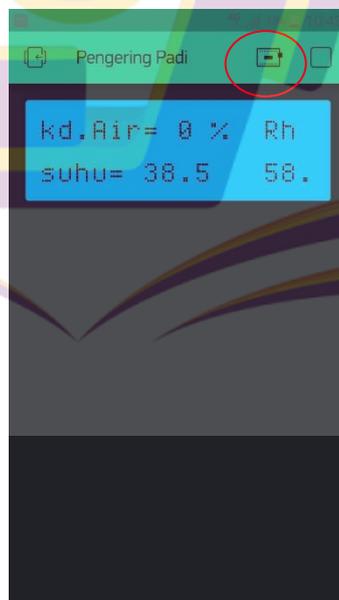
4. Pengujian tampilan ke aplikasi Blynk

Pada pengujian aplikasi ini akan di tampilkan data yang sesuai pada tampilan LCD.



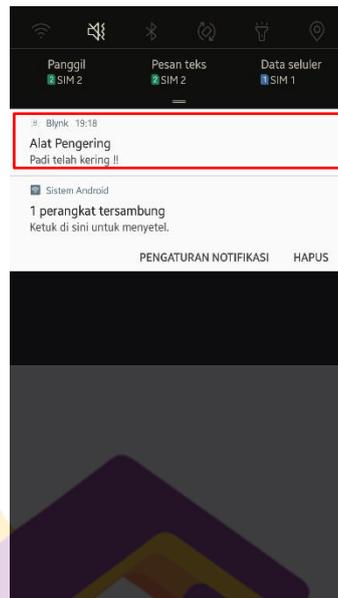
Gambar 4.28 Jika aplikasi tidak terhubung ke server

Gambar 4.28 merupakan gambar tampilan layar pada aplikasi Blynk jika tidak terhubung ke server.



Gambar 4.29 Jika aplikasi terhubung ke server

Gambar 4.9 merupakan gambar tampilan layer pada aplikasi Blynk yang sudah terkoneksi dengan server.



Gambar 4.30 Notifikasi pada *smartphone*

Gambar 4.30 merupakan gambar tampilan notifikasi pada layar *smartphone* jika padi sudah kering.

5. Pengujian keseluruhan

Pada pengujian keseluruhan semua komponen ini akan diuji apakah komponen dapat bekerja secara maksimal atau tidak. Pengujian keseluruhan ini diantaranya:

- a. Pengujian sensor DHT11 sebagai input suhu ruangan.
- b. Pengujian sensor kapasitif sebagai input kadar air pada gabah.
- c. Pengujian heater sebagai output untuk menghantarkan panas pada ruang pengering.
- d. Pengujian fan in dan fan out sebagai keluar masuknya udara pada alat pengering.

- e. Pengujian motor pengaduk sebagai alat untuk mengaduk gabah yang ada didalam ruang pengering agar gabah dapat menerima panas secara merata.
- f. Pengujian *Buzzer* sebagai output alarm saat semua proses sudah selseai.

Tabel 4.4 Pengujian Keseluruhan

No	Suhu	Kadar air	Kondisi				
			Heater	Fan in	Fan out	Motor pengaduk	Buzzer
1	24°C	30%	ON	Hidup	Mati	Hidup	-
2	30°C	28%	ON	Hidup	Mati	Hidup	-
3	39°C	27%	ON	Hidup	Hidup	Hidup	-
4	41°C	25%	ON	Hidup	Hidup	Hidup	-
5	43°C	22%	ON	Hidup	Hidup	Hidup	-
6	44°C	19%	ON	Hidup	Hidup	Hidup	-
7	47°C	16%	ON	Hidup	Hidup	Hidup	-
8	50°C	16%	ON	Hidup	Hidup	Hidup	-
9	54°C	15%	ON	Hidup	Hidup	Hidup	-
10	58°C	14%	Mati	Mati	Mati	Mati	ON

Pada tabel 4.4 pengujian keseluruhan berjalan dengan optimal. Semua komponen mulai dari sensor DHT11 dan sensor kapasitif dapat membaca suhu dalam ruang dan kadar air dan menjadikan input data analog yang kemudian dikirim ke arduino mega, lalu oleh arduino mega dikonversikan menjadi data digital yang kemudian akan dikirimkan ke LCD dan NodeMCU. Komponen lain seperti *heater*, fan in, fan out,

motor pengaduk, dan *buzzer* juga berfungsi sesuai perintah yang sudah terprogram didalam arduino mega.

