

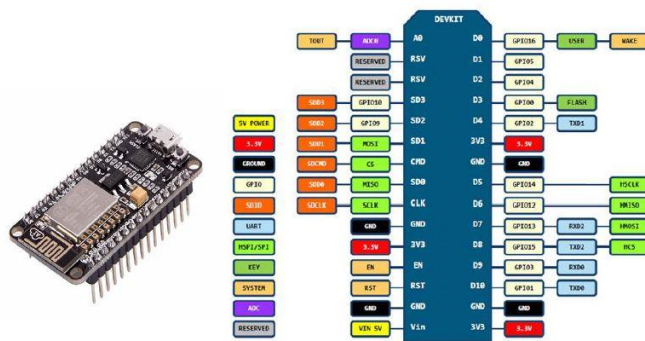
## BAB II

### LANDASAN TEORI

#### 2.1 Modul WiFi NodeMCU

Pada penelitian ini digunakan *board Nodemcu Esp8266*. Modul ini merupakan salah satu platform yang terbilang murah untuk ukuran *mikrokontroler* [7]. Meskipun murah namun nyatanya Esp-8266 ini terbilang efektif untuk mengontrol dan berkomunikasi melalui jaringan internet. Board ini bisa digunakan secara *stand alone* maupun dipasangkan dengan *mikrokontroler* tambahan seperti *Arduino uno*.

*NodeMCU* dapat dianalogikan sebagai *board arduino* nya ESP8266. Program ESP8266 sedikit merepotkan karena diperlukan beberapa teknik *wiring* serta tambahan modul USB *to* serial untuk mengunduh program. Namun *NodeMCU* telah mengemas ESP8266 ke dalam sebuah board yang kompak dengan berbagai fitur layaknya mikrokontroler + kapabilitas akses terhadap *Wifi* juga chip komunikasi USB *to* serial. Sehingga untuk memprogramnya hanya diperlukan ekstensi kabel data USB persis yang digunakan *charging smartphone*. *NodeMCU* ESP8266 merupakan *board* interaktif berbasis LUA. *NodeMCU* ESP8266 memiliki memori 4MB dan mempunyai *clock speed* 80 – 160 Mhz [3].



Gambar 2.1 Tampilan Fisik dan skema pin dari *NodeMCU* ESP8266

Masing-masing dari pin *NodeMCU* ESP8266 di jelaskan pada tabel 2.1 :

Tabel 2.1 Deskripsi *NodeMCU*

<b>Nama Pin</b>	<b>Detail</b>
Tx / GPIO-1	Dihubungkan dengan pin Rx untuk mengupload program
GPIO-2	Tx pada pin D1
GPIO – 0	I/O
CH_EN	Mengaktifkan chip- Aktif High
Rx/ GPIO-3	Menerima sinyal yang dikirim oleh Tx
GPIO4	I/O
GPIO5	I/O
GPIO9	SDD2
GPIO10	SDD3
GPIO12	HMISO
GPIO13	Rx pada D2 sebagai HMOSI
GPIO14	HSCLK
GPIO15	Tx pada D2 sebagai HCS
GPIO16	User
Vin	Power 5V
3V3	Input tegangan
Chipenable, active high	Chipenable, active high
RST	Untuk me-reset
GND	Ground

A0	Input ADC
----	-----------

## 2.2 Daya ( Power)

*Board* ESP8266 dapat disuplai dengan menghubungkan kabel USB ke komputer atau sumber tegangan yang didapat dari adaptor AC-DC atau baterai untuk menggunakannya. ESP8266 beroperasi dengan suplai tegangan 4.5 V sampai 9V biasanya menggunakan *port* USB.

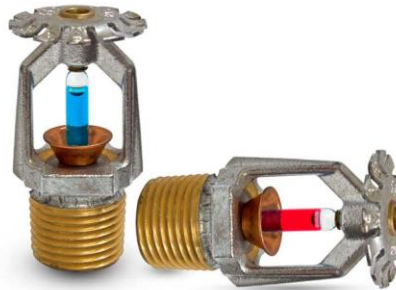
## 2.3 Spesifikasi

- Tegangan antarmuka komunikasi: 3.3V.
- Jenis antena: Tersedia antena PCB internal.
- Standar nirkabel 802.11 b / g / n
- WiFi di 2.4GHz, mendukung mode keamanan WPA / WPA2
- Mendukung tiga mode operasi STA / AP / STA + AP
- Tumpukan protokol TCP / IP bawaan untuk mendukung beberapa koneksi Klien TCP (5 MAX)
- D0 ~ D8, SD1 ~ SD3: digunakan sebagai GPIO, PWM, IIC, dll., Kemampuan driver port 15mA
- AD0: 1 saluran ADC
- Input daya: 4.5V ~ 9V (10VMAX), bertenaga USB
- Saat ini: transmisi kontinu: 70mA (200mA MAX), Siaga: 200uA
- Kecepatan transfer: 110-460800bps
- Mendukung antarmuka komunikasi data UART / GPIO
- Pembaruan firmware jarak jauh (OTA)
- Mendukung Smart Link Smart Networking
- Suhu kerja: -40 Deg ~ + 125 Deg
- Tipe Drive: Driver H-bridge ganda berdaya tinggi
- ESP8266 memiliki IO Pin
- Tidak perlu mengunduh pengaturan ulang

- Seperangkat alat yang bagus untuk mengembangkan ESP8266
- Ukuran flash: 4Mbyte

#### 2.4 *Sprinkle Air*

*Sprinkle* air ini berfungsi sebagai alat untuk menyiramkan air yang dipompa dari mesin pompa dc.



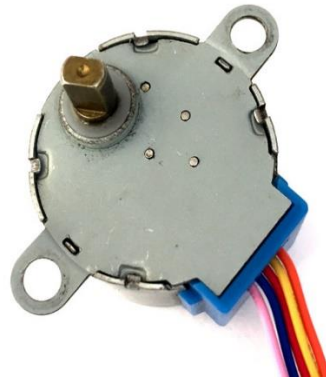
Gambar 2.2 Sprinkle Air

#### 2.5 *Stepper Motor*

*Stepper motor* adalah salah satu jenis motor DC yang bergerak atau berputar berdasarkan Langkah (step). *Input* pada motor *stepper* ini berasal dari pulsa digital yang diberikan oleh modul motor *stepper*. Langkah yang mengendalikan motor ini berasal dari kumparan-kumparan yang disusun menjadi sebuah *phase*. Motor *stepper* ini mengubah sinyal listrik menjadi Gerakan mekanis. Konstruksi tersebut menyebabkan motor *stepper* dapat diatur posisinya pada posisi tertentu, misalnya searah jarum jam atau berlawanan dengan arah jarum jam. Motor ini bergerak sesuai dengan sudut step yang bisa bervariasi tergantung dari motor *stepper* yang digunakan.

### 2.5.1 Motor Stepper 28byj-48

28BYJ-48 *Stepper Motor* adalah sebuah motor *stepper* 4 *phase* dengan tegangan *supply* sebesar 5V. Motor *Stepper* ini ideal digunakan dalam berbagai aplikasi elektronika seperti pembuatan robot, *remote control* dan perangkat lainnya yang membutuhkan suatu motor yang stepnya dapat diatur secara tepat.



Gambar 2.3 28BYJ-48 Stepper Motor

*Stepper* motor 28byj-48 ini banyak menjadi pilihan karena walaupun ukurannya kecil akan tetapi dia mempunyai torsi sekitar 34 Nm. Motor *Stepper* 28BYJ-48 adalah motor *stepper unipolar* dengan tegangan sebesar 5V DC, jumlah fase sebanyak 4, rasio variasi kecepatan 1/64, *stride angle* 5,625 derajat/64 dan frekuensi 100 Hz [6].

### 2.6 Driver Motor DC

*Driver Motor DC* adalah rangkaian yang tersusun dari transistor yang digunakan untuk menggerakkan motor DC. Motor memang dapat berputar hanya dengan daya DC, tapi tidak bisa diatur tanpa menggunakan *driver*, maka diperlukan suatu rangkaian *driver* yang berfungsi untuk mengatur kerja dari *motor*.

### 2.6.1 *Driver Motor DC ULN-2003A*

*Driver motor* ini memakai IC ULN-2003. IC ini merupakan penguat arus yang didalamnya menggunakan konfigurasi transistor Darlington yang setiap *chip* nya berisi 7 pasang NPN Darlington [2].

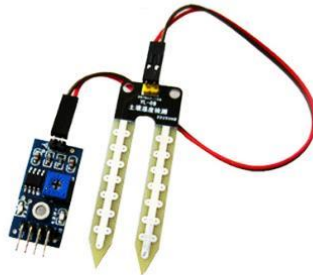


Gambar 2.4 Driver Motor DC ULN-2003A

Dari gambar di atas dapat dilihat ada 7 buah pin *input* dari IN1-IN7 serta memiliki pin untuk dihubungkan kepada motor *stepper* yang akan digunakan dan ada 2 buah *input* untuk tegangan.

### 2.7 *Sensor Soil Moisture*

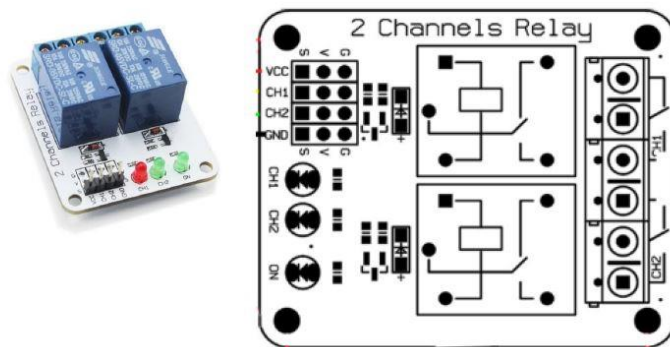
*Soil moisture* merupakan sensor yang digunakan untuk mengukur kelembaban tanah antara 0% sampai 100% atau nilai ADC 0-1024 dan memiliki tingkat akurasi sekitar 2% [11]. Sensor ini terdiri dari dua *elektrode (probe)* yang nantinya akan membaca kadar air didaerah sekitarnya, sehingga arus melewati dari satu elektrode ke elektrode yang lain. Arus dilewatkan pada *elektroda* di dalam tanah sehingga pengukuran nilai resistansi tanah menentukan kelembapannya dan sensor ini mempunyai 4 pin.



Gambar 2.5 Sensor Soil Moisture

### 2.8 Relay

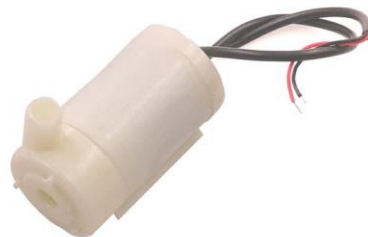
Modul *relay* adalah saklar yang dioperasikan secara elektrik yang memungkinkan untuk menghidupkan atau mematikan sirkuit dengan menggunakan voltase atau arus yang jauh lebih tinggi daripada yang dapat ditangani oleh *NodeMCU*. Tidak ada hubungan antara rangkaian tegangan rendah yang dioperasikan oleh *NodeMCU* dan rangkaian daya tinggi. *Relay* melindungi setiap rangkaian dari satu sama lain. Setiap saluran dalam modul ini memiliki tiga koneksi bernama NC, COM, dan NO. Bagian NC dan NO *relay* digunakan untuk menghubungkan sumber listrik (kabel fasa) dengan terminal SPO [3]. Jenis kontak yang digunakan di perangkat ini ialah *Normally Closed* (NC) sehingga pada kondisi arus normal sambungan sumber ke SPO tertutup. Sedangkan pada saat arus lebih, kontak akan otomatis diputuskan (open). Bagian lilitan (coil) *relay* disambungkan ke pin pengendali *NodeMCU* melalui *switch transistor*. Rangkaian *relay* yang dipilih ialah modul *relay 2 channel 5 V* seperti diperlihatkan pada gambar 2.6 berikut :



Gambar 2. 6 Relay

## 2.9 Mini Pompa Air DC

Alat ini merupakan motor dc yang berfungsi untuk memompa air



Gambar 2. 7 Mini Pompa Air DC

Pompa air mini ini menggunakan tegangan 5-12V biasanya di dapat dari baterai atau adaptor yang bernilai 12V dan pompa air mini ini bisa memompa air sebanyak 120L/Jam.



## 2.10 Aplikasi Blynk

Blynk adalah aplikasi untuk iOS dan OS Android untuk mengontrol Arduino, *NodeMCU*, Raspberry Pi dan sejenisnya melalui Internet. Aplikasi ini dapat digunakan untuk mengendalikan perangkat *hardware*, menampilkan data sensor, menyimpan data, visualisasi dan lain-lain [3].

Aplikasi Blynk memiliki 3 komponen utama yaitu Aplikasi, *Server*, dan *Libraries*. Blynk server berfungsi untuk menangani semua komunikasi diantara smartphone dan hardware. Widget yang tersedia pada Blynk di antaranya adalah *Button*, *Value Display*, *History Graph*, *Twitter* dan *Email*.

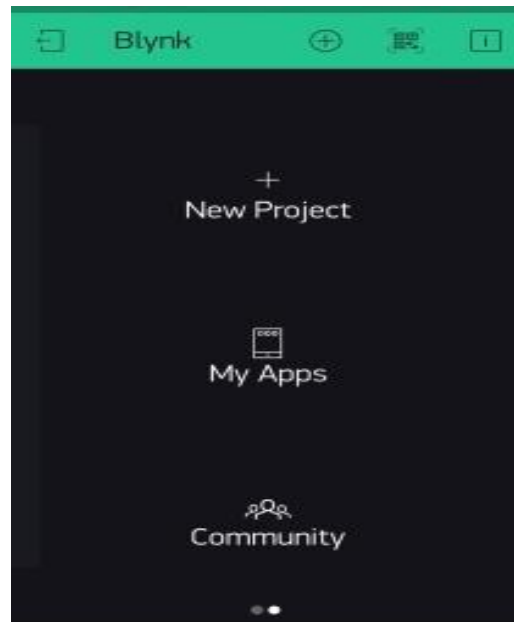


Gambar 2. 8 Logo Aplikasi Blynk

Blynk tidak terikat dengan beberapa jenis *mikrokontroller* namun harus didukung *hardware* yang dipilih. *NodeMCU* dikontrol dengan Internet melalui *WiFi*, chip ESP8266, Blynk akan dibuat *online* dan siap untuk *Internet of Things*. Cara pembuatan *user interface* pada Blynk sebagai berikut:

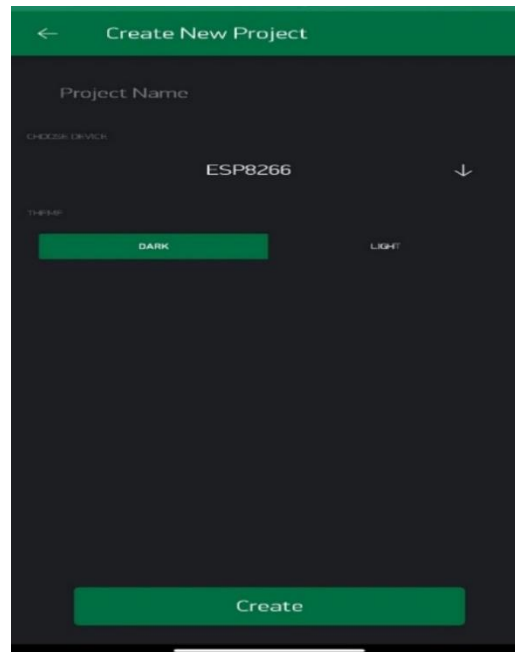
1. Membuka aplikasi
  - a. Membuka aplikasi Blynk, pertama membuat akun untuk mendapatkan auth token yang dikirim melalui email. Disarankan untuk meng install aplikasi Blynk versi 2.7.24 karena apabila mengunduh versi terbaru

maka tidak akan bisa untuk mendaftarkan akun baru dan *hardware* yang digunakan, kemudian pilih *create* seperti pada gambar berikut :



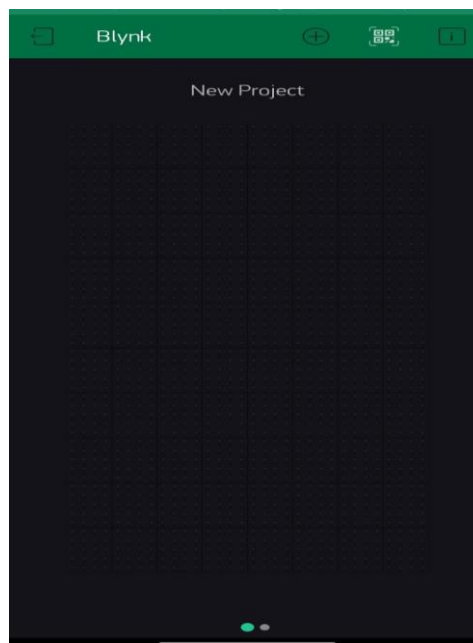
Gambar 2. 9 Tampilan Menu Aplikasi BYLNK

2. Setelah memasukkan akun maka kita akan menekan tombol *New Project* lalu masukan nama *project*, setelah itu kalian akan mendapat pilihan platform, di sini kita pilih *board* Esp8266 dan pilih koneksi ke *wi-fi* seperti pada gambar :



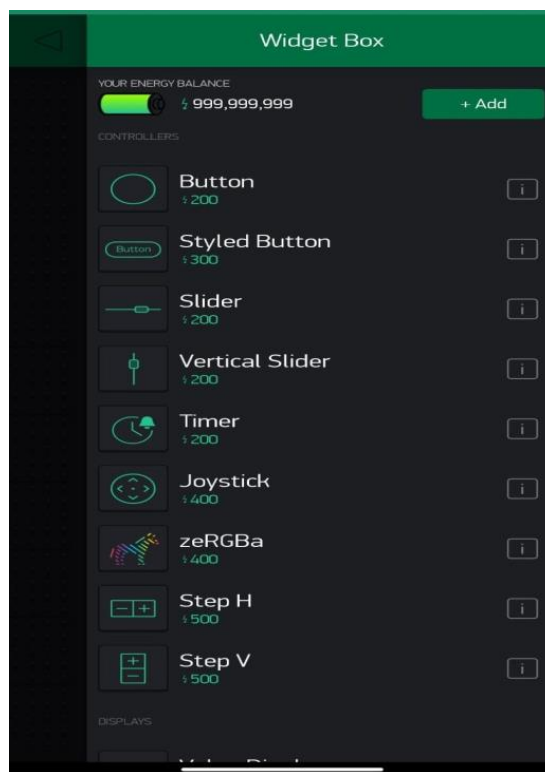
Gambar 2. 10 Tampilan Menu Aplikasi BYLNK

3. Setelah itu maka kita akan disuguhkan lembar *project* kosong yang akan siap untuk kita gunakan seperti gambar di bawah ini :



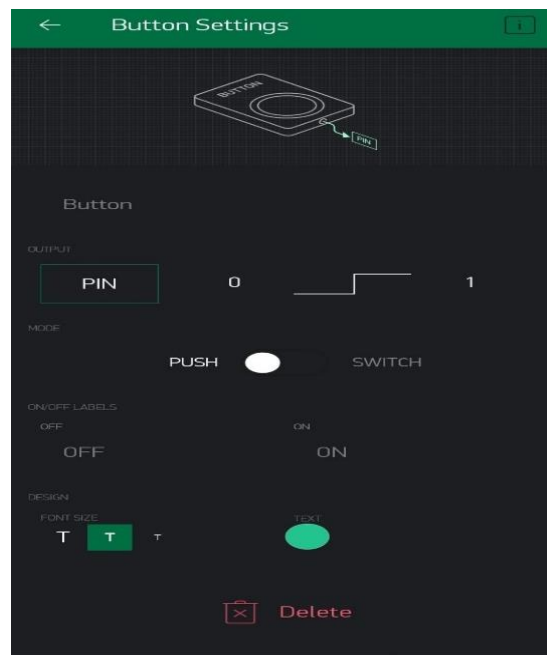
Gambar 2. 11 Lembar Project Kosong Aplikasi BYLNK

4. Jika ingin menambahkan tombol atau *widget* untuk menampilkan atau mengontrol perangkat yang terhubung maka langsung saja menekan tombol ( + ) pada bagian pojok kanan atas layar seperti berikut :



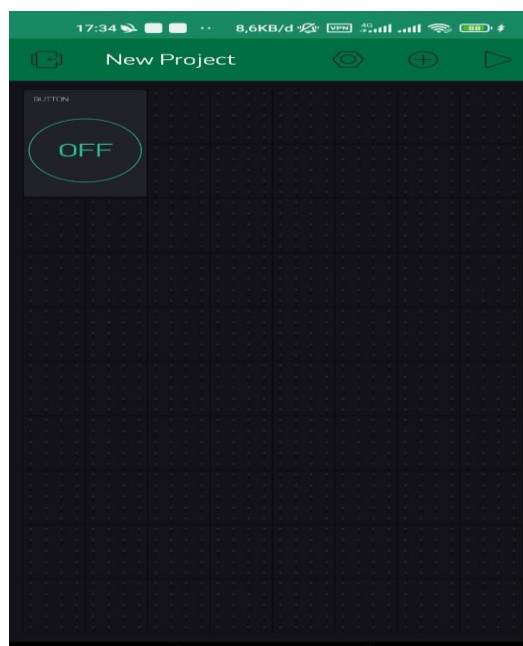
Gambar 2. 12 Menu Widget Aplikasi BYLNK

5. Jika sudah memilih *widget* yang akan dipakai kita harus mengkonfigurasi *widget* agar bisa terkontrol dan menampilkan data dari perangkat yang terhubung, konfigurasi pin disesuaikan dengan perintah yang diberikan pada kode program. Contoh konfigurasi dapat dilihat pada gambar berikut:



Gambar 2. 13 Menu Konfigurasi Aplikasi BYLNK

6. Apabila sudah terkonfigurasi sesuai dengan pin yang dipakai maka kita siap untuk menjalankan program dengan cara menekan tombol segitiga pada pojok kanan atas layar seperti gambar berikut :



Gambar 2. 14 Menu Menjalankan Program Aplikasi BYLNK

## 2.11 Arduino IDE

*Arduino Integrated Development Environment* (Arduino IDE) merupakan perangkat lunak pengolah teks program yang ditujukan untuk keluarga *arduino* dan *cloninya*. Program ini meliputi *code editor*, *compiler* dan *uploader*. Selain menuliskan kode secara manual, pada *Arduino IDE* juga telah tersedia contoh program yang bisa digunakan sebagai referensi. *Arduino IDE* cukup mudah digunakan karena menggunakan bahasa pemrograman C/C++ yang umum digunakan. Selain itu, pada editornya juga terdapat kode warna yang membedakan antara sintak, komentar dan label dengan tulisan lainnya.

Pada program ini terdapat menu bar dan *tool bar* yang memudahkan penggunanya mengakses fitur *arduino*. Beberapa *tool bar* yang umum digunakan beserta fungsinya yakni *Verify* untuk mengecek kode dari kesalahan sintak ataupun nilai. *Upload* untuk mengunggah kode ke perangkat, *New* untuk membuat *sketch* (sebutan untuk lembar kerja pada *Arduino IDE*) baru dan *Serial Monitor* untuk membuka tampilan komunikasi serial *arduino*.

```

50PERSE
#include <Wire.h>
#define BLYNK_PRINT Serial
#include <Stepper.h>
#include <SPI.h>
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <DHT.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C LCD(0x27,16,2);
Stepper my_Stepper(200, D5, D6, D7, D8);

int pump=2; //pin 2 pada nodemcupin 4
bool Right = false;
bool Left = false;

char auth[] = "ibgazmKpbF4AcywxwhuyEiMoann0lToJ"; // TOKEN PADA A

```

Done Saving

```

30 | | //lcd to android as LCD to board
| ^
exit status 1
a function-definition is not allowed here before '{' token

```

FS:2MB OTA~1019KB), 2, v2 Lower Memory, Disabled, None, Only Sketch, 115200 on COM5

Gambar 2. 15 Arduino Integrated Development Environment (Arduino IDE)

### 2.11.1 *Library* Arduino IDE

Ketika modul elektronika dihubungkan dengan *Arduino*, *Arduino* biasanya dapat mengirimkan sinyal *high*, *low*, dan membaca data analog jika modul tersebut merupakan modul *input* analog atau membaca data digital jika memang *input* datanya digital. Ini artinya dibutuhkan semacam pengenalan agar modul elektronika yang digunakan dapat dikenali dengan baik oleh *Arduino* dan berfungsi sebagaimana mestinya, maka dari itu dalam aplikasi *Arduino* IDE dikenal sebagai *Library* yang fungsinya menginisialisasi atau mengenal modul elektronika yang akan dipakai agar data dan fungsi modul tersebut dapat berjalan sebagai mestinya. Misalnya *Library* (*LiquidCrystal\_I2C.h*) berfungsi untuk membaca modul dari *driver* I2C atau misalnya *Library* (*dht.h*) berfungsi sebagai inisialisasi sensor DHT 11.