

BAB III

TINJAUAN PUSTAKA

3.1 Tinjauan Penelitian

Penelitian sebelumnya didapatkan dari hasil studi literatur yang mempunyai kesamaan atau berhubungan dengan penelitian yang sedang dilakukan. Literatur sejenis yang didapatkan berupa jurnal, dan penulisan skripsi yang kemudian dipahami dan dibuat perbandingan sehingga penelitian ini dapat menjadi pelengkap atau penyempurnaan dari penelitian-penelitian yang sudah dilakukan sebelumnya. Berikut adalah berbagai jurnal yang digunakan sebagai tinjauan studi :

Tabel 3.1 Studi Pustaka Terkait

Tahun	Judul	Hasil
2020	Qory Hidayat, Fathur Zaini Rachman, Muhammad Arif Surya Rimbawan “SISTEM MONITORING KUALITAS UDARA BERBASIS <i>FUZZY LOGIC</i> ”	Sistem Monitoring Kualitas udara ini menggunakan Arduino UNO R3 sebagai otak dari sistem dengan memanfaatkan sensor debu (GP2Y1010AU0F), sensor co (MQ-7), dan sensor co2 (MQ-135) sebagai input masukkan pembacaan kualitas udara yang digabungkan dengan logika <i>Fuzzy</i> metode sugeno guna mengoptimalkan kerja sensor agar data yang didapat menjadi satuan <i>Fuzzy</i> , sehingga alat ini dapat mengklasifikasikan tingkat kualitas udara dan dapat menampilkan kondisi udara dari tingkat baik, sedang, tidak sehat, sangat tidak sehat dan berbahaya yang akan ditampilkan pada Dot Matrix P10 (Hidayati et al., 2020).

2023	Dalila Husna Yunardi, Alim Misbullah, dan Gilang Gemilang “RANCANG BANGUN SISTEM WEB MONITORING KUALITAS UDARA DI DALAM RUANGAN MENGGUNAKAN LOGIKA FUZZY”	Penelitian ini bertujuan untuk mengembangkan sistem berbasis web yang dapat menampilkan informasi kualitas udara di dalam ruangan berupa suhu, kelembapan, dan kadar <i>Volatile Organic Compound</i> (VOC); yang merupakan indeks penilaian kualitas udara dalam ruangan. Sensor digunakan untuk mendeteksi suhu, kelembapan dan VOC, sedangkan mikrokontroler digunakan untuk mengirimkan data dari sensor ke database secara <i>real time</i> (Yunardi et al., 2023).
2021	Diemas Ibnu Pasedja dan Mohammad Yanuar Hariyawan “SISTEM MONITORING KUALITAS UDARA BERBASIS NODEMCU ESP8266 DENGAN METODE FUZZY LOGIC (SOFTWARE)”	Sistem ini dirancang menggunakan NodeMCU ESP8266 sebagai Master Control Unit (MCU) serta menggunakan beberapa sensor, diantaranya GP2Y1010AU0F sebagai pendeteksi kadar PM10 udara, sensor DHT11 sebagai pendeteksi suhu dan kelembapan, sensor MQ-7 sebagai pendeteksi kadar karbon monoksida di udara. Data sensor dikirimkan pada database dan ditampilkan pada halaman <i>website</i> lokal (<i>localhost</i>) secara <i>realtime</i> (Pasedja & Hariyawan, 2021).
2019	Rommy Zohara Shoma “PENERAPAN	Penelitian ini mengajukan perancangan simulasi sistem kontrol

	LOGIKA <i>FUZZY</i> UNTUK PENGENDALIAN KUALITAS UDARA PADA RUANGAN <i>SMOKING AREA</i> DENGAN MIKROKONTROLER”	suhu dan dan asap beserta implementasinya berupa prototype sistem pengendalian kualitas udara pada smoking room menggunakan mikrokontroller arduino. Sistem pengendalian kualitas udara pada smoking room ini dilengkapi dengan kemampuan pengatur kecepatan kipas yang sudah diprogram pada mikrokontroler berdasarkan kaidah logika <i>Fuzzy</i> yang diterapkan (Zohara Shoma, 2020).
2021	M Arfan Ravy W P, Yosep Agus Pranoto, Mira Orisa “SISTEM MONITORING SUHU DAN KELEMBAPAN RUANGAN PASIEN ISOLASI COVID-19 BERBASIS IOT”	Penelitian ini menerapkan <i>Fuzzy logic</i> metode Mamdani pada sistem monitoring suhu dan kelembapan ruangan pasien isolasi Covid-19 berbasis IOT (<i>Internet of Things</i>). Dari pengujian metode <i>Fuzzy Logic</i> tersebut diambil 10 data yang dibandingkan nilainya dengan Matlab, dapat dihasilkan rata-rata <i>error output</i> PWM sebesar 0.1%. Berdasarkan pengujian metode <i>black box</i> yang dilakukan dapat diketahui tampilan sistem sesuai dengan tampilan yang diharapkan, dengan persentase kesesuaian 100% (Arfan Ravy Wahyu Pratama et al., 2021).

Dalam penelitian ini, saya merujuk pada berbagai sumber penelitian yang tercantum dalam Tabel 3.1 untuk menyusun sistem monitoring kualitas udara dan suhu. Hasil penelitian sebelumnya menjadi acuan penting dalam melakukan penelitian yang lebih mendalam dengan tujuan mengembangkan sebuah sistem Monitoring Kualitas Udara dan Suhu di Ruang Penyimpanan Obat dengan Metode *Fuzzy Logic* berbasis IOT. Sistem ini menggunakan mikrokontroler NodeMCU ESP8266, dan berbagai sensor seperti sensor gas MQ-135, dan sensor suhu DHT11, serta LCD 16x2 I2C sebagai alat pendukung untuk memonitor dan menganalisis kualitas udara, suhu, dan kelembaban dalam ruangan penyimpanan obat. Metode yang akan digunakan dalam penelitian ini adalah *Fuzzy* mamdani untuk menentukan kondisi udara, suhu, dan kelembaban dalam ruangan penyimpanan obat secara akurat. Untuk mengimplementasikan sistem ini, saya memanfaatkan platform Blynk sebagai antarmuka data yang dapat diakses secara *real-time* melalui internet dan perangkat Android.

3.2 Landasan Teori

3.2.1. Monitoring

Monitoring adalah proses pengumpulan informasi secara berkelanjutan yang bertujuan untuk dapat mengawasi kegiatan yang telah dilakukan guna meningkatkan penyempurnaan tujuan yang ingin di capai (Putri & Zakaria, 2023).

monitoring juga didefinisikan sebagai pemantauan yang dapat dijelaskan sebagai kesadaran (*awareness*) tentang apa yang ingin diketahui, pemantauan berkadar tingkat tinggi dilakukan agar dapat membuat pengukuran melalui waktu yang menunjukkan pergerakan kearah tujuan atau menjauh dari itu. Sebuah sistem monitoring melakukan proses pengumpulan data mengenai dirinya sendiri dan melakukan analisis terhadap data-data tersebut dengan tujuan untuk memaksimalkan seluruh sumber daya yang dimiliki. Data yang dikumpulkan pada umumnya merupakan data yang *real-time*, baik data yang diperoleh dari sistem yang *hard real-time* maupun sistem yang *soft real-time* (Rahmat et al., 2022).

3.2.2. Konsep Penyimpanan

3.2.2.1. Definisi Penyimpanan Obat

Penyimpanan obat merupakan kegiatan menyimpan dan memelihara obat pada tempat yang aman dari pencurian dan juga tempat yang bisa menjaga yang didistribusikan (Elani, 2021). Sebelum didistribusikan perlu dilakukan penyimpanan yang dapat memastikan kualitas dan keamanan sediaan farmasi, alat kesehatan, dan bahan medis habis pakai agar sesuai dengan persyaratan kefarmasian seperti stabilitas dan keamanan, kelembaban, sanitasi, ventilasi, cahaya, dan penggolongan jenis pada sediaan farmasi, alat kesehatan ataupun bahan medis habis pakai.

3.2.2.2. Tujuan Penyimpanan Obat

Tujuan penyimpanan obat yaitu (Munawaroh, 2020):

1. Menjaga kualitas obat agar tidak terjadi kerusakan karena buruknya penyimpanan
2. Memudahkan saat pencarian dan pengawasan di gudang/ kamar penyimpanan
3. Mencegah penggunaan yang tidak bertanggung jawab
4. Menjaga ketersediaan stok obat

3.2.2.3. Faktor yang Mempengaruhi Penyimpanan Obat

Terdapat beberapa faktor yang mempengaruhi penyimpanan suatu obat, diantaranya:

a Suhu

Berdasarkan (Farmakope Indonesia Edisi VI, 2020), suhu penyimpanan obat diantaranya yaitu lemari pembeku, dingin, sejuk, suhu ruang dingin terkendali, suhu ruang, suhu ruang terkendali, hangat dan panas berlebih, perlindungan dari pembekuan, tempat kering.

1. Lemari pembeku: Ruangan penyimpanan dengan suhu dipertahankan secara termostatik antara -25° dan -10°

2. Dingin: termasuk dingin pada suhu kurang dari 8 ° C, suhu ini berasal dari dalam lemari pendingin dengan rentang 2 ° sampai dengan 8 ° C.
3. Sejuk: termasuk sejuk pada rentang suhu 8 ° dan 15 ° C, jika tidak disebutkan, obat yang perlu disimpan pada suhu sejuk dapat disimpan disuhu dingin
4. Suhu ruang dingin terkendali: suhu yang dipertahankan secara termostatik antara 2° dan 8° dan berdasarkan pengalaman penyimpangan antara 0° dan 15°. Lonjakan suhu hingga 25° diperbolehkan jika terdapat keterangan demikian dan tidak boleh lebih dari 24 jam kecuali didukung oleh data stabilitas atau produsen menyarankan.
5. Suhu ruang: suhu pada ruang kerja tidak lebih dari 30°C
6. Suhu ruang terkendali: Suhu yang dipertahankan secara termostatik antara 20° dan 25°, dengan toleransi penyimpangan antara 15° dan 30° dengan rata-rata suhu kinetik tidak $\geq 25^\circ$. Boleh pada lonjakan suhu 40 ° C. tetapi tidak lebih dari 24 jam.
7. Hangat: disebut hangat jika suhu pada rentang 30° dan 40°C.
8. Panas berlebih jika suhu lebih dari 40° C
9. Perlindungan dari pembekuan: Biasanya terdapat keterangan pada etiket bahwa zat harus terhindar dari pembekuan agar tidak terjadi kerusakan isi.
10. Tempat kering: Tempat dengan kelembaban tidak lebih dari 40% pada suhu ruang terkendali.

b Kelembaban

Penyimpanan obat atau produk farmasi merupakan hal yang perlu diperhatikan untuk menjaga kualitas atau mutu produk agar memenuhi syarat. Salah satu faktor dalam penyimpanan yang perlu dikhawatirkan yaitu kelembaban, nilai kelembaban ruangan yang baik yaitu 50-70% (Elani, 2021).

c Kualitas Udara

Kualitas udara dapat mempengaruhi kualitas obat dengan berbagai cara. Udara yang terkontaminasi atau memiliki kandungan zat-zat berbahaya dapat merusak obat dan mengurangi efektivitas serta stabilitasnya.

Udara yang mengandung partikel-partikel polutan seperti debu, asap, atau partikel berbahaya lainnya dapat menempel pada obat-obatan dan menyebabkan kontaminasi fisik. Partikel-partikel ini dapat merusak kualitas obat dan dapat mengganggu daya larut dan serapan obat di dalam tubuh.

Kandungan karbon dioksida (CO₂) yang tinggi dalam udara dapat menandakan masalah dengan ventilasi dan sirkulasi udara, yang dapat mempengaruhi kualitas udara di dalam ruangan penyimpanan obat.

Terdapat 3 kategori untuk nilai ppm CO₂ yaitu (Hareendran, 2021) :

- Baik untuk kesehatan : 400 ppm - 750 ppm
- Cukup (Hati-hati) : 750 ppm - 1200 ppm
- Bahaya untuk Kesehatan : >1200 ppm

3.2.3. Internet of Things

Internet of Things atau dikenal juga dengan singkatan IoT, merupakan sebuah konsep yang memiliki tujuan untuk memperlebar manfaat dari konektivitas internet yang tersambung secara terus menerus. Semakin berkembangnya Infrastruktur teknologi internet, semakin berkembang pula sistem komputerisasi yang ada, dimana tidak hanya *smartphone* atau komputer yang dapat terkoneksi dengan internet, tetapi berbagai macam benda elektronik dalam kehidupan sehari-hari misalnya mobil, mesin, perabot rumah, saklar lampu dan benda apa saja yang dapat tersambung ke jaringan lokal dan global menggunakan sensor atau actuator yang tertanam.

Menurut (Mukhammad, 2020), Konsep *Internet of Things* mencakup tiga elemen utama yaitu: benda fisik atau nyata yang telah diintegrasikan pada modul sensor, koneksi internet, dan pusat data pada server untuk menyimpan data ataupun informasi dari aplikasi.

Internet of Things (IoT) adalah konsep yang mengacu pada jaringan perangkat fisik yang terhubung dan saling berkomunikasi melalui internet. IoT menggabungkan teknologi, perangkat keras, dan perangkat lunak untuk memungkinkan objek sehari-hari, seperti perangkat rumah tangga, kendaraan, atau bahkan bangunan, untuk mengumpulkan dan bertukar data. Tujuan utama dari IoT adalah meningkatkan efisiensi, kenyamanan, dan kehidupan yang lebih baik bagi pengguna. Dengan menggunakan sensor, pengendali, dan perangkat lunak cerdas, IoT memungkinkan objek untuk berinteraksi secara otomatis atau diperintah dari jarak jauh melalui jaringan internet.

3.2.4. Logika *Fuzzy*

Kata *Fuzzy* adalah kata sifat yang berarti kabur atau tidak jelas. Fuzziness atau kekaburan atau ketidakjelasan selalu ada di kehidupan manusia. Logika *Fuzzy* adalah suatu cara yang tepat untuk memetakan ruang input kedalam suatu ruang output.

Logika *Fuzzy* dikembangkan oleh Prof. Lotfi Zadeh dari Universitas California USA, pada tahun 1965. Zadeh memodifikasi teori himpunan, dimana setiap anggotanya memiliki derajat keanggotaan antara 0 sampai 1 yang disebut dengan himpunan kabur. Pada logika *Fuzzy* sebuah nilai bisa benar dan salah secara bersamaan namun berapa besar nilai kebenaran dan kesalahan tergantung pada bobot keanggotaannya (Batubara, 2017).

Menurut (Mukhammad, 2020), Logika *Fuzzy* bekerja menggunakan derajat keanggotaan dari sebuah nilai yang kemudian digunakan untuk menentukan hasil yang diinginkan berdasarkan faktor yang telah ditentukan. Sebelumnya telah dijelaskan bahwa logika *Fuzzy*

memetakan ruang input ke ruang output. Antara input dan output terdapat suatu kotak hitam yang harus memetakan input ke output yang sesuai.

3.2.4.1. Himpunan *Fuzzy*

Himpunan *Fuzzy* adalah himpunan yang elemen-elemennya memiliki derajat keanggotaan. Himpunan *Fuzzy* telah diperkenalkan oleh L. A. Zadeh (1965) sebagai perluasan dari pengertian himpunan klasik (Davvaz et al., 2021). Himpunan *Fuzzy* merupakan suatu konsep dalam logika *Fuzzy* yang digunakan untuk menggambarkan ketidakpastian dan keambiguan dalam pemodelan matematis. Berbeda dengan himpunan crisp dalam logika klasik yang hanya memiliki dua nilai keanggotaan, yaitu 0 dan 1, himpunan *Fuzzy* memungkinkan elemen atau variabel memiliki tingkat keanggotaan yang berkisar antara 0 hingga 1.

Sebuah himpunan *Fuzzy* terdiri dari beberapa fungsi keanggotaan yang menggambarkan sejauh mana suatu elemen atau variabel termasuk dalam himpunan tersebut. Fungsi keanggotaan ini dapat berbentuk segitiga, trapesium, kurva Gaussian, atau jenis fungsi lainnya, tergantung pada karakteristik dan kebutuhan dari himpunan yang didefinisikan.

Misalnya, jika kita ingin mendefinisikan himpunan *Fuzzy* "tinggi" untuk ketinggian seseorang, kita dapat menggunakan fungsi keanggotaan segitiga dengan tiga parameter: rendah, sedang, dan tinggi. Sehingga, suatu nilai ketinggian tertentu akan memiliki tingkat keanggotaan dalam himpunan "tinggi" sesuai dengan sejauh mana nilai tersebut mendekati ketinggian tinggi yang ditentukan.

Himpunan *Fuzzy* dapat digunakan untuk menggambarkan dan memodelkan sifat-sifat yang tidak dapat diukur dengan tepat atau memiliki tingkat ketidakpastian dalam pengambilan keputusan. Dalam aplikasi praktis, himpunan *Fuzzy* sering digunakan dalam sistem kontrol, pengambilan keputusan, pengenalan pola, dan bidang-bidang lain di mana pemodelan matematis yang eksak sulit

dilakukan. Penggunaan himpunan *Fuzzy* memungkinkan kita untuk menggambarkan dan memanipulasi konsep-konsep yang bersifat kabur dan subjektif, sehingga logika *Fuzzy* dapat memberikan solusi yang lebih adaptif dan fleksibel dalam situasi-situasi yang kompleks.

Dalam memahami logika *Fuzzy*, ada beberapa hal lain yang perlu diketahui, antara lain:

1. Fungsi Keanggotaan (*Membership Functions*): Logika *Fuzzy* menggunakan fungsi keanggotaan untuk menentukan sejauh mana suatu elemen atau variabel termasuk dalam suatu himpunan *Fuzzy*. Fungsi keanggotaan ini dapat berbentuk segitiga, trapesium, atau jenis fungsi lainnya.
2. Operasi Logika *Fuzzy*: Logika *Fuzzy* menggunakan operasi logika *Fuzzy* seperti konjungsi (*AND Fuzzy*), disjungsi (*OR Fuzzy*), dan negasi (*NOT Fuzzy*) untuk menggabungkan dan memanipulasi himpunan *Fuzzy*.
3. Implikasi *Fuzzy*: Implikasi *Fuzzy* digunakan dalam aturan *Fuzzy* untuk menghubungkan kondisi dan konklusi dalam suatu sistem logika *Fuzzy*. Implikasi *Fuzzy* menentukan sejauh mana suatu kondisi menyebabkan konklusi tertentu.
4. Penggabungan Aturan *Fuzzy*: Dalam logika *Fuzzy*, aturan-aturan *Fuzzy* yang berbeda dapat digabungkan untuk menghasilkan suatu keluaran yang akhir. Penggabungan aturan *Fuzzy* ini dapat dilakukan menggunakan metode seperti metode Mamdani atau metode Sugeno.
5. Defuzzifikasi: Defuzzifikasi adalah proses mengubah hasil dari operasi-operasi logika *Fuzzy* menjadi nilai tegas yang dapat digunakan untuk pengambilan keputusan. Proses ini melibatkan mengubah himpunan *Fuzzy* menjadi suatu nilai tunggal yang mewakili keluaran sistem *Fuzzy*.

6. Aplikasi Logika *Fuzzy*: Logika *Fuzzy* telah diterapkan dalam berbagai bidang seperti kendali sistem, pengambilan keputusan, pengenalan pola, pengolahan citra, dan lain sebagainya. Kemampuan logika *Fuzzy* untuk mengatasi ketidakpastian dan kompleksitas membuatnya berguna dalam situasi di mana pemodelan matematis yang eksak sulit dilakukan.
7. Tingkat Keanggotaan: Dalam logika *Fuzzy*, suatu elemen atau variabel dapat memiliki tingkat keanggotaan yang berkisar antara 0 hingga 1. Tingkat keanggotaan ini menunjukkan sejauh mana elemen atau variabel termasuk dalam suatu himpunan *Fuzzy*. Tingkat keanggotaan 0 berarti tidak ada keanggotaan, sedangkan tingkat keanggotaan 1 berarti keanggotaan penuh.
8. Penerapan *Fuzzy*: Dalam penerapan logika *Fuzzy*, pemodelan linguistik digunakan untuk menggambarkan variabel-variabel dan aturan-aturan dalam suatu sistem. Pemodelan ini memungkinkan penggunaan bahasa manusia dan pengetahuan intuitif dalam pengambilan keputusan.

3.2.4.2. Metode *Fuzzy Mamdani*

Fuzzy mamdani adalah metode yang juga sering di kenal dengan metode *max-min* atau *max-product*. Metode Mamdani dikenal juga sebagai metode min-max, diperkenalkan oleh Ebrahim Mamdani pada tahun 1975 (Santya et al., 2021).

Fuzzy Mamdani, adalah salah satu metode yang paling umum digunakan dalam logika *Fuzzy* untuk pengambilan keputusan. Sistem *Fuzzy Mamdani* menggunakan aturan-aturan *Fuzzy* yang terdiri dari kondisi dan konklusi untuk menghasilkan keluaran berdasarkan pemrosesan data *input* yang kabur. Aturan-aturan ini menggunakan fungsi keanggotaan untuk menentukan sejauh mana suatu elemen atau variabel termasuk dalam himpunan *Fuzzy*.

Berikut adalah langkah-langkah umum dalam proses sistem *Fuzzy Mamdani*:

1. Fuzzifikasi

Data input yang *crisp (non-Fuzzy)* diubah menjadi data input yang *Fuzzy* dengan menggunakan fungsi keanggotaan. Fuzzifikasi mengkonversi nilai-nilai *crisp* ke dalam tingkat keanggotaan dalam himpunan *Fuzzy*.

2. Penyamaan Aturan

Aturan-aturan *Fuzzy* diterapkan pada data input *Fuzzy* untuk menghasilkan konklusi *Fuzzy*. Setiap aturan terdiri dari kondisi (*premises*) dan konklusi (*conclusion*), yang saling terhubung dengan implikasi *Fuzzy*.

3. Aturan

Konklusi-konklusi *Fuzzy* yang dihasilkan dari aturan-aturan *Fuzzy* dikombinasikan untuk menghasilkan suatu himpunan *Fuzzy* gabungan. Metode penggabungan yang umum digunakan dalam sistem *Fuzzy Mamdani* adalah menggunakan operasi logika *Fuzzy OR*.

4. Defuzzifikasi

Hasil gabungan himpunan *Fuzzy* dikonversi kembali menjadi nilai *crisp* yang dapat digunakan untuk pengambilan keputusan. Ada beberapa metode defuzzifikasi yang umum digunakan dalam *Fuzzy Mamdani*, seperti metode *centroid*, metode *weighted average*, atau metode lainnya.

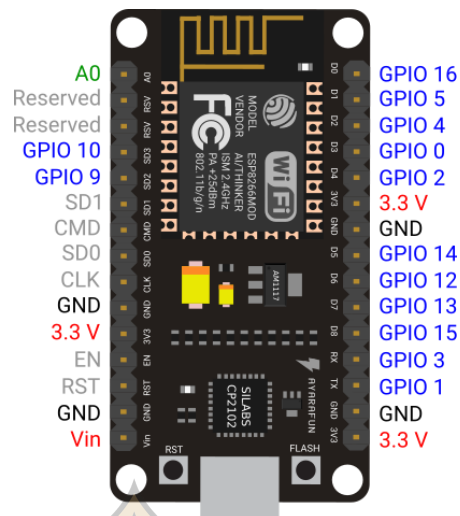
Sistem *Fuzzy Mamdani* memiliki kelebihan dalam menggambarkan pengetahuan manusia secara intuitif dan dapat mengatasi ketidakpastian. Namun, kelemahan utamanya adalah kompleksitas perhitungan dan interpretasi yang rumit karena menggunakan aturan-aturan *Fuzzy* yang kompleks.

3.2.5. NodeMCU ESP8266

NodeMCU adalah sebuah *board* elektronik yang berbasis *chip* ESP8266 dengan kemampuan menjalankan fungsi mikrokontroler dan juga koneksi internet (WiFi). Terdapat beberapa pin I/O sehingga dapat dikembangkan menjadi sebuah aplikasi monitoring maupun *controlling* pada proyek IOT. NodeMCU ESP8266 dapat diprogram dengan *compiler*-nya Arduino, menggunakan Arduino IDE. Bentuk fisik dari NodeMCU ESP 8266, terdapat port USB (mini USB) sehingga akan memudahkan dalam pemrogramannya (Nurul, 2019).

ESP8266 memiliki kemampuan untuk *networking* yang lengkap dan menyatu baik sebagai *client* maupun sebagai *Access Point*. *Firmware* yang dimiliki ESP8266 begitu banyak, dapat juga sebuah chip ESP8266 diprogram dengan tujuan khusus sesuai dengan kebutuhan sebagai contoh kemampuan untuk berkomunikasi dengan web yang menggunakan port HTTPS.

Chip ESP8266 disempurnakan oleh Tensilica's seri L106 Diamond dengan prosesor 32-bit. Ada 3 cara menggunakan ESP8266 : sebagai wifi *access* menggunakan AT *command*, dimana biasanya dimanfaatkan oleh Arduino untuk koneksi wifi, sebagai sistem yang berdiri sendiri menggunakan NodeMCU dan menggunakan bahasa LUA, sebagai sistem yang berdiri sendiri dengan menggunakan Arduino IDE yang sudah bisa terhubung dengan ESP8266. ESP 8266 dapat bertindak sebagai *client* ke suatu wifi router, sehingga saat konfigurasi dibutuhkan setting nama access pointnya dan juga passwordnya, selain itu ESP8266 dapat digunakan sebagai *Access Point* dimana ESP8266 dapat menerima akses wifi.



Gambar 3.1 NodeMCU ESP8266

NodeMCU ESP8266 memiliki total pin sejumlah 30 pin yang memiliki fungsi bermacam-macam, berikut adalah fungsi dari tiap pin:

1. Power Pin

Terdapat 4 sumber power output, satu buah pin VIN dan tiga buah pin 3.3V pin VIN dapat digunakan secara langsung sebagai power supply untuk board NodeMCU, Kemudian pin 3.3V dapat digunakan untuk mensuplai power komponen tambahan dari NodeMCU. Selain penggunaan pin terdapat juga port USB yang dapat digunakan untuk power supply langsung ke board dan juga pengiriman sketch dari kode editor (Arduino IDE).

2. GND Pin

Pin yang digunakan untuk sambungan ke *Ground* dari *board* NodeMCU, terdapat 4 buah pin GND.

3. GPIO Pin

Terdapat 17 pin GPIO yang dapat digunakan untuk berbagai fungsi yang berbeda I2C, I2S, PWM, IR Remote, LED dan tombol yang terprogram. Pin ini dapat digunakan sebagai *input* dan *output*.

4. Control Pin

Pin yang digunakan untuk mengontrol NodeMCU, pin ini terdiri dari pin EN, pin RST yang dapat digunakan untuk *restart* NodeMCU.

5. Pin SPI

Komponen yang terhubung dengan pin SPI akan bersifat *Full Duplex*, yakni adanya hubungan *Master* dan *Slave*. Dengan pin ini dapat digunakan untuk sinkronisasi deteksi bit pada *receiver*.

6. Pin Analog

Tidak seperti Arduino, NodeMCU hanya memiliki 1 buah pin analog yang dapat digunakan untuk pembacaan data secara analog, misalnya pembacaan sensor.

7. Pin SDIO

NodeMCU ESP8266 memiliki fitur *Secure Digital/Input Interface* yang dapat digunakan secara langsung untuk *interfacing SD Card*.

Table 3.2 Spesifikasi NodeMCU ESP8266

Spesifikasi	NodeMCU ESP8266
Mikrokontroler	Tensilica 32-bit RISC CPU Xtensa LX106
Tegangan Operasi	3.3 V
Tegangan Masukan	7 – 12 V
Pin Digital I/O	16
Pin Analog Input (ADC)	1
AURTs	2
SPIs	1
I2Cs	1
Flash Memory	4 MB
SRAM	64 KB
Clock Speed	80 MHZ
PCB Antenna	1

3.2.6. Sensor MQ-135

Sensor MQ-135 adalah sensor yang mempunyai kemampuan untuk mendeteksi banyak gas, seperti gas CO, CO₂, NH₄, Acetone, Toulene dan Ethanol. Sensor ini bekerja dengan cara menerima perubahan nilai resistansi (pin analog) bila terkena gas. Sensor ini memiliki daya tahan yang baik untuk penggunaan penanda bahaya polusi karena praktis dan tidak memakan daya yang besar.

Penyesuaian sensitifitas sensor ditentukan oleh nilai resistansi dari MQ-135 yang berbeda-beda untuk berbagai konsentrasi gas (Rosa, 2020). Nilai resistansi dari MQ-135 berbeda untuk berbagai jenis dan berbagai konsentrasi gas. Maka dari itu kalibrasi sangat diperlukan untuk perhitungan agar dapat dilakukan penyesuaian sensitivitas. Sebelum dilakukan perhitungan proses *pre-heat* selama 24 jam harus dilakukan karena nantinya akan mempengaruhi sensitivitas (Huanwei Electronics, 2014).



Gambar 3.2 MQ-135

Table 3.3 Spesifikasi standar kerja Sensor MQ-135

Parameter	Kondisi Teknis	Keterangan
<i>Circuit Voltage</i>	5V±0,1	AC atau DC
<i>Heating Voltage</i>	5V±0,1	AC atau DC
<i>Load Resistance</i>	Bisa menyesuaikan	
<i>Heater Resistance</i>	33 Ω ± 5%	Suhu Ruangan
<i>Heating Consumption</i>	Kurang dari 800mw	

3.2.7. Sensor DHT11

DHT11 adalah sensor digital yang dapat mengukur suhu dan kelembaban udara di sekitarnya. Sensor ini sangat mudah digunakan bersama dengan mikrokontroler sejenis Arduino. Memiliki tingkat stabilitas yang sangat baik serta fitur kalibrasi yang sangat akurat. Koefisien kalibrasi disimpan dalam *OTP program memory* (Triyanto, 2016), sehingga ketika internal sensor mendeteksi sesuatu, maka modul ini menyertakan koefisien tersebut dalam kalkulasinya. DHT11 ini termasuk sensor yang memiliki kualitas terbaik, dinilai dari respon, pembacaan data yang cepat, dan kemampuan anti-interference. Ukurannya yang kecil, dan dengan transmisi sinyal hingga 20 meter, dengan spesifikasi: *Supply Voltage: +5 V, Temperature range : 0-50 °C error of ± 2 °C, Humidity : 20-90% RH $\pm 5\%$ RH error*, dengan spesifikasi digital interfacing system. Produk ini cocok digunakan untuk banyak aplikasi-aplikasi pengukuran suhu dan kelembaban.



Gambar 3.3 Sensor DHT11

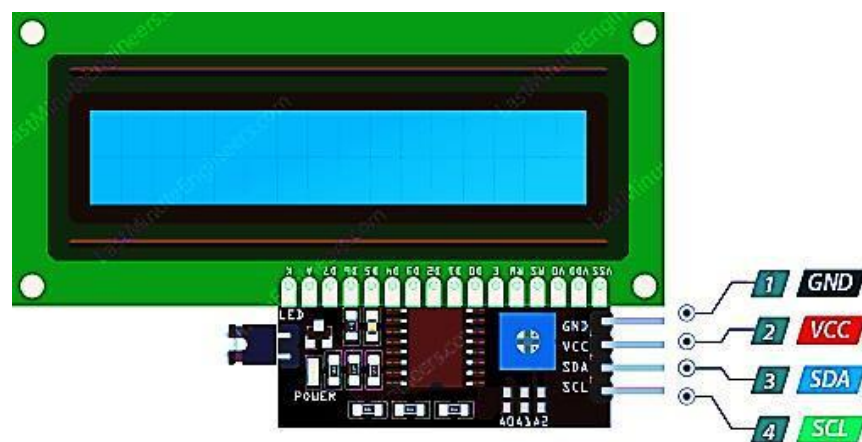
Table 3.4 karakteristik sensor DHT11

Model	DHT11
<i>Power supply</i>	3-5,5 V DC
<i>Output signal</i>	<i>digital signal via single-bus</i>
<i>Measuring range</i>	<i>humidity 20-90% RH $\pm 5\%$ RH error</i>

	<i>temperature 0-50 °C error of ± 2 °C</i>
<i>Accuracy</i>	<i>humidity $\pm 4\%$ RH (Max $\pm 5\%$ RH); temperature ± 2.0 Celsius</i>
<i>Resolution or Sensitivity</i>	<i>humidity 1% RH; temperature 0.1 Celsius</i>
<i>Repeatability</i>	<i>humidity $\pm 1\%$ RH; temperature ± 1 Celsius</i>
<i>Humidity hysteresis</i>	<i>$\pm 1\%$ RH</i>
<i>Long-term Stability</i>	<i>$\pm 0.5\%$ RH/ year</i>
<i>Sensing period</i>	<i>Average: 2s</i>
<i>Interchangeability</i>	<i>fully interchangeable</i>
<i>Dimensions size</i>	<i>12*15.5*5.5 mm</i>

3.2.8. LCD 12C (*Liquid Crystal Display Inter-Integrated Circuit*)

Liquid Crystal Display (LCD) I2C adalah jenis layar LCD yang menggunakan protokol komunikasi I2C (*Inter-Integrated Circuit*) untuk menghubungkan LCD dengan mikrokontroler atau perangkat lainnya. Dengan menggunakan komunikasi I2C, pengguna dapat mengontrol dan menampilkan informasi pada LCD dengan menggunakan hanya dua jalur komunikasi, yaitu SDA (*Serial Data Line*) dan SCL (*Serial Clock Line*) (Susanto, 2020).



Gambar 3.4 LCD 12C (*Liquid Crystal Display Inter-Integrated Circuit*)

Berikut adalah beberapa jenis LCD I2C yang umum digunakan:

1. LCD 16x2 I2C

LCD ini memiliki 2 baris dan masing-masing baris dapat menampilkan hingga 16 karakter. Dengan total 32 karakter yang dapat ditampilkan, LCD 16x2 I2C sangat umum digunakan dalam berbagai proyek elektronik dan prototipe.

2. LCD 20x4 I2C

LCD ini memiliki 4 baris dan masing-masing baris dapat menampilkan hingga 20 karakter. Dengan total 80 karakter yang dapat ditampilkan, LCD 20x4 I2C memberikan lebih banyak ruang untuk menampilkan informasi dibandingkan dengan LCD 16x2.

3. LCD Graphical I2C

Jenis ini merupakan LCD I2C yang dapat menampilkan grafis dan ikon selain karakter teks. LCD Graphical I2C tersedia dalam berbagai resolusi dan ukuran, seperti 128x64 piksel atau 240x320 piksel, dan dapat digunakan untuk menampilkan informasi grafis seperti gambar, grafik, dan antarmuka pengguna (UI) yang lebih kompleks.

3.2.9. Kabel *Jumper*

Kabel *Jumper* adalah suatu istilah kabel yang ber-diameter kecil yang di dalam dunia elektronika digunakan untuk menghubungkan dua titik atau lebih dan dapat juga untuk menghubungkan 2 komponen elektronika. Kabel *Jumper* disebut juga dengan istilah kabel dupont. Kabel dupont adalah kabel yang di kedua ujungnya dilengkapi dengan bagian yang memudahkan untuk dihubungkan ke komponen lain. Tiga jenis kabel ini dicantumkan (Pradana, 2020).

Jenis kabel *Jumper* dibedakan berdasarkan konektor kabelnya, yaitu:

1. *Male to Male*

Kabel *Jumper* ini adalah kabel yang kedua ujungnya dapat digunakan untuk koneksi (memiliki pin / *connector*).



Gambar 3.5 Kabel *Jumper Male to Male*

2. *Male to Female*

Kabel *Jumper* ini adalah kabel salah satu ujungnya terdapat pin yang bisa digunakan untuk *connector*, sedangkan salah satunya lagi tidak terdapat pin / *connector*.



Gambar 3.6 Kabel *Jumper Male to Female*

3. *Female to Female*

Kabel *Jumper* jenis ini digunakan untuk koneksi *female to female* dimana pada kedua ujungnya tidak terdapat pin / *connector*.



Gambar 3.7 Kabel *Jumper Female to Female*

3.2.10. 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 (Pradana, 2020).



Gambar 3.8 Blynk IoT

Cara kerja Blynk adalah pengguna akan mengoperasikan *hardware* melalui aplikasi Blynk. Aplikasi ini kemudian akan mengirimkan data berupa perintah yang sudah dituliskan sebelumnya melalui Blynk *Libraries*. Selanjutnya komunikasi akan diteruskan dengan bantuan koneksi internet melalui WiFi hingga sampai dan dikerjakan oleh *hardware*. Aplikasi Blynk menyediakan token yang berfungsi sebagai keamanan dan integrasi antara Blynk dengan mikrokontroler.

Pada blynk versi terbaru yakni Blynk IoT terdapat dua macam *dashboard* yakni untuk *mobile* dan juga web. Dalam *dashboard* web memiliki fitur yang mirip dengan *mobile*, dimana tersedia 80 buah widget yang dapat digunakan seperti *widget switch*, *slider*, LCD, label dan lainnya. Dalam penggunaan aplikasi *mobile* blynk IoT tidak dapat terlepas dari *dashboard website* blynk, dikarenakan pembuatan *datastream* hanya bisa dilakukan dalam *dashboard website*.

3.2.11. *FlowChart*

Menurut (Pradana, 2020), *FlowChart* adalah bagan - bagan yang mempunyai arus yang menggambarkan langkah - langkah penyelesaian suatu masalah.

Fungsi utama dari *FlowChart* adalah memberikan gambaran jelas tentang jalannya sebuah program dari satu proses ke proses lainnya, sehingga alur program dapat dengan mudah dipahami oleh semua orang. Selain itu, *FlowChart* juga memiliki fungsi lain yaitu menyederhanakan rangkaian prosedur agar informasi tersebut lebih mudah dipahami.

FlowChart sendiri terdiri dari lima jenis yang masing-masing memiliki karakteristik penggunaan yang berbeda. Berikut adalah jenis-jenis *FlowChart*:

1. *FlowChart* proses

FlowChart proses Digunakan untuk menggambarkan urutan langkah-langkah dalam proses bisnis atau alur kerja yang sistematis.

2. *FlowChart* keputusan

FlowChart keputusan Digunakan untuk menunjukkan cabang keputusan dalam suatu proses, dengan berbagai pilihan dan kondisi yang harus dipertimbangkan.

3. *FlowChart* data

FlowChart data Digunakan untuk menggambarkan aliran data dan informasi dalam suatu sistem atau program.


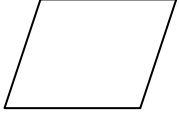
4. *FlowChart* dokumentasi

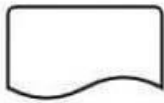



FlowChart dokumentasi Digunakan untuk menyajikan informasi rinci tentang suatu proses atau alur kerja, termasuk data masukan, *output*, dan langkah-langkah yang terlibat.

5. *FlowChart* sistem

FlowChart sistem Digunakan untuk menggambarkan interaksi antara komponen-komponen dalam sistem yang kompleks, dengan menggambarkan alur data dan kendali.

Table 3.5 symbol-simbol *Flowchart*

Simbol	Nama	Fungsi
	<i>Flow</i>	Simbol yang digunakan untuk menggabungkan antara symbol yang satu dengan simbol yang lain, simbol ini disebut dengan <i>connecting line</i> .
	<i>On-Page Reference</i>	Simbol untuk keluar-masuk atau penyambungan proses dalam lembar kerja yang sama.
	<i>Off-Page Reference</i>	Simbol untuk keluar-masuk atau penyambungan proses dalam lembar kerja yang berbeda.
	Terminator	Simbol yang menyatakan awal atau akhir suatu program.
	Proses	Simbol yang menyatakan suatu proses yang dilakukan dengan computer.
	<i>Decision</i>	Simbol yang menunjukan kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban yaitu ya dan tidak.
	<i>Input atau Output</i>	Simbol yang menyatakan proses <i>input</i> atau <i>output</i> tanpa tergantung peralatan.
	<i>Manual Operation</i>	Simbol yang menyatakan suatu proses yang tidak dilakukan oleh komputer.

	<i>Document</i>	Simbol yang menyatakan bahwa input berasal dari dokumen dalam bentuk fisik atau <i>output</i> yang perlu dicetak.
	<i>Predefine Proses</i>	Simbol untuk pelaksanaan suatu bagian sub-program atau <i>procedure</i> .
	<i>Display</i>	Simbol yang menyatakan peralatan <i>output</i> yang digunakan.
	<i>Preparation</i>	Simbol yang menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberikan nilai awal.



USM