

LAPORAN
***PROJECT* SISTEM DIGITAL**
” MONITORING SUHU DAN KELEMBABAN UDARA
MENGGUNAKAN SENSOR DHT22 BERBASIS IOT
(INTERNET OF THINGS)”



KELOMPOK 9:

Mujib Chusni Mubarak	E1E122022
Farras Sida Toruntju	E1E122052
La Ode Muhammad Yudhy Prayitno	E1E122064
Rahma Damayanti	E1E122076
Fabelina Agsaria	E1E122096
Muhammad Fahreza Aryanta Umarella	E1E122116
Rama Qubra Putra	E1E122136

JURUSAN TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS HALU OLEO

KENDARI

2023

KATA PENGANTAR

Puji syukur kita panjatkan kehadiran Tuhan Yang Maha Esa, karena atas rahmat dan karunianyalah kami belum menyelesaikan laporan proyek ini dengan judul “Monitoring Suhu Dan Kelembaban Udara Menggunakan Sensor DHT 22 Berbasis Iot (*Internet of Things*)” ini tepat pada waktunya. Adapun tujuan dari penulisan laporan ini agar dapat memenuhi syarat dari tugas Sistem Digital. Selain itu, laporan ini bertujuan untuk menambah wawasan mengenai mengaplikasikan sistem digital dalam kehidupan sehari-hari bagi para pembaca dan penulis.

Kami mengucapkan terima kasih kepada Ibu Isnawaty, S.Si., M.T. selaku dosen mata kuliah Sistem Digital, serta Kakak Farid Muhammad dan Kakak Abdi Aman Bangsa selaku asisten dosen dalam penyusunan *project* ini, kami berterima kasih atas bimbingannya dalam pembuatan laporan ini sehingga bisa meningkatkan wawasan pemikiran kami tentang bidang studi yang kami tekuni, dan kami juga berterima kasih kepada rekan-rekan anggota kelompok 9 yang telah membantu dan bekerja sama dalam proses pembuatan laporan.

Dalam laporan ini kami merasa masih ada kekurangan baik dari segi penulisan dan materi yang kami paparkan maka dari itu kami sangat meminta kritik dan saran yang membangun dari para pembaca guna untuk memperbaiki pembuatan laporan selanjutnya.

Kendari, 7 Juli 2023

Penyusun

MONITORING SUHU DAN KELEMBABAN UDARA MENGUNAKAN SENSOR DHT22 BERBASIS IOT (*INTERNET OF THINGS*)

**Mujib Chusni Mubarak, Farras Sida Toruntju, L.M. Yudhy Prayitno,
Rahma Damayanti, Fabelina Agsaria, Muhammad Fahreza Aryanta U,
Rama Qubra Putra**

ABSTRAK

Permintaan terhadap otomatisasi dan sistem intelejen sangat tinggi, itu sebabnya masyarakat menunjukkan ketertarikan terhadap perangkat pintar. Masyarakat dapat mengontrol atau memonitor suatu ruangan atau laboratorium melalui *web* atau aplikasi melalui telepon genggam dengan sistem *Internet of Things* (IoT). Sistem IOT (*Internet of Things*) sangat mempermudah masyarakat untuk dapat memantau dan mengakses suhu dan kelembaban udara pada suatu ruangan atau laboratorium dimana dan kapan pun.

Dalam proyek akhir ini telah dirancang alat untuk memonitoring suhu dan kelembaban udara. Pada alat ini digunakan sensor jenis DHT22 yang berfungsi sebagai pendeteksi suhu dan kelembababan udara dan digunakan NodeMCU ESP8266 yang berfungsi sebagai pengolah data sehingga hasil deteksi dapat ditampilkan pada layar *smartphone* atau PC agar pengguna dapat membaca langsung hasil pengukuran dari suhu dan kelembaban udara pada suatu ruangan.

Hasil dari proyek ini adalah berhasilnya implementasi alat monitoring suhu dan kelembaban udara berbasis IoT menggunakan sensor DHT22 dan NodeMCU ESP8266. Alat ini mampu mengukur suhu dan kelembaban udara secara akurat dan menyajikan hasil pengukuran melalui tampilan pada *smartphone* atau PC. Dengan demikian, pengguna dapat dengan mudah memantau kondisi suhu dan kelembaban udara pada suatu ruangan atau laboratorium.

Kata kunci: Suhu, Kelembapan, NodeMcu ESP8266, DHT 22, Blynk, IoT (*Internet of Things*)

DAFTAR ISI

KATA PENGANTAR	i
ABSTRAK.....	ii
DAFTAR TABEL	vi
DAFTAR GAMBAR	vii
BAB I.....	1
PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan	2
1.4 Manfaat	3
BAB II	4
LANDASAN TEORI.....	4
2.1 Sensor DHT 22	4
2.1.1 Suhu	5
2.2.2 Kelembapan	5
2.2 Node MCU ESP8266.....	6
2.2.1 Pengenalan Node MCU ESP8266	6
2.2.2 Konfigurasi Pin NodeMCU ESP8266	7
2.3 Kabel Jumper	9
2.4 I2C LCD karakter	10
2.6 <i>Software</i> Arduino IDE	11
2.7 <i>Internet of Things</i> (IoT)	12
2.7.1 Blynk	12
BAB III	13
METODOLOGI PENELITIAN	13

3.1 Jenis Penelitian	13
3.1 Waktu dan Tempat Penelitian	13
3.1.1 Waktu	13
3.1.2 Tempat	13
3.2 Alat dan Bahan.....	13
3.2.1 Alat.....	13
3.2.2 Bahan	15
3.2 Studi Literatur	16
3.3 Prosedur Penelitian	16
BAB IV	18
PERANCANGAN DAN IMPLEMENTASI ALAT	18
4.1 Perancangan Sistem	18
4.1.1 Diagram Blok Rangkaian	18
4.1.2 <i>FLOWchart</i> System	19
4.2 Perancangan Perangkat Keras.....	21
4.2.1 Rangkaian LCD I2C	21
4.2.3 Rangkaian Sensor DHT 22	23
4.3 Perancangan Perangkat Lunak.....	25
4.3.1 Penggunaan <i>Software</i> Arduino IDE	25
4.3.2 Perencanaan Program Aplikasi <i>Smartphone</i>	26
4.2 Implementasi Alat	27
4.3.1 Pengujian Adaptor	27
4.3.2 Pengujian LCD (Liquid Crystal Display)	28
4.3.3 Pengujian Rangkaian NodeMCU	29
4.3.4 Pengukuran Rangkaian Sensor DHT 22	30
4.2.4 Pengujian Alat.....	32

BAB V	40
PENUTUP	40
5.1 Kesimpulan	40
5.2 Saran	40
DAFTAR PUSTAKA	41

DAFTAR TABEL

Tabel 3. 1 Alat yang dibutuhkan	13
Tabel 3. 2 Bahan yang dibutuhkan	15
Tabel 4. 1 <i>Library</i> yang dibutuhkan	26
Tabel 4. 2 Data Pengujian	29
Tabel 4. 3 Visualisai Data	38

DAFTAR GAMBAR

Gambar 2. 2 <i>Board</i> NodeMcu ESP8266	6
Gambar 2. 3 Susunan Pin NodeMCU ESP8266	8
Gambar 2. 4 Kabel Jumper	10
Gambar 2. 5 Modul I2C LCD 16x2 karakter	11
Gambar 2. 6 <i>Software</i> Arduino IDE	11
Gambar 4. 1 Sensor DHT 22	4
Gambar 4. 2 Diagram Blok Rangkaian	19
Gambar 4. 3 <i>FLOWchart</i> Sistem.....	20
Gambar 4. 4 Skematik Rangkaian LCD I2C.....	21
Gambar 4. 5 Skematik Koneksi LCD I2C dengan mikrokontroler NodeMCU... ..	22
Gambar 4. 6 Skematik Rangkaian Sensor DHT 22.....	23
Gambar 4. 7 Skematik Koneksi DHT 22 dengan Mikrokontroler NodeMCU	23
Gambar 4. 8 Skematik sistem monitoring suhu dan kelembapan	24
Gambar 4. 9 Pengkabelan sistem monitoring suhu dan kelembapan.....	24
Gambar 4. 10 Tampilan hasil desain di Aplikasi Blynk.....	27
Gambar 4. 11 Pengujian Adaptor	28
Gambar 4. 12 <i>Source code</i> Menampilkan Hello World ke LCD.....	29
Gambar 4. 13 <i>Source code</i> untuk Pengujian Pin Digital pada NodeMCU	30
Gambar 4. 14 <i>Source code</i> Pengujian Sensor DHT 22	31
Gambar 4. 15 Tampilan Suhu dan Kelembapan di <i>Serial Monitor</i>	31
Gambar 4. 16 Tampilan Dalam Alat.....	32
Gambar 4. 17 Tampilan Luar Alat.....	33
Gambar 4. 18 Konfigurasi Blynk	33
Gambar 4. 19 <i>Library</i> yang digunakan	33
Gambar 4. 20 Inisialisasi objek LCD	33
Gambar 4. 21 Konfigurasi WiFi dan autentikasi Blynk.....	34
Gambar 4. 22 Konfigurasi pin dan sensor DHT	34
Gambar 4. 23 Deklarasi timer Blynk dan variabel tampilan LCD.....	34
Gambar 4. 24 Fungsi deteksi_sensor().....	35
Gambar 4. 25 Fungsi send_sensor_blynk().....	35

Gambar 4. 26 Fungsi <code>setup()</code>	36
Gambar 4. 27 Fungsi <code>loop()</code>	37
Gambar 4. 28 Tampilan hasil pada LCD dan <i>Smartphone</i> (1)	37
Gambar 4. 29 Tampilan hasil pada LCD dan <i>Smartphone</i> (2)	38

BAB I

PENDAHULUAN

1.1 Latar Belakang

Saat ini, permintaan terhadap otomatisasi dan sistem intelejen sangat tinggi, itu sebabnya masyarakat menunjukkan ketertarikan terhadap perangkat pintar. Contohnya, masyarakat dapat mengontrol atau memonitor alat-alat rumah tangga mereka melalui *web* atau aplikasi melalui telepon genggam. *Internet of Things* (IoT) yang dapat membuat alat-alat atau perangkat keras tersebut dapat berkomunikasi, bertukar data, dan saling mengendalikan melalui *web* atau aplikasi telepon genggam. Suhu dan kelembaban udara di lingkungan pun dapat dimonitor melalui *web* dengan menggunakan (IoT) agar udara di lingkungan tersebut tetap sehat dan terjaga. Menurut data dari medicalogy.com kelembaban udara (*relative humidity*) adalah satuan untuk menyatakan jumlah uap air yang terkandung pada udara. Semakin banyak uap air yang dikandung dalam udara, maka semakin lembab udara tersebut.

Kelembaban udara dinyatakan dalam persen (%) dan rentang kelembaban udara dalam ruangan (*indoor*) yang dianggap ideal adalah 40%- 60% tergantung dimana Anda tinggal. Biasanya angka 45% dianggap sebagai angka yang paling ideal bagi kelembaban udara indoor. Jika kelembaban udara di ruangan tersebut rendah maka beresiko menyebabkan munculnya penyakit flu dan batuk, sedangkan jika kelembaban udara tinggi beresiko menyebabkan infeksi pernapasan yang lebih tinggi. Untuk suhu udara sendiri, suhu ideal untuk indoor adalah 20-29°C. Menurut cnnindonesia.com suhu yang berada diatas range ideal tersebut dapat meningkatkan resiko tekanan darah rendah dan memicu sakit jantung. Oleh karena itu Saya membuat suatu alat yang bisa memonitoring suhu dan kelembaban di ruangan atau rumah menggunakan sensor yang dapat langsung dipantau atau dimonitor oleh para penggunanya melalui tampilan antarmuka *web* agar mereka dapat mengetahui berapa suhu dan kelembaban di ruangan tersebut serta mengetahui apakah suhu dan kelembaban udara di ruangan tersebut aman atau tidak.

Kenyamanan merupakan bagian dari salah satu sasaran karya arsitektur, definisi kenyamanan merupakan interaksi dan reaksi manusia terhadap

lingkungan yang bebas dari rasa negative dan bersifat subjektif. Kenyamanan terdiri atas kenyamanan psikis dan kenyamanan fisik. Kenyamanan psikis yaitu kenyamanan kejiwaan (rasa, aman, tenang, gembira dan lain – lain) yang terukur secara subjektif (kualitatif). Sedangkan kenyamanan fisik dapat terukur secara obyektif (kuantitatif) yang meliputi kenyamanan spasial, visual, auditorial dan termal. bahkan istilah – istilah kenyamanan termal yang paling bias sampai yang tidak bias adalah dari gerah, nyaman, panas, dingin, sejuk dan pengap.

Dalam konteks ini, *project* yang dikerjakan mengulas hal-hal yang berkaitan dengan kenyamanan termal dan konsep untuk memperhitungkan pengaruh penggunaan machine learning dalam menganalisa data sensor yang berisi variabel suhu dan kelembaban. Sistem yang dikerjakan didasarkan pada teknologi *Internet of Things* sehingga dapat memberi kemudahan dalam implementasi perancangan sistem.

1.2 Rumusan Masalah

Berdasarkan uraian diatas, penulis tertarik untuk mengangkat permasalahan yang dihadapi masyarakat dengan memberikan sistem monitoring suhu dan kelembaban udara menggunakan sensor DHT 22 dengan sistem IoT (*Internet of Things*) dengan harapan masyarakat Indonesia mampu mengetahui keadaan suhu dan kelembaban udara pada suatu tempat melalui aplikasi ataupun *website*.

1.3 Tujuan

Adapun penulisan laporan proyek ini adalah untuk :

1. Membuat dan mengetahui cara kerja alat dan bagaimana penerapan *Internet of Things* dalam memonitoring suhu dan kelembaban udara Berbasis NodeMCU ESP8266.
2. Untuk memenuhi tugas proyek sistem digital sebagai salah satu syarat pemenuhan nilai mata kuliah Sistem Digital.

1.4 Manfaat

Berdasarkan permasalahan yang telah dirumuskan, berikut adalah manfaat yang dapat diperoleh dari penelitian ini:

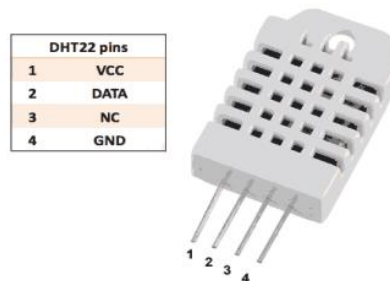
1. *Output* dari penelitian ini dapat dimanfaatkan oleh seluruh masyarakat Indonesia. Salah satunya adalah pendekatan metode data science dalam mendukung optimalisasi penggunaan sensor untuk kebutuhan kehidupan sehari-hari.
2. Metode klasifikasi kenyamanan termal dari penelitian ini berpotensi untuk diimplementasikan pada sejumlah besar data sensor dan dapat diunduh secara *real time* melalui jaringan internet (*Internet of Things*) yang belum dikelola dengan baik.

BAB II

LANDASAN TEORI

2.1 Sensor DHT 22

DHT22 adalah sensor suhu dan kelembaban, juga dikenal sebagai sensor AM2302. Sensor ini mirip dengan DHT11 dan memiliki empat pin yaitu *Power Supply*, sinyal data, NC, *Ground* [12]. Tegangan sumber biasanya 5V yang dihubungkan ke kaki Vs karena tegangan sumber yang digunakan mengikuti tegangan operasi mikrokontroler yaitu sama sebesar 5V. DHT22 menggunakan teknik pengumpulan sinyal digital eksklusif dan teknologi penginderaan kelembaban dan dapat mensuplai sinyal digital yang dikalibrasi.



Gambar 4. 1 Sensor DHT 22

Sensor yang di gunakan untuk mengetahui nilai suhu dan kelembaban adalah sensor DHT22, untuk dapat digunakan pada arduino sensor DHT22 membutuhkan beberapa komponen untuk pengkondisian tegangan yang masuk ke dalam sensor berupa resistor. Pada alat yang dibuat digunakan sebuah modul sensor DHT22 yang sudah siap digunakan pada *board* NodeMCU. Sensor DHT22 terdiri dari 2 buah sensor didalamnya yaitu sensor kelembaban yang berupa capacitivetype *humidity* untuk pengukur kelembaban sensor ini bekerja berdasarkan perubahan kapasistas kapasitor apabila ada objek yang berada dalam daerah deteksinya yaitu adanya molekulair di udara dan sebuah temperature module untuk mengatur suhu yang terbuat dibuat dari campuran bahan semikonduktor yang dapat menghasilkan hambatan interistik yang akan berubah terhadap temperature. Cara kerja dari sensor DHT22, bahan semikonduktor pada sensor suhu dan kelembaban membaca nilai suhu dna kelembaban kemudian data dikirimkan ke wemos dalam bentuk digital secara beriringan, waktu pengiriman data antara data suhu dan kelembaban sangatlah singkat yaitu

kurang dari 40ms, sehingga pembacaan akan terlihat seperti bersamaan. Untuk dapat mengirim data digital pengukuran pertama mikro kontroler.

2.1.1 Suhu

Suhu adalah besaran yang menyatakan derajat panas atau dingin suatu benda. Secara mikroskopis suhu menunjukkan energi yang dimiliki oleh suatu zat. Setiap atom dalam suatu benda selalu berada dalam keadaan bergerak, baik itu perpindahan ataupun gerakan ditempat yang berupa getaran. Semakin besar energi atom penyusun suatu benda, maka semakin besar pula suhu benda tersebut. Alat ukur yang digunakan untuk mengukur besarnya suhu adalah thermometer. Mengacu pada SI, satuan suhu adalah Kelvin (K), akan tetapi ada skala-skala lain yang digunakan selain Kelvin, yaitu Celcius, Fahrenheit dan Rankine.

Di Indonesia, skala yang umum digunakan untuk mengukur suhu adalah skala Celcius, pada skala Celcius 0°C adalah titik dimana air membeku dan 100°C adalah titik didih air pada tekanan 1 atmosfer. Skala Celcius dan Kelvin memiliki tingkatan kenaikan skala yang sama, kenaikan 1°C sama dengan kenaikan suhu 1 K, yang membedakan hanyalah titik 0 skala. Saat skala Celcius menunjukkan 0°C maka pada skala Kelvin bernilai 273 K (atau 273.15 untuk lebih tepatnya) sehingga untuk mengkonversikan nilai Celcius ke skala Kelvin hanya perlu menambahkan 273 (atau 273.15 untuk lebih tepatnya).

2.2.2 Kelembapan

Kelembaban udara relatif (atau RH, *Relative Humidity*), adalah rasio antara tekanan uap air aktual pada temperatur tertentu dengan tekanan uap air jenuh pada temperatur tersebut. Pengertian lain dari Kelembapan adalah perbandingan antara jumlah uap air yang terkandung dalam udara pada suatu waktu tertentu dengan jumlah uap air maksimal yang dapat ditampung oleh udara tersebut pada tekanan dan temperatur yang sama. Kandungan uap air dalam udara hangat lebih banyak dari kandungan uap air dalam udara dingin. Perubahan tekanan parsial uap air di udara berhubungan dengan perubahan suhu. Konsentrasi air di udara pada tingkat permukaan laut mencapai 3% pada suhu 30°C dan tidak berubah 0.5% pada 0°C . Kelembaban udara dapat

dinyatakan sebagai kelembaban udara absolut dan kelembaban dengan massa uap air atau tekanannya per satuan volume (Kg/m^3).

2.2 Node MCU ESP8266

2.2.1 Pengenalan Node MCU ESP8266

NodeMcu merupakan sebuah opensource *platform* IoT dan pengembangan Kityang menggunakan bahasa pemrograman Lua untuk membantu programmer dalam membuat *prototype* produk IoT atau bisa dengan memakai sketch dengan arduino IDE. Pengembangan Kit ini didasarkan pada modul ESP8266, yang mengintegrasikan GPIO, PWM (Pulse Width Modulation), IIC, 1-Wire dan ADC (Analog to Digital Converter) semua dalam satu *board*. Keunikan dari Nodemcu yaitu *Board*nya yang berukuran sangat kecil yaitu panjang 4.83cm, lebar 2.54cm, dan dengan berat 7 gram. Tapi walaupun ukurannya yang kecil, *board* ini sudah dilengkapi dengan fitur wifi dan firmwarena yang bersifat opensource. Penggunaan NodeMcu lebih menguntungkan dari segi biaya maupun efisiensi tempat, karena NodeMcu yang ukurannya kecil, lebih praktis dan harganya jauh lebih murah dibandingkan dengan Arduino Uno. Arduino Uno sendiri merupakan salah satu jenis mikrokontroler yang banyak diminati dan memiliki bahasa pemrograman C++ sama seperti NodeMcu, namun Arduino Uno belum memiliki modul wifi dan belum berbasis IoT. Untuk dapat menggunakan wifi Arduino Uno memerlukan perangkat tambahan berupa wifi shield. Node Mcu merupakan salah satu produk yang mendapatkan hak khusus dari Arduino untuk dapat menggunakan aplikasi Arduino sehingga bahasa pemrograman yang digunakan sama dengan *board* Arduino pada umumnya.



Gambar 2. 1 *Board* NodeMcu ESP8266

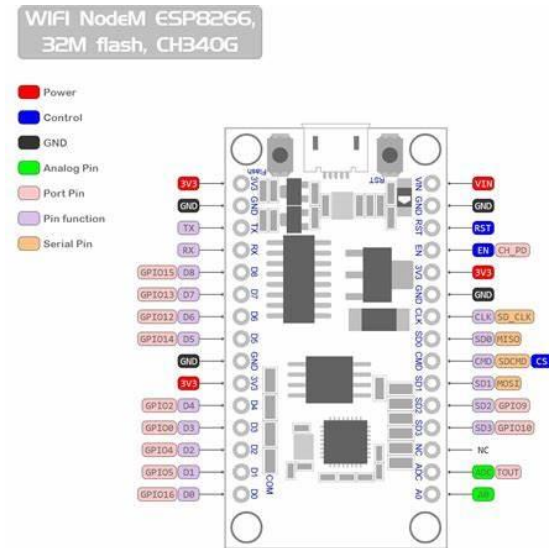
Spesifikasi yang dimiliki oleh NodeMCU sebagai berikut :

1. Berbasis ESP8266 serial WiFi SoC (Single on Chip) dengan fitur *onboard* USB to TTL.
2. Wireless yang digunakan adalah IEEE 802.11b/g/n.
3. Dilengkapi dengan 2 kapasitor tantalum, yaitu 100 mikrofarad dan 10 mikrofarad.
4. Memiliki regulator LDO 3.3V.
5. Terdapat lampu LED biru sebagai indikator.
6. Memiliki tombol reset, *port* USB, dan tombol flash.
7. Terdapat 9 pin GPIO, termasuk 3 pin PWM, 1 kanal ADC, dan pin RX dan TX.
8. Terdapat 3 pin *ground*.
9. Pin S3 dan S2 dapat digunakan sebagai pin GPIO.
10. Mendukung komunikasi SI MOSI (Master *Output* Slave Input), yaitu jalur data dari master ke slave.
11. Mendukung komunikasi SO MISO (Master Slave Input), yaitu jalur data dari slave ke master.
12. Terdapat pin SK yang berfungsi sebagai SCLK (Serial Clock) dari master ke slave.
13. Terdapat pin Vin untuk input tegangan daya (DC 4 ~ 9V).
14. Dilengkapi dengan MCU (Microcontroller Unit) berbasis 32-bit yang terintegrasi.
15. WiFi modern ESP8266 dengan kapasitas memori 32MB.
16. Dilengkapi dengan konverter serial USB-TTL CH340G dan soket mini USB.
17. Terdapat 15 baris pin dengan jarak 2.54mm

2.2.2 Konfigurasi Pin NodeMCU ESP8266

Rangkaian NodeMCU ESP8266-CH340G ini adalah sebuah otak dan sistem kendali rangkaian alat monitoring suhu dan kelembaban menggunakan *web* secara online berbasis ESP8266. Pengembangan Kit ini didasarkan pada modul ESP8266, yang mengintegrasikan GPIO, PWM (Pulse Width Modulation), IIC , 1-Wire dan ADC (Analog to Digital

Converter) semua dalam satu *board*.



Gambar 2. 2 Susunan Pin NodeMCU ESP8266

Dari gambar diatas dapat dilihat masing-masing pin NodeMCU ESP8266 sebagai berikut :

1. RST : berfungsi sebagai modul
2. ADC : Analog Digital Converter. Rentang tegangan masukan 0-1v, dengan skup nilai digital 0-1024
3. EN: Chip *Enable*, Active *HIGH*
4. IO16 : GPIO16, dapat digunakan membangunkan chipset dari mode deep sleep
5. IO14 : GPIO14; HSPI_CLK
6. IO12 : GPIO12; HSPI_MISO
7. IO13 : GPIO13; HSPI_MOSI;UART0_CTS
8. VCC: Catu daya 3,3V (VDD)
9. CSO: Chip selection
10. MISO : Slave *output*, Main input
11. IO9 :GPIO9
12. IO10 GBIO10
13. MOSI : Main *output* slave input
14. SCLK : Clock
15. GND : *Ground*

16. IO15 : GPIO15; MTDO; HSPICS; UART0_RTS
17. IO2 : GPIO2; UART1_TXD
18. IO0 : GPIO0
19. IO4 : GPIO4
20. IO5 : GPIO5
21. RXD : UART0_RXD; GPIO3
22. TXD : UART0_TXD; GPIO1

2.3 Kabel Jumper

Kabel jumper adalah sebuah jenis kabel yang digunakan dalam rangkaian elektronik untuk menghubungkan komponen atau sirkuit dengan mudah dan cepat. Kabel jumper umumnya terbuat dari bahan konduktif yang baik seperti tembaga, yang memungkinkan aliran listrik yang lancar dan minim resistansi. Ini penting untuk menjaga kualitas sinyal dan menghindari penurunan tegangan yang tidak diinginkan saat mentransfer data atau sinyal listrik.

Kabel jumper memiliki ukuran yang relatif kecil, dengan panjang yang bervariasi, mulai dari beberapa sentimeter hingga beberapa puluh sentimeter. Ukurannya yang kecil memungkinkan penggunaan yang fleksibel dan memungkinkan penghubungan yang presisi dan rapi antara komponen atau pin-pin pada papan rangkaian.

Selain itu, kabel jumper biasanya memiliki konektor pada kedua ujungnya, yang dapat berupa male (jantan) atau female (betina). Konektor male memiliki pin atau paku yang dapat dimasukkan ke dalam konektor female, sedangkan konektor female memiliki lubang atau slot untuk menerima pin pada konektor male. Konektor ini memudahkan penghubungan dan pemutusan yang cepat dan mudah tanpa perlu melakukan soldering. Kabel jumper sendiri mempunyai banyak varian yang dapat kita temui, di antaranya yaitu:

- a. Male-Female
- b. Male-Male
- c. Female-Female

Berikut merupakan gambar dari kabel jumper yang ditunjukkan oleh gambar 2.4:



Gambar 2. 3 Kabel Jumper

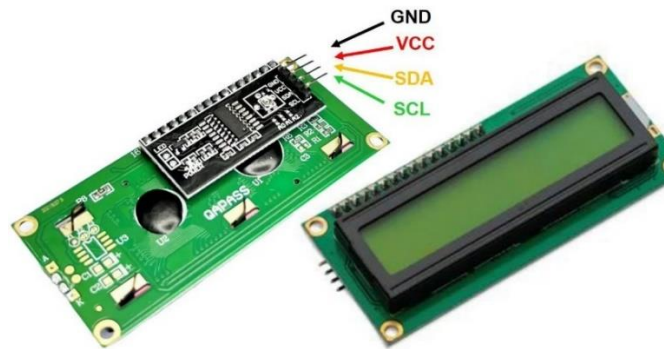
Karakteristik lain dari kabel jumper adalah fleksibilitasnya. Kabel ini dapat ditekuk atau diluruskan sesuai kebutuhan, sehingga memudahkan dalam menyesuaikan posisi dan bentuk penghubungan komponen atau sirkuit. Fleksibilitas ini memungkinkan pemaduan yang lebih baik dalam desain dan pengaturan kabel di dalam kotak rangkaian atau papan sirkuit.

2.4 I2C LCD karakter

LCD (*Liquid Crystal Display*) merupakan suatu perangkat elektronika yang terkonfigurasi dengan kristal cair dalam gelas plastik atau kaca sehingga mampu memberikan tampilan berupa titik, garis, simbol, huruf, angka maupun gambar. Dibuat dengan teknologi Complementary metal–oxide–semiconductor (CMOS) *logic* yang bekerja dengan tidak menghasilkan cahaya tetapi memantulkan cahaya yang berada di sekelilingnya. LCD terbagi menjadi dua macam berdasarkan bentuk tampilannya, yaitu Text-LCD(LCD karakter)yang menampilkan huruf atau angka dan Graphic-LCD yang menampilkan bentuk titik, garis, dan gambar.

LCD karakter berukuran 16x2 yang digunakan dalam *project* ini dapat diakses secara serial menggunakan protocol I2C. Modul LCD memiliki empat kaki yaitu VCC, GND, SDA dan SCL dengan alamat yang dapat diatur mulai 0x20 sampai 0x27. Tingkat ketajaman karakter LCD dapat diatur dengan memutar variable resistor yang disediakan pada modul dan hanya membutuhkan sumber tegangan 5V DC. LCD 2x16 merupakan LCD yang mempunyai ukuran dua baris dan enam belas baris kolom. Dalam LCD mempunyai 192 karakter

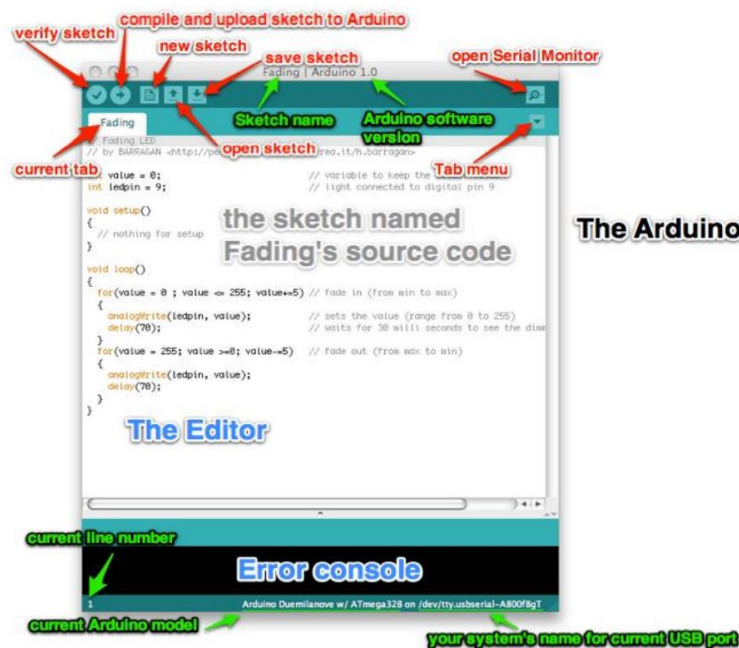
tersimpan dan dapat dialamati dengan mode 4-bit dan 8-bit. Berikut merupakan gambar LCD ukuran 2x16 yang di tunjukkan dengan gambar



Gambar 2. 4 Modul I2C LCD 16x2 karakter

2.6 Software Arduino IDE

Software arduino adalah sebuah *software* yang digunakan untuk membuat sebuah program mikrokontroler. *Software* Arduino menggunakan bahasa pemrograman sendiri yang menyerupai bahasa C. *Software* arduino ini dibuat dari bahasa pemrograman JAVA dan juga dilengkapi dengan *library* C/C++ yang biasanya disebut *Wiring* yang membuat operasi input dan *output* menjadi lebih mudah. Bukan hanya digunakan untuk memprogram mikrokontroler arduino saja tetepi *software* arduino ini juga dapat digunakan untuk memprogram mikrokontroler lainnya yang sama-sama menggunakan bahasa C/C++.



Gambar 2. 5 Software Arduino IDE

2.7 Internet of Things (IoT)

Dituliskan dalam sebuah karya ilmiah dalam McKinsey Global Institute, bahwa *Internet of Things* adalah sebuah teknologi yang memungkinkan kita untuk menghubungkan mesin, peralatan, dan benda fisik lainnya dengan sensor jaringan dan aktuator untuk memperoleh data dan mengelola kinerjanya sendiri, sehingga dimungkinkan adanya mesin untuk saling berkolaborasi dan bahkan bertindak berdasarkan informasi baru yang diperoleh secara independen. Sebuah publikasi mengenai *Internet of Things* menjelaskan bahwa *Internet of Things* adalah suatu keadaan ketika benda memiliki identitas, bisa beroperasi secara intelijen, dan bisa berkomunikasi dengan sosial, lingkungan, dan penggunanya. Tujuannya adalah untuk membuat manusia berinteraksi dengan benda lebih mudah, bahkan dengan tujuan supaya benda juga bisa berinteraksi dengan benda lainnya.

2.7.1 Blynk

Blynk adalah *platform* untuk aplikasi OS Mobile (iOS dan Android) yang bertujuan untuk kendali *module* Arduino, Raspberry Pi, ESP8266, WEMOS D1, dan module sejenisnya melalui Internet. Aplikasi ini merupakan wadah kreatifitas untuk membuat antarmuka grafis untuk proyek yang akan diimplementasikan hanya dengan metode drag and drop widget. Penggunaannya sangat mudah untuk mengatur semuanya dan dapat dikerjakan dalam waktu kurang dari 5 menit. Blynk tidak terikat pada papan atau module tertentu. Dari *platform* aplikasi inilah dapat mengontrol apapun dari jarak jauh, dimanapun kita berada dan waktu kapanpun. Dengan catatan terhubung dengan internet dengan koneksi yang stabil dan inilah yang dinamakan dengan sistem *Internet of Things* (IoT).

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 diantaranya adalah Button, Value Display, History Graph, Twitter, dan Email. Blynk tidak terikat dengan beberapa jenis microcontroller namun harus didukung *Hardware* yang dipilih. NodeMCU dikontrol dengan Internet melalui WiFi, chip ESP8266, Blynk akan dibuat online dan siap untuk *Internet of Things*.

BAB III

METODOLOGI PENELITIAN

3.1 Jenis Penelitian

Dalam pembuatan *project* ini digunakan metode deskripsif yang menggambarkan fakta-fakta dan informasi secara sistematis, faktual dan akurat. Penelitian ini dilakukan melalui internet yang dapat memberikan sumber data dan pengetahuan mengenai sistem yang diteliti, kemudian mencocokkan dengan kemungkinan yang terjadi dalam usaha penyelesaian masalah.

3.1 Waktu dan Tempat Penelitian

3.1.1 Waktu

Adapun waktu pelaksanaan praktikum Sistem Digital untuk membuat “Alat Monitoring Suhu dan Kelembapan Udara Menggunakan Sensor DHT 22 Berbasis IoT” dimulai tanggal “31 Mei 2023” sampai tanggal “13 Juli 2023”.

3.1.2 Tempat


Adapun tempat pelaksanaan praktikum Sistem Digital untuk membuat “Alat Monitoring Suhu dan Kelembapan Udara Menggunakan Sensor DHT 22 Berbasis IoT” dilaksanakan secara *offline* di Laboratorium *Artificial Intelegence*, Laboratorium *Network Engineering*, dan Laboratorium *Software Engineering*, Fakultas Teknik, Universitas Halu Oleo.


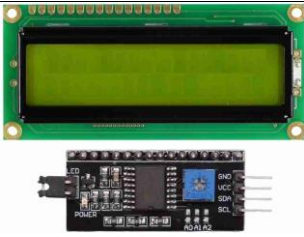


3.2 Alat dan Bahan

3.2.1 Alat

Adapun alat-alat yang dibutuhkan pada penelitian ini adalah sebagai berikut:

Tabel 3. 1 Alat yang dibutuhkan

Alat	Gambar	Fungsi
Sensor DHT 22		Untuk mengukur suhu dan kelembapan udara di sekitar lingkungan yang akan dipantau.

NodeMcu ESP8266 CH340G		sebagai mikrokontroler yang mengendalikan seluruh sistem. Mengambil data dari sensor DHT22, mengirimkan data ke server Blynk, dan mengontrol tampilan pada LCD.
LCD I2C 1602		sebagai tampilan untuk menampilkan informasi suhu dan kelembapan udara yang diambil oleh sensor DHT22 serta informasi lainnya seperti pesan pembuka dan status koneksi.
Kabel Jumper (Male to Male)		Untuk menghubungkan secara fisik antara sensor, mikrokontroler, dan LCD.
Adaptor Charger HP (Output 5V)		Menyediakan daya untuk mengoperasikan mikrokontroler dan LCD.

Kabel Data Micro USB 2.0		untuk menghubungkan mikrokontroler NodeMcu Esp8266 CH340 dengan komputer saat memprogram atau mengatur konfigurasi.
<i>Smartphone</i>		Sebagai perangkat untuk memantau suhu dan kelembapan udara melalui aplikasi Blynk.

3.2.2 Bahan

Adapun bahan yang dibutuhkan pada penelitian ini adalah sebagai berikut:

Tabel 3. 2 Bahan yang dibutuhkan

Bahan	Gambar	Fungsi
Arduino IDE		sebagai perangkat lunak untuk menulis, mengedit, dan mengunggah program ke mikrokontroler.
Blynk		sebagai perangkat lunak untuk membuat proyek dan menghubungkan mikrokontroler dengan aplikasi <i>smartphone</i> . Serta untuk memantau dan mengontrol data

		suhu dan kelembapan udara secara <i>real-time</i> .
--	--	---

3.2 Studi Literatur

Pada tahap ini dilakukan pengamatan dan studi literatur guna untuk mendapatkan informasi yang digunakan sebagai data awal untuk dasar penentuan kebutuhan perancangan sistem dan komponen untuk melakukan desain serta penelitian. Pengumpulan informasi awal ini mengenai:

1. Modul Sensor DHT 22
2. Suhu dan Kelembapan Udara

3.3 Prosedur Penelitian

Adapun tahap-tahap prosedur dalam pembuatan alat sistem monitoring suhu dan kelembapan udara menggunakan sensor DHT 22 berbasis IoT (*Internet of Things*) adalah sebagai berikut:

1. Menyiapkan mikrokontroler NodeMcu Esp 8266 CH340G, sensor DHT22, Liquid Crystal Display (LCD), dan *smartphone* dengan aplikasi Blynk terpasang
2. Menyambungkan sensor DHT22 ke pin yang sesuai pada mikrokontroler NodeMcu Esp 8266 CH340G.
3. Menghubungkan LCD dengan mikrokontroler NodeMcu Esp 8266 CH340G menggunakan koneksi I2C.
4. Mengunduh dan menginstal *library* Arduino IDE yang diperlukan.
5. Membuat proyek baru di aplikasi Blynk dan mendapatkan token otentikasi.
6. Menentukan nama WiFi dan kata sandi untuk koneksi WiFi pada mikrokontroler NodeMcu Esp 8266 CH340G.
7. Mengatur alamat LCD dan melakukan inisialisasi *port* virtual Blynk pada program.
8. Memulai LCD dengan mengatur ukuran tampilan dan mengaktifkan backlight.
9. Menampilkan pesan pembuka pada LCD untuk memberikan informasi awal kepada pengguna.
10. Memulai koneksi WiFi pada mikrokontroler NodeMcu Esp 8266 CH340G

dengan menggunakan nama dan kata sandi WiFi yang telah ditentukan.

11. Menginisialisasi koneksi Blynk menggunakan token otentikasi dari proyek Blynk.
12. Memeriksa, membaca, dan menampilkan informasi suhu dan kelembapan dari sensor DHT 22 pada LCD, serta Mengirim data suhu dan kelembapan ke server Blynk.
13. Memperbarui tampilan data suhu dan kelembapan pada LCD serta mengirim data secara berkala ke server Blynk.

Setelah tahap perancangan *prototype* telah dibuat serta pemasangan perangkat yang dibutuhkan telah terpasang, selanjutnya dilakukan pengujian, yang mana pada tahap pengujian dilakukan dalam 3 bagian yaitu, pengujian tahap pertama dimulai dari pengujian kelayakan masing-masing komponen, pengujian tahap kedua yaitu pengujian pengontrolan suhu, kelembaban, dan yang terakhir adalah pengujian sistem secara keseluruhan. Apabila terdapat masalah pada pengujian sistem yang telah dibuat maka akan dilakukan identifikasi eror dan melakukan perancangan dan perakitan ulang.

BAB IV

PERANCANGAN DAN IMPLEMENTASI ALAT

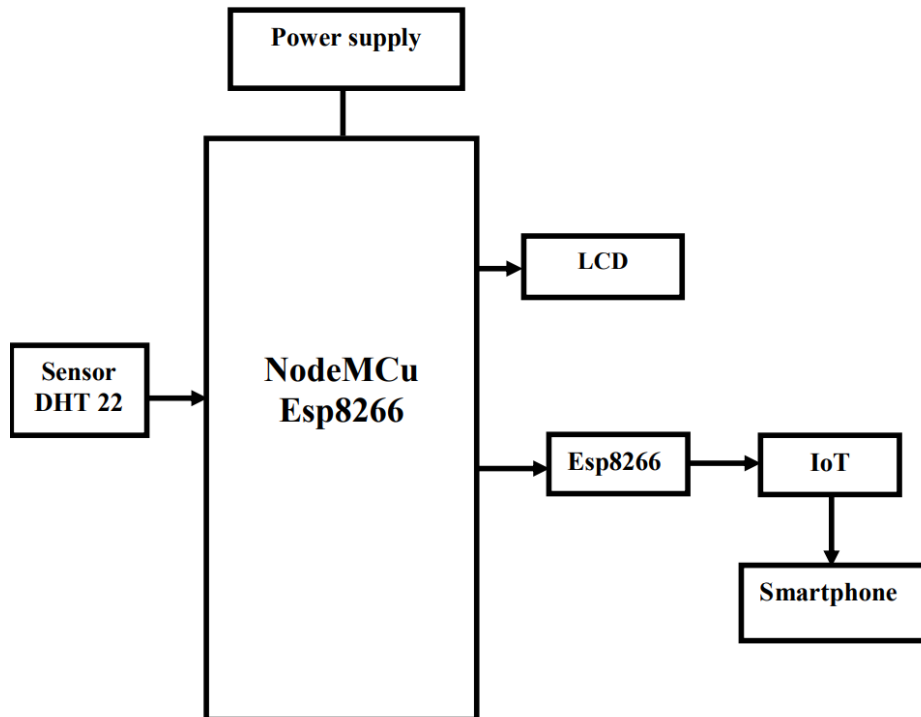
4.1 Perancangan Sistem

Perancangan sistem monitoring suhu dan kelembapan udara berbasis IoT ini menggunakan beberapa perangkat keras seperti sensor DHT 22, mikrokontrol NodeMcu Esp 8266 CH340G, Liquid Cyrstal Display (LCD), dan *smartphone*. Prinsip kerja dari sistem monitoring suhu dan kelembapan udara adalah Sensor DHT22 akan mengambil data suhu dan kelembapan udara secara periodik. Mikrokontrol NodeMcu Esp 8266 CH340 bertugas sebagai pusat pengendalian sistem. Mikrokontrol ini akan menerima data yang dikirim oleh sensor DHT22 dan memprosesnya. Selain itu, mikrokontrol juga bertanggung jawab untuk mengirimkan data yang telah diproses ke server Blynk melalui koneksi internet. LCD digunakan sebagai tampilan untuk menampilkan informasi suhu dan kelembapan udara yang diambil oleh sensor DHT22. Informasi tersebut akan ditampilkan pada layar LCD sehingga pengguna dapat dengan mudah membaca dan memantau kondisi suhu dan kelembapan udara. Pengguna dapat memantau kondisi suhu dan kelembapan udara melalui aplikasi Blynk yang terpasang di *smartphone*. Aplikasi ini terhubung dengan server Blynk dan menerima data suhu dan kelembapan udara yang dikirim oleh mikrokontrol. Pengguna dapat melihat informasi tersebut secara *real-time* dan juga menerima notifikasi jika kondisi suhu atau kelembapan udara melebihi batas yang ditentukan.

4.1.1 Diagram Blok Rangkaian

Diagram blok merupakan komponen penting dalam menjelaskan alur sistem dan membantu pembaca dalam menganalisis rangkaian suatu alat pengukur atau parameter keberhasilan sistem. Dengan menggunakan diagram blok, pembaca dapat dengan mudah memahami hubungan antara komponen-komponen dalam sistem, bagaimana sinyal atau data mengalir di antara komponen, serta fungsi masing-masing komponen dalam menjalankan tugasnya. Diagram blok memberikan gambaran visual yang jelas tentang struktur dan interaksi antara bagian-bagian sistem, sehingga memudahkan dalam pemahaman dan evaluasi terhadap keseluruhan sistem. Dengan demikian, penggunaan

diagram blok sangat berguna dalam perancangan, implementasi, dan analisis sistem yang kompleks untuk memastikan sistem berfungsi dengan efektif dan efisien.



Gambar 4. 2 Diagram Blok Rangkaian

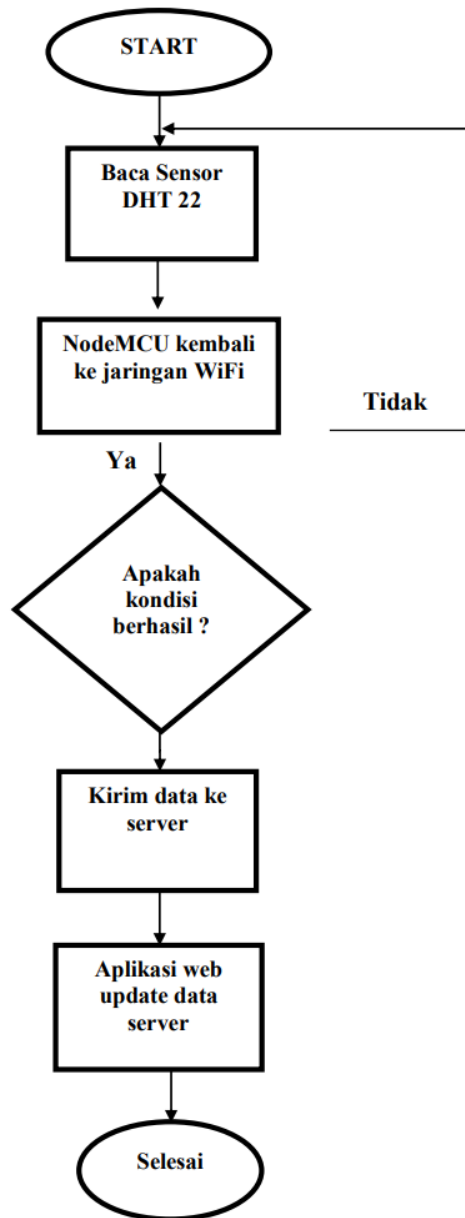
Fungsi-fungsi dari diagram blok di atas adalah sebagai berikut:

1. Blok *Power Supply* sebagai pemberi tegangan ke seluruh system
2. Blok sensor DHT22 sebagai pendeteksi Kelembaban
3. Blok Nodemcu esp8266 sebagai pengolah data dan pemberitahuan
4. Blok LCD sebagai pemberitahuan melalui tampilan layar
5. Blok esp sebagai konekt to *smartphone* via iot

4.1.2 *FLOWchart* System

FLOWchart adalah sebuah representasi grafis yang menggunakan simbol-simbol khusus untuk menggambarkan urutan proses secara rinci dan hubungan antara satu proses dengan proses lainnya dalam suatu program. *FLOWchart* memberikan gambaran visual yang jelas tentang langkah-langkah yang harus diikuti dan aliran informasi dalam suatu program. Simbol-simbol yang digunakan dalam *FLOWchart* mencakup simbol untuk proses, keputusan, input/output, loop, dan kondisi. Dengan menggunakan *FLOWchart*, pengembang perangkat lunak dapat dengan mudah memahami

dan menganalisis alur kerja program, mengidentifikasi bagian-bagian yang perlu diperbaiki atau ditingkatkan, serta melakukan *debugging* dan pengujian dengan lebih efisien. Berikut perancangan sistem *fLOWchart*. monitoring suhu dan kelembapan udara menggunakan sensor DHT 22 berbasis IoT:



Gambar 4. 3 *FLOWchart* Sistem

Keterangan Diagram Alir (*FLOWchart*):

1. Pertama-tama program dirancang untuk inisialisasi *port*, inisialisasi *port* berfungsi untuk mendefinisikan pin-pin I/O mikrokontroler yang akan digunakan dalam rangkaian.
2. Berikan perintah untuk membaca suhu dan kelembapan udara ke DHT 22.

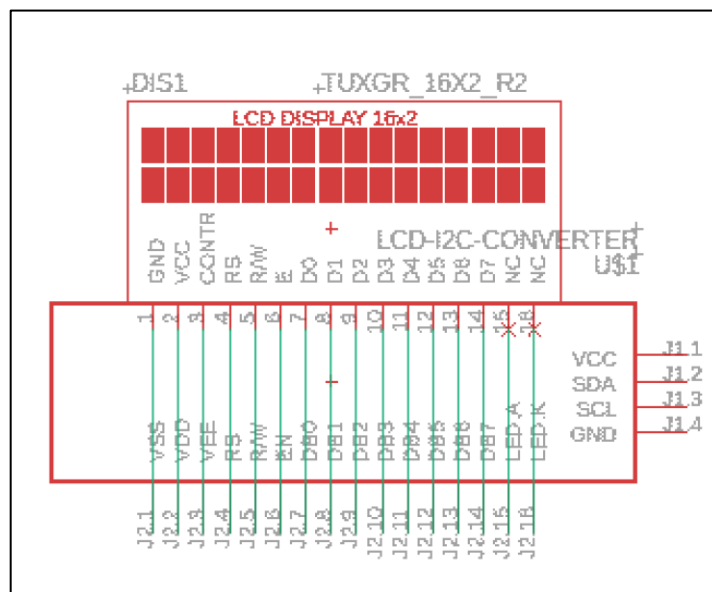
3. Di baca penginderaan suhu dan kelembaban udara yang dikirim oleh DHT 22.
4. Tampilkan hasil pengukuran ke LCD dan Android
5. Kemudian beri perlakuan dingin atau panas untuk memonitoring suhu dan kelembaban sesuai keinginan dengan keadaan baik dan sehat.

4.2 Perancangan Perangkat Keras

Pada perancangan perangkat keras (*Hardware*) menjelaskan beberapa alat yang digunakan pada penelitian ini. Adapun beberapa rangkaian perangkat keras yang digunakan sebagai berikut:

4.2.1 Rangkaian LCD I2C

Display LCD adalah sebuah display yang memberikan informasi sistem, berupa status atau data hasil olahan. Display yang digunakan adalah LCD 16x2, yaitu display 2x16 karakter. Karakter yang dapat ditampilkan LCD M16x2 adalah M16x2 adalah karakter ASCII. Display LCD memperoleh data melalui kontroler ATMEGA328. Dalam hal ini, display LCD digunakan untuk menampilkan posisi x, posisi y, dan posisi z pada suatu bidang. I2C LCD merupakan modul untuk LCD yang dikendalikan secara serial menggunakan protokol I2C atau Inter Integrated Circuit. Modul I2C LCD berfungsi untuk meringkas pin-pin pada LCD 2x16. Berikut rangkaian LCD I2C:

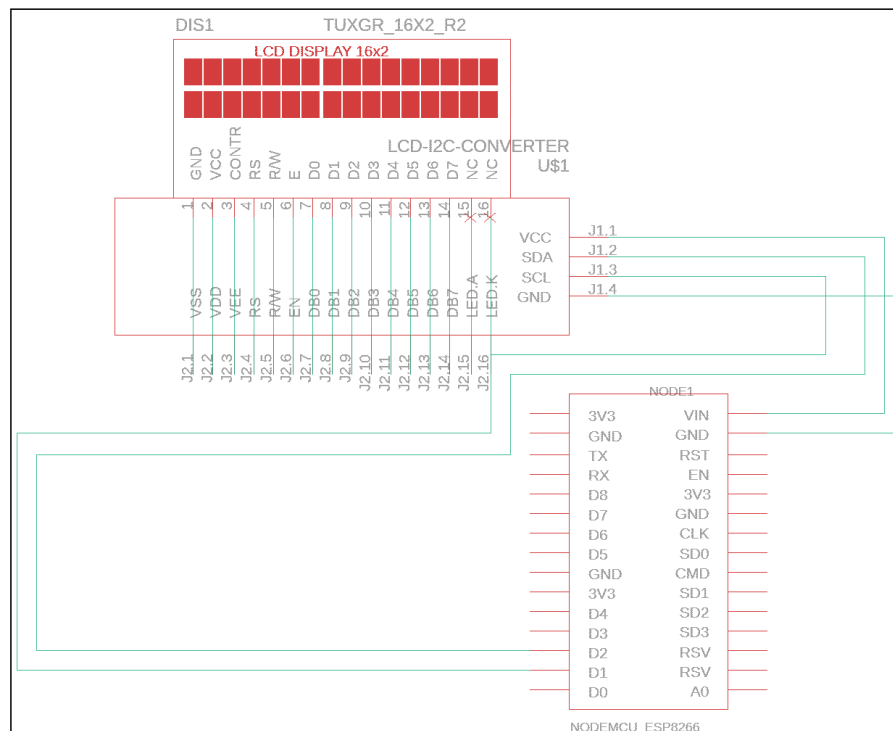


Gambar 4. 4 Skematik Rangkaian LCD I2C

Pin – pin LCD I2C dan keterangannya

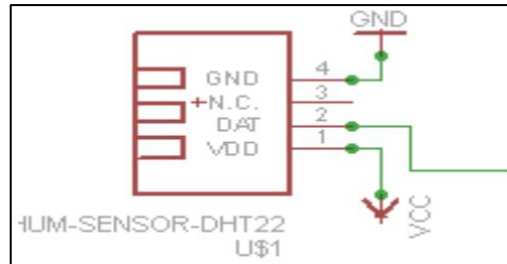
1. VCC: memberikan catu daya ke modul LCD. Biasanya terhubung ke sumber tegangan positif 5V.
2. GND: *ground* atau referensi tegangan nol.
3. SDA: *Serial Data* mentransmisikan data serial antara mikrokontroler dan modul LCD.
4. SCL: *Serial Clock* sebagai sinyal *clock* untuk sinkronisasi komunikasi antara mikrokontroler dan modul LCD.

Untuk menghubungkan modul LCD dengan antarmuka I2C (I2C LCD) ke mikrokontroler NodeMCU ESP8266, perlu menghubungkan pin SDA pada modul LCD I2C ke pin GPIO yang ditentukan pada NodeMCU, misalnya pin D2, dan menghubungkan pin SCL pada modul LCD I2C ke pin GPIO yang ditentukan pada NodeMCU, misalnya pin D1. Selain itu, menghubungkan pin VCC pada modul LCD I2C ke pin VIN pada NodeMCU untuk pasokan daya, dan menghubungkan pin GND pada modul LCD I2C ke pin GND pada NodeMCU untuk *ground*. Koneksi LCD I2C dapat dilihat dibawah ini.



Gambar 4. 5 Skematik Koneksi LCD I2C dengan mikrokontroler NodeMCU

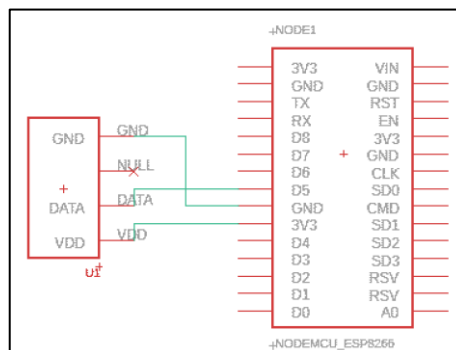
4.2.3 Rangkaian Sensor DHT 22



Gambar 4. 6 Skematik Rangkaian Sensor DHT 22

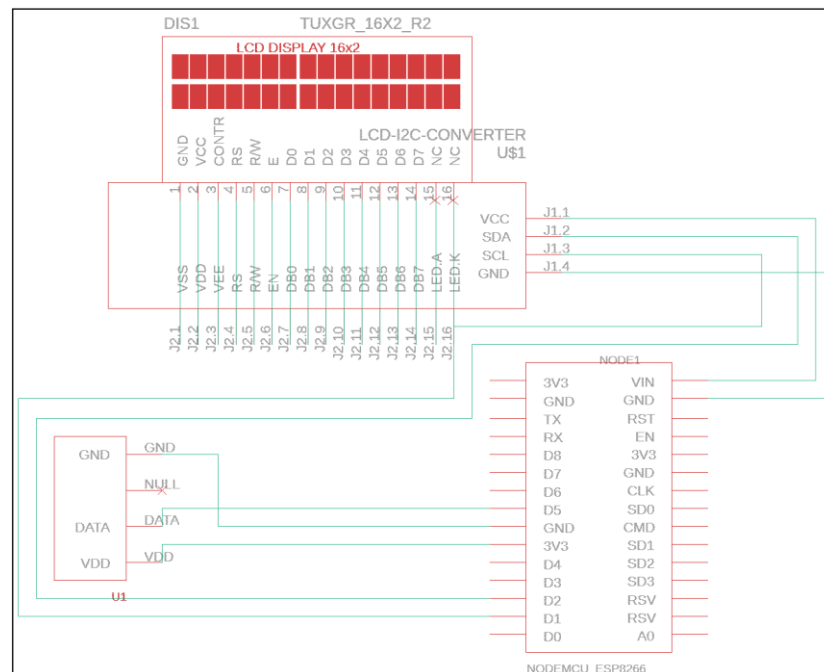
Prinsip kerja rangkaian sensor di atas akan dijelaskan Sensor Suhu dan kelembapan DHT 22 berfungsi untuk mendeteksi keberadaan suhu dan kelembapan yang berasal dari gas mudah terbakar di udara. Pada dasarnya sensor ini terdiri dari tabung aluminium yang dikelilingi oleh silikon dan di pusatnya ada elektroda yang terbuat dari aurum di mana ada element pemanasnya. Ketika terjadi proses pemanasan, kumparan akan dipanaskan sehingga SnO₂ keramik menjadi semikonduktor atau sebagai penghantar sehingga melepaskan elektron dan ketika suhu dan kelembapan dideteksi oleh sensor dan mencapai aurum elektroda maka *output* sensor DHT 22 akan menghasilkan tegangan analog.

Sensor (DHT 22) terdiri dari 4 kaki dengan spesifikasi VCC, DATA, Null dan GND. Dimana Data dihubungkan ke Nodemcu esp8266 melalui salah satu pin GPIO (pin D5), sedangkan VCC dengan tegangan 3V3, pin Null pada DHT22 tidak digunakan dan tidak perlu dihubungkan ke apa pun, dan GND DHT22 ke *ground* (GND) pada NodeMCU. Koneksi DHT 22 dapat dilihat dibawah ini.

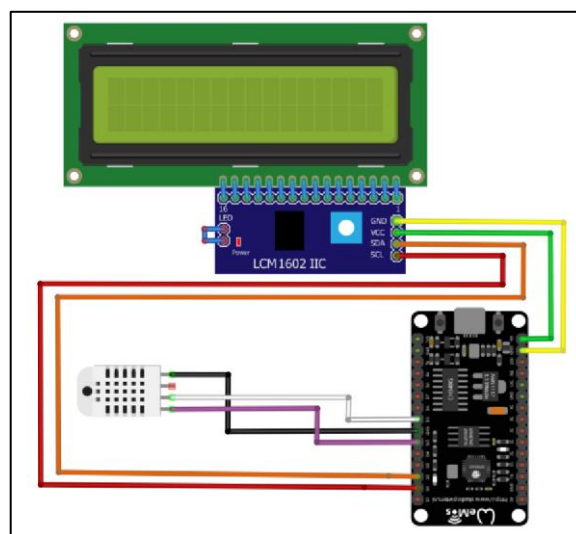


Gambar 4. 7 Skematik Koneksi DHT 22 dengan Mikrokontroller NodeMCU

Secara keseluruhan, rangkaian perangkat keras terdiri dari modul LCD I2C yang terhubung dengan mikrokontroler NodeMCU dan sensor DHT22 yang juga terhubung dengan mikrokontroler NodeMCU. Dengan koneksi yang tepat, sistem monitoring suhu dan kelembapan udara dapat bekerja dengan baik, dan informasi yang dihasilkan dapat ditampilkan pada LCD. Berikut adalah skema rangkaian keseluruhan yang menggambarkan koneksi antara LCD I2C, sensor DHT22, dan mikrokontroler NodeMCU ESP8266.



Gambar 4. 8 Skematik sistem monitoring suhu dan kelembapan



Gambar 4. 9 Pengkabelan sistem monitoring suhu dan kelembapan

4.3 Perancangan Perangkat Lunak

Perancangan perangkat lunak dimulai setelah perancangan perangkat keras dilakukan. Hal ini dilakukan karena perangkat lunak yang berfungsi untuk mengendalikan peralatan tersebut. Untuk mempermudah perancangan perangkat lunak maka terlebih dahulu dibuat algoritma dan *flowchart* untuk menggambarkan jalannya program secara keseluruhan terhadap sistem.

Perancangan perangkat lunak dilakukan untuk menyiapkan kebutuhan *software* maupun rancangan desain interface aplikasi yang akan mengontrol sistem lampu pada ruangan rumah. Perancangan perangkat lunak yang dibahas adalah pemasangan *software* IDE dan perancangan antarmuka aplikasi android menggunakan aplikasi Blynk sebagai aplikasi pengontrol pada *smartphone* yang dapat di download di Appstore ataupun Playstore secara gratis.

4.3.1 Penggunaan *Software* Arduino IDE

1. Instalasi *Software* IDE

Untuk menulis program pada *board* NodeMCU ESP8266 CH340 menggunakan *software* Arduino IDE (Integrated Development Environment). *Software* Arduino IDE dapat diunduh secara gratis dari *website* resmi Arduino di <http://arduino.cc/en/Main/Software>. Setelah mengunduh *software* tersebut, file kompresi dengan nama Arduino-1.8.2-windows.zip dapat diekstrak dan aplikasinya dijalankan dengan mengklik dua kali *arduino.exe*. Penulisan program pada NodeMCU ESP8266 CH340 menggunakan bahasa pemrograman yang didasarkan pada bahasa C/C++ yang telah disesuaikan dengan *platform* Arduino. Sintaks dan fungsi yang telah ditentukan dalam bahasa tersebut digunakan untuk mengontrol *board* NodeMCU.

Untuk menghubungkan komputer dengan *board* NodeMCU ESP8266 CH340, diperlukan kabel USB yang sesuai. Kabel USB yang umum digunakan adalah USB to Micro USB. Setelah menghubungkan *board* NodeMCU dengan komputer menggunakan kabel USB, *port* yang sesuai perlu dipilih dalam aplikasi Arduino IDE. Dalam kasus ini, *port* COM10 dipilih karena itulah yang terdeteksi oleh komputer. Selain itu, jenis *board* yang digunakan juga dipilih dalam pengaturan Arduino IDE. Untuk *board* NodeMCU ESP8266 CH340, "LOLIN(WeMos) D1 R1" dapat dipilih sebagai jenis *board* yang sesuai.

2. Penginstallan *Library*

Dalam penggunaan *software* Arduino IDE, langkah pertama yang perlu dilakukan adalah menginstal *library* yang dibutuhkan untuk mengakses perangkat keras yang digunakan. *Library* adalah kumpulan kode program yang telah ditulis sebelumnya dan dapat digunakan untuk mempermudah pengembangan aplikasi.

Untuk proyek ini, beberapa *library* yang diperlukan antara lain adalah *library* untuk komunikasi dengan sensor DHT22, *library* untuk mengontrol LCD I2C, *library* untuk mikrokontroller NodeMCU ESP8266, dan *library* untuk koneksi dalam aplikasi Blynk. Untuk menginstal *library* tersebut, langkah-langkahnya cukup sederhana. Pertama, buka *software* Arduino IDE dan pilih menu "Sketch" kemudian "Include *Library*" dan "Manage Libraries". Maka akan muncul jendela "*Library* Manager" yang berisi daftar *library* yang tersedia.

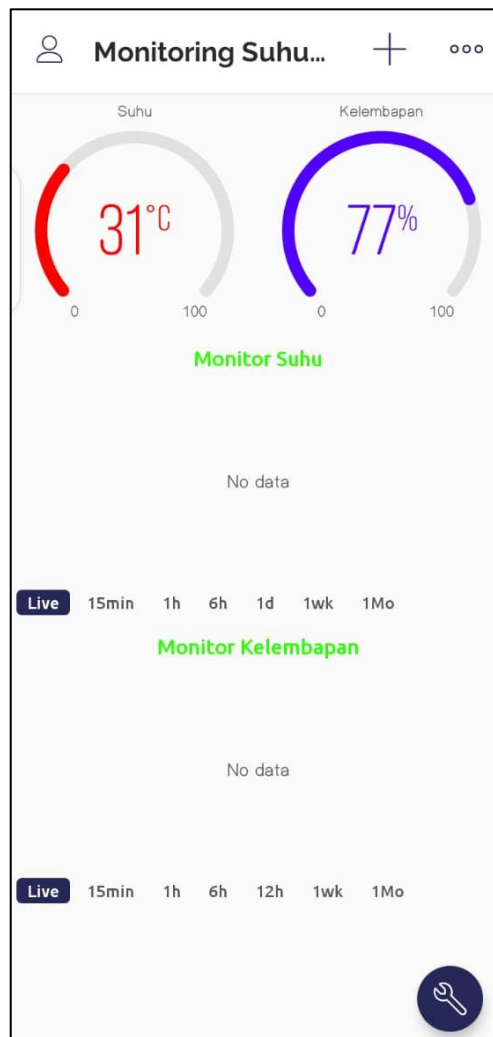
Tabel 4. 1 *Library* yang dibutuhkan

Nama <i>Library</i>	Pembuat
Blynk	Volodymyr Shymanskyy
LiquidCrystalIO	Arduino, Adafruit, theCodersCorner
LCD_ST7032	Olav Kallhovd
DHT sensor <i>library</i>	Adafruit

4.3.2 Perencanaan Program Aplikasi *Smartphone*

Pada perencanaan ini, penulis memilih menggunakan aplikasi Blynk sebagai *software* yang digunakan. Aplikasi Blynk menyediakan berbagai fitur yang dapat digunakan sesuai dengan fungsi dan program yang akan dibuat. Dalam desain perencanaan antarmuka aplikasi, penulis merancang tampilan yang *user-friendly* dan informatif. Tampilan antarmuka aplikasi Blynk terdiri dari beberapa *widget*, seperti tampilan grafik, tampilan nilai numerik, dan tampilan status. Desain ini memungkinkan pengguna untuk dengan mudah memantau data suhu dan kelembapan secara *real-time*. Selain itu, penulis juga menambahkan tombol kontrol yang memungkinkan pengguna untuk mengaktifkan atau menonaktifkan perangkat yang terhubung dengan sistem.

Gambar hasil desain antarmuka aplikasi Blynk dapat dilihat pada gambar berikut:



Gambar 4. 10 Tampilan hasil desain di Aplikasi Blynk

4.2 Implementasi Alat

4.3.1 Pengujian Adaptor

Pengujian ini bertujuan untuk mengetahui tegangan yang dihasilkan oleh adaptor cas HP dengan sumber tegangan *output* 5 volt, menggunakan multimeter digital untuk mengukur tegangan keluaran dari adaptor tersebut. Setelah dilakukan pengukuran, diperoleh hasil tegangan keluaran sebesar 5 volt. Hal ini penting untuk memastikan apakah rangkaian berfungsi dengan baik atau terdapat kesalahan. Jika saat diukur dengan multimeter, tegangan keluaran tidak mencapai 5 volt yang diharapkan, hal tersebut dapat dianggap sebagai masalah dan perlu diperiksa lebih lanjut. Pengukuran tegangan yang sesuai penting agar

daya yang disalurkan ke setiap komponen dalam rangkaian cukup, karena jika daya yang diberikan tidak mencukupi, dapat terjadi gangguan dalam kinerja alat. Gambar hasil pengukuran dapat dilihat di bawah ini:

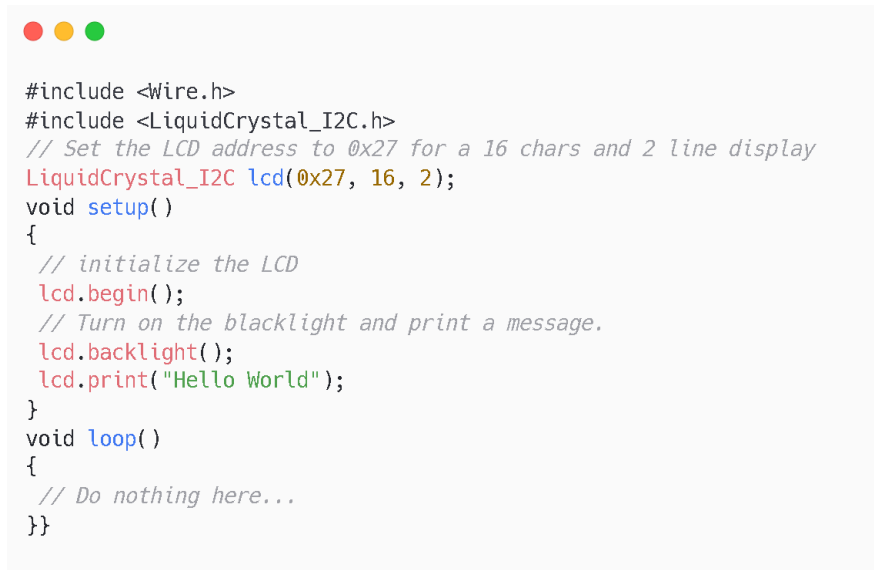


Gambar 4. 11 Pengujian Adaptor

4.3.2 Pengujian LCD (Liquid Crystal Display)

Pengujian ini bertujuan untuk memastikan apakah LCD I2C berfungsi dengan baik dalam menampilkan karakter sesuai yang diharapkan. Pengujian dilakukan dengan memprogram karakter atau tulisan yang ingin ditampilkan dan memastikan bahwa tampilan pada layar LCD I2C sesuai dengan yang diharapkan. Pada LCD I2C, pengaturan karakter ditentukan oleh pin EN, RS, dan RW. Pin EN (*Enable*) digunakan untuk memberi tahu LCD bahwa data akan dikirim. Untuk mengirim data ke LCD, logika pin EN harus diatur ke *LOW* "0", sementara pin RS dan pin RW diatur ke *HIGH* untuk mode penulisan. Pin RW adalah pin kontrol Read/Write yang digunakan untuk menentukan apakah akan menulis atau membaca data dari LCD. Namun, dalam aplikasi umum, pin RW sering diatur ke *LOW* "0".

Berdasarkan penjelasan di atas, dapat dibuat program untuk menampilkan karakter pada LCD I2C. Program tersebut diisikan ke mikrokontroler untuk mengontrol tampilan pada layar LCD I2C. Adapun program yang diisikan ke mikrokontroller untuk menampilkan karakter pada display LCD adalah sebagai berikut:



```

#include <Wire.h>
#include <LiquidCrystal_I2C.h>
// Set the LCD address to 0x27 for a 16 chars and 2 line display
LiquidCrystal_I2C lcd(0x27, 16, 2);
void setup()
{
  // initialize the LCD
  lcd.begin();
  // Turn on the backlight and print a message.
  lcd.backlight();
  lcd.print("Hello World");
}
void loop()
{
  // Do nothing here...
}

```

Gambar 4. 12 *Source code* Menampilkan Hello World ke LCD

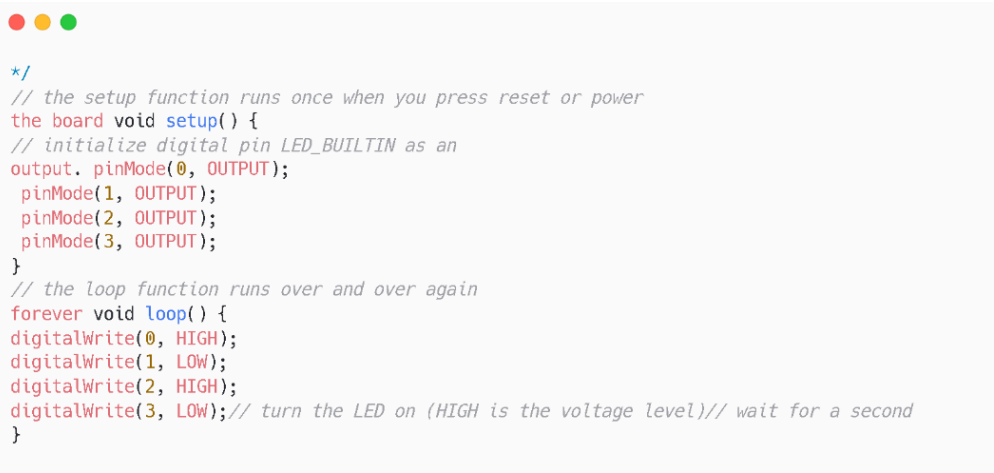
4.3.3 Pengujian Rangkaian NodeMCU

Pengujian pada rangkaian Nodemcu ini dapat dilakukan dengan menghubungkan rangkaian ini dengan multimeter sebagai pendeteksi sumber tegangan. Pada pengujian ini metode yang dipakai ialah dengan menguji melalui pin digital yang ada ada rangkaian nodemcu dengan menggunakan multimeter sebagai pendeteksi tegangan yang dihasilkan. Pada program uji arduino dimasukkan perintah “*HIGH*” dan “*LOW*” yang artinya menghidupkan dan mematikan pin yang diberi perintah “*HIGH*” dan “*LOW*” dan menghasilkan tegangan 3,3 volt pada multimeter. Pada Pin digital tegangan yang terdapat ialah sebesar 3,3 volt, maka dari itu pada saat pin digital diuji oleh multimeter maka hasil yang terdapat di multimeter haruslah sebesar 3,3 volt. Jika yang dihasilkan 3,3 volt maka Nodemcu tersebut dapat digunakan dan berfungsi dengan baik. Berikut data yang diperoleh dari hasil pengujian:

Tabel 4. 2 Data Pengujian

Pin Digital	Perintah	Tegangan (Volt)
0	<i>HIGH</i>	3.3
1	<i>LOW</i>	0.0457
2	<i>HIGH</i>	3.3
3	<i>LOW</i>	004.9

Berikut Programnya:



```
*/  
// the setup function runs once when you press reset or power  
the board void setup() {  
  // initialize digital pin LED_BUILTIN as an  
  output. pinMode(0, OUTPUT);  
  pinMode(1, OUTPUT);  
  pinMode(2, OUTPUT);  
  pinMode(3, OUTPUT);  
}  
// the loop function runs over and over again  
forever void loop() {  
  digitalWrite(0, HIGH);  
  digitalWrite(1, LOW);  
  digitalWrite(2, HIGH);  
  digitalWrite(3, LOW); // turn the LED on (HIGH is the voltage level) // wait for a second  
}
```

Gambar 4. 13 *Source code* untuk Pengujian Pin Digital pada NodeMCU

Program di atas digunakan untuk menguji dan mengontrol pin digital pada NodeMCU ESP8266. Di dalam fungsi `setup()`, pin 0, 1, 2, dan 3 diatur sebagai *output* dengan menggunakan perintah `pinMode()`. Sedangkan di dalam fungsi `loop()`, pin tersebut diatur menjadi *HIGH* dan *LOW* secara bergantian menggunakan perintah `digitalWrite()`.

Maksud dari *source code* ini adalah untuk menghidupkan dan mematikan pin digital pada NodeMCU ESP8266 secara berulang. Pada setiap iterasi loop, pin 0 dan 2 dinyalakan (*HIGH*) sedangkan pin 1 dan 3 dimatikan (*LOW*). Hal ini dapat dilihat dari perintah `digitalWrite()` yang digunakan. Dengan demikian, melalui pengukuran tegangan pada pin digital menggunakan multimeter, kita dapat memastikan bahwa tegangan yang dihasilkan adalah sebesar 3,3 volt sesuai dengan yang diharapkan.

Dengan melakukan pengujian ini dan melihat data hasil pengujian yang diperoleh, kita dapat memastikan bahwa NodeMCU tersebut berfungsi dengan baik dan menghasilkan tegangan yang sesuai.

4.3.4 Pengukuran Rangkaian Sensor DHT 22

Pengujian sensor DHT 22 dilakukan untuk menentukan suhu dan kelembaban dari beberapa sampel pengujian dilakukan dengan memanaskan objek dan memanaskannya dibawah sensor. Berikut adalah contoh *source code* yang digunakan untuk membaca suhu dan kelembapan dari sensor DHT22 dan menampilkannya di *Serial Monitor*:

```

#include <DHT.h>
#define DHTPIN D5 // Tentukan PIN tempat Anda menghubungkan sensor DHT
#define DHTTYPE DHT22 // Gunakan sensor DHT22
DHT dht(DHTPIN, DHTTYPE);
void setup() {
  Serial.begin(115200);
  dht.begin();
}
void loop() {
  float h = dht.readHumidity(); //Membaca Kelembapan
  float t = dht.readTemperature(); //Membaca Suhu dalam Celcius
  float f = dht.readTemperature(true); //Membaca Suhu dalam Fahrenheit
  if (isnan(h) || isnan(t)) {
    Serial.println("Gagal membaca sensor DHT!");
    return;
  }
  Serial.print("Suhu: ");
  Serial.print(t);
  Serial.print(" °C");
  Serial.print(" | ");
  Serial.print("Suhu: ");
  Serial.print(f);
  Serial.println(" °F");
  Serial.print("Kelembapan: ");
  Serial.print(h);
  Serial.println(" %");

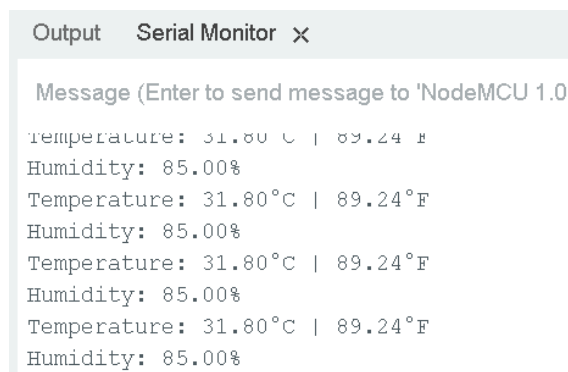
  delay(2000);
}

```

Gambar 4. 14 *Source code* Pengujian Sensor DHT 22

Dalam *source code* tersebut, kita menggunakan *library* DHT untuk membaca suhu dan kelembapan dari sensor DHT22. Kita menentukan pin tempat sensor DHT22 terhubung (D5) dan jenis sensor yang digunakan (DHT22). Pada bagian `setup()`, program menginisialisasi komunikasi Serial dengan baud rate 115200 dan memulai komunikasi dengan sensor DHT22. Selanjutnya, pada `loop()` program akan membaca suhu dan kelembapan dari sensor menggunakan fungsi `dht.readTemperature()` dan `dht.readHumidity()`. Jika pembacaan sukses, data suhu dan kelembapan akan ditampilkan di *Serial Monitor* dengan format yang sesuai. Jeda selama 2 detik (`delay(2000)`) digunakan agar hasil pembacaan dapat dilihat dengan jelas.

Dengan menggunakan *source code* di atas, kita dapat melihat data suhu dan kelembapan yang dibaca dari sensor DHT22 melalui *Serial Monitor*. Berikut tampilan di *Serial Monitor*:



The screenshot shows the 'Serial Monitor' window with the following output:

```

Temperature: 31.80 C | 89.24 F
Humidity: 85.00%
Temperature: 31.80°C | 89.24°F
Humidity: 85.00%
Temperature: 31.80°C | 89.24°F
Humidity: 85.00%
Temperature: 31.80°C | 89.24°F
Humidity: 85.00%

```

Gambar 4. 15 Tampilan Suhu dan Kelembapan di *Serial Monitor*

4.2.4 Pengujian Alat

Setelah perancangan dan pengujian komponen-komponen, kemudian kami mengimplementasikan alat ini, kami menggunakan NodeMCU ESP8266 yang terhubung dengan sensor DHT22 untuk mengukur suhu dan kelembapan. Kami juga menggunakan modul LCD I2C untuk menampilkan data suhu dan kelembapan. Alat ini dirancang untuk melakukan monitoring suhu dan kelembapan ruangan secara *real-time*.

Pada saat alat dijalankan, NodeMCU akan terhubung ke jaringan WiFi yang telah ditentukan. Kemudian, alat akan membaca data suhu dan kelembapan menggunakan sensor DHT22. Data tersebut akan ditampilkan pada layar LCD dengan menggunakan tampilan bergantian antara suhu dan kelembapan. Selain menampilkan data pada LCD, alat ini juga terintegrasi dengan *platform* Blynk. Data suhu dan kelembapan akan dikirim ke *platform* Blynk untuk dipantau melalui aplikasi Blynk di perangkat seluler. Berikut adalah foto hasil implementasi alat:



Gambar 4. 16 Tampilan Dalam Alat



Gambar 4. 17 Tampilan Luar Alat

Berikut adalah *source code* yang digunakan untuk menjalankan alat:

1. Definisi konfigurasi Blynk:

```
#define BLYNK_TEMPLATE_ID "TMPL6ARpXN06c"
#define BLYNK_DEVICE_NAME "Monitoring Suhu dan Kelembapan"
#define BLYNK_AUTH_TOKEN "mCW5XQSTYwC69KDgw30db-sd0S9PgPrS"
#define BLYNK_PRINT Serial
```

Gambar 4. 18 Konfigurasi Blynk

2. Penggunaan *Library*

```
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <DHT.h>
```

Gambar 4. 19 *Library* yang digunakan

3. Inisialisasi objek LCD

```
LiquidCrystal_I2C lcd(0x27, 16, 2); //Set alamat LCD menjadi 0x27 dengan tampilan 16 karakter dan 2 baris
```

Gambar 4. 20 Inisialisasi objek LCD

4. Konfigurasi WiFi dan autentikasi Blynk



```
char auth[] = BLYNK_AUTH_TOKEN;  
char ssid[] = "Yudhy's Network"; // Tulis nama WiFi Anda di sini  
char pass[] = "123Yudhy!456"; // Tulis kata sandi WiFi Anda di sini
```

Gambar 4. 21 Konfigurasi WiFi dan autentikasi Blynk


5. Konfigurasi pin dan sensor DHT



```
#define DHTPIN D5 // Tentukan PIN tempat Anda menghubungkan sensor DHT  
#define DHTTYPE DHT22 // Gunakan sensor DHT22  
DHT dht(DHTPIN, DHTTYPE);
```

Gambar 4. 22 Konfigurasi pin dan sensor DHT

6. Deklarasi timer Blynk dan variabel tampilan LCD



```
BlynkTimer timer;  
int displayMode = 0;  
unsigned long lastDisplayChange = 0;
```

Gambar 4. 23 Deklarasi timer Blynk dan variabel tampilan LCD

7. Fungsi `deteksi_sensor()` untuk membaca dan menampilkan data suhu dan kelembapan

```
void deteksi_sensor() {
  float h = dht.readHumidity();
  float t = dht.readTemperature();
  float f = dht.readTemperature(true);
  // Periksa kegagalan membaca sensor DHT
  if (isnan(h) || isnan(t)) {
    Serial.println("Gagal membaca sensor DHT!");
    return;
  }
  lcd.clear();
  // Tampilan mode 0: Tampilkan suhu dalam derajat Celsius dan Fahrenheit
  if (displayMode == 0) {
    lcd.setCursor(0, 0);
    lcd.print("Temperature:");
    lcd.setCursor(1, 1);
    lcd.print(t);
    lcd.print((char)223);
    lcd.print("C | ");
    lcd.print(f);
    lcd.print((char)223);
    lcd.print("F");
  }
  // Tampilan mode 1: Tampilkan kelembapan
  else if (displayMode == 1) {
    lcd.setCursor(0, 0);
    lcd.print("Humidity: ");
    lcd.setCursor(1, 1);
    lcd.print(h);
    lcd.print("%");
  }
  // Tampilkan data suhu dan kelembapan di Serial Monitor
  Serial.print("Temperature: ");
  Serial.print(t);
  Serial.print((char)223);
  Serial.print("C | ");
  Serial.print(f);
  Serial.print((char)223);
  Serial.print("F");
  Serial.print("Humidity: ");
  Serial.print(h);
  Serial.println("%");
}
```

Gambar 4. 24 Fungsi `deteksi_sensor()`

8. Fungsi `send_sensor_blynk()` untuk mengirim data suhu dan kelembapan ke *platform* Blynk

```
void send_sensor_blynk() {
  float h = dht.readHumidity();
  float t = dht.readTemperature();

  // Kirim data suhu dan kelembapan ke port virtual Blynk
  Blynk.virtualWrite(BLYNK_SENSOR_HUM, h);
  Blynk.virtualWrite(BLYNK_SENSOR_TEMP, t);

  // Log event jika suhu melebihi batas yang ditentukan
  if (t > 35) {
    Blynk.email("mr.yudhyt@gmail.com", "Alert", "Temperature over 35C!");
    Blynk.logEvent("temp_alert", "Suhu Melebihi Batas Yang Ditentukan");
  }
}
```

Gambar 4. 25 Fungsi `send_sensor_blynk()`

9. Fungsi setup() untuk inisialisasi LCD, koneksi WiFi, dan Blynk



```
void setup() {  
  lcd.begin(16, 2);  
  lcd.init();  
  lcd.backlight();  
  
  lcd.setCursor(0, 0);  
  lcd.print("KELOMPOK 9");  
  lcd.setCursor(0, 1);  
  lcd.print("SUHU RUANGAN 1");  
  delay(3000);  
  lcd.clear();  
  
  Serial.begin(115200);  
  delay(100);  
  lcd.setCursor(0, 0);  
  lcd.print("Menghubungkan ke");  
  lcd.setCursor(0, 1);  
  lcd.print(ssid);  
  delay(2000);  
  lcd.clear();  
  
  WiFi.begin(ssid, pass);  
  while (WiFi.status() != WL_CONNECTED) {  
    delay(500);  
    Serial.print(".");  
  }  
  lcd.setCursor(0, 0);  
  lcd.print("WiFi terhubung..");  
  delay(3000);  
  lcd.clear();  
  
  lcd.setCursor(0, 0);  
  lcd.print("Mulai ....");  
  delay(2000);  
  lcd.clear();  
  
  Blynk.begin(auth, ssid, pass);  
  dht.begin();  
  timer.setInterval(1000L, deteksi_sensor);  
  timer.setInterval(1000L, send_sensor_blynk);  
}
```

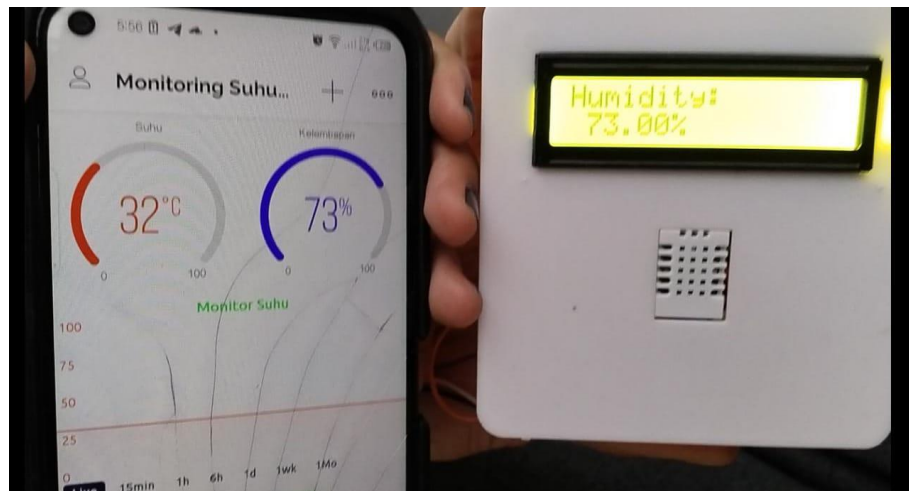
Gambar 4. 26 Fungsi setup()

10. Fungsi loop() untuk menjalankan Blynk dan timer

```
void loop() {  
  Blynk.run();  
  timer.run();  
  
  if (millis() - lastDisplayChange >= 5000) {  
    displayMode = (displayMode + 1) % 2;  
    lastDisplayChange = millis();  
  }  
}
```

Gambar 4. 27 Fungsi loop()

Setelah berhasil menjalankan rangkaian dengan source kode di atas, hasil pengukuran suhu dan kelembapan akan ditampilkan secara simultan baik pada LCD maupun pada *smartphone* melalui aplikasi Blynk. Dengan demikian, kita dapat dengan mudah memantau kondisi suhu dan kelembapan ruangan melalui perangkat yang dipilih, baik itu melalui tampilan fisik di LCD atau melalui aplikasi di *smartphone*. Berikut hasilnya pada tampilan LCD dan *smartphone*:



Gambar 4. 28 Tampilan hasil pada LCD dan *Smartphone* (1)



Gambar 4. 29 Tampilan hasil pada LCD dan *Smartphone* (2)

Selanjutnya, untuk memastikan keakuratan data suhu di beberapa ruangan, dilakukan validasi data dengan melakukan pengukuran suhu dan kelembapan pada ruangan-ruangan yang berbeda. Pengukuran dilakukan menggunakan alat yang sama dengan mengacu pada standar suhu dan kelembapan yang telah ditetapkan. Hasil pengukuran tersebut kemudian dicatat dalam tabel berikut:

Tabel 4. 3 Visualisai Data

No.	Suhu (°C)	Kelembapan (%)	Tempat
1	29.54	50	Lab RPL
2	30.80	90	Kamar Tidur
3	32.30	73	Lab Jaringan
4	28.5	78	Lab AI

Dari evaluasi hasil kerja alat dapat diambil beberapa kegunaan dari tugas *project* ini adalah untuk membantu dalam memonitoring suatu ruangan atau laboratorium agar tetap terjaga dan terkondisi suhu dan kelembabannya dengan baik. Piranti elektronik yang dibutuhkan dalam rangkaian sistem monitoring suhu dan kelembaban udara ini adalah NodeMCU ESP8266, Sensor DHT22, *Power Supply*. Alat ini dibuat dengan merangkai piranti-piranti elektronik yang menjadi suatu sistem yang dapat mengatur suhu dan mengukur kelembaban udara pada suatu ruangan khusus yang menjadi inti digunakan penghuni rumah dan akan menampilkannya secara otomatis pada

LCD dan Android dengan memberi perlakuan dingin (misalnya kipas atau Ac) atau panas (mematikan Ac dan Kipas atau menambah api unggun dengan tempat yang khusus untuk menghasilkan panas) sesuai kenyamanan kita.

Manfaat alat ini adalah untuk:

1. Mempermudah dalam memonitoring suhu dan kelembaban suatu ruangan atau laboratorium dengan cara mengakses aplikasi Blynk pada tampilan *smartphone* yang telah dibuat, dapat diakses dimanapun dan kapanpun tanpa ada batas jarak.
2. Sistem ini dapat melakukan monitoring suhu dan kelembaban dalam ruangan yang dapat membantu penghuni rumah supaya dapat menyesuaikan dengan tingkat suhu dan kelembaban yang sehat.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan evaluasi hasil kerja alat dalam tugas akhir ini, dapat disimpulkan bahwa sistem monitoring suhu dan kelembapan udara berbasis NodeMCU ESP8266 dan Sensor DHT22 dapat digunakan secara efektif untuk memantau dan mengatur kondisi suhu dan kelembapan pada ruangan atau laboratorium. Pengujian alat menunjukkan bahwa hasil pengukuran suhu dan kelembapan pada berbagai ruangan memiliki tingkat keakuratan yang baik jika dibandingkan dengan standar kalibrasi. Hal ini menunjukkan bahwa alat ini dapat memberikan informasi yang akurat mengenai kondisi suhu dan kelembapan udara di lingkungan yang diukur.

Selain itu, penggunaan piranti elektronik seperti NodeMCU ESP8266, Sensor DHT22, dan *Power Supply* pada rangkaian sistem telah membuktikan kehandalannya dalam mengukur suhu dan kelembapan serta mengatur perangkat untuk menjaga keseimbangan suhu dan kelembapan. Melalui tampilan LCD dan aplikasi Android, informasi suhu dan kelembapan dapat dengan mudah dipantau dan dianalisis.

5.2 Saran

Untuk perbaikan dan pengembangan lebih lanjut, beberapa saran yang dapat diberikan adalah sebagai berikut:

1. Perlu dilakukan rancangan yang lebih teliti pada alat untuk memastikan kinerja yang lebih optimal. Hal ini meliputi perencanaan yang lebih detail terkait pengaturan komponen dan pemilihan parameter yang sesuai.
2. Dalam pengembangan selanjutnya, dapat dilakukan peningkatan kemampuan alat dengan menggabungkannya dengan komponen lain yang relevan. Misalnya, integrasi dengan sistem kontrol otomatis untuk mengoptimalkan pengaturan suhu dan kelembapan berdasarkan kebutuhan spesifik.
3. Penting untuk melakukan penelitian lebih lanjut terkait karakteristik sensor dan metode kalibrasi yang lebih cermat. Ini akan membantu meningkatkan akurasi dan keandalan pengukuran suhu dan kelembapan.

DAFTAR PUSTAKA

Biswas, A., Nandy, K. and Chakrabarty, R., 2018. Unlimited Range Surveillance Robot Controlled by Android Application through the Internet.

Fezari, M. and Al Dahoud, A., 2018. Integrated development environment “IDE” for Arduino. *WSN applications*, pp.1-12.

Ghosh, K.U.N.D.A.N., Bhowmick, M. and Joddar, D., 2018. Globally controlled multiple relays using NODE MCU.

Hasibuan, A., Qodri, A. and Isa, M., 2021. Temperature monitoring system using arduino uno and *smartphone* application. *Bulletin of Computer Science and Electrical Engineering*, 2(2), pp.46-55.

NURUL HIDAYATI LUSITA DEWI, N.H.L.D., 2019. *Prototype smart home dengan modul nodemcu esp8266 berbasis Internet of Things (iot)* (Doctoral dissertation, UNIVERSITAS ISLAM MAJAPAHIT MOJOKERTO).

Raharjo, E.B., Marwanto, S. and Romadhona, A., 2019. Rancangan Sistem Monitoring Suhu Dan Kelembapan Ruang Server Berbasis *Internet of Things*. *Teknika*, 6(2), pp.61-68.

Randis, R. and Sarminto, S., 2018. Aplikasi *Internet of Things* Monitoring Suhu Engine Untuk Mencegah Terjadinya Over Heat. *Turbo: Jurnal Program Studi Teknik Mesin*, 7(2).

Safii, M. and Indrayani, N., 2020. Perancangan piranti lunak responsive untuk monitoring ruangan server menggunakan nodemcu esp8266 berbasis *Internet of Things*. *Jurnal Ilmiah Matrik*, 22(3), pp.270-277.

Satya, T.P., Oktiawati, U.Y., Fahrurrozi, I. and Prisyanti, H., 2020. Analisis Akurasi Sistem sensor DHT22 berbasis Arduino terhadap Thermohygrometer Standar. *Jurnal Fisika dan aplikasinya*, 16(1), pp.40-45.

Selao, A. and Hidayat, T., 2022. *Prototype* Peternakan Ayam Broiler Berbasis *Internet of Things*. *Jurnal Sintaks Logika*, 2(1), pp.287-295.