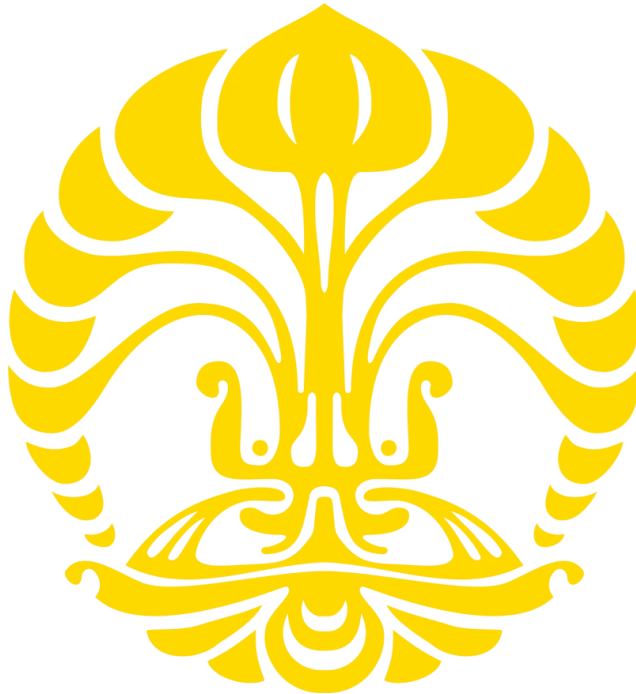


**SISTEM EMBEDDED**  
**Monitoring Kualitas Udara**



Kelas Sistem Embedded 1 - 02

Anggota Kelompok :

<b>Mustofa Kamal</b>	<b>1706043020</b>
<b>Muhammad Alfiyansyah</b>	<b>1706986044</b>
<b>Muhammad Koku</b>	<b>1706043203</b>
<b>Muhammad Ilham Akbar</b>	<b>1706042970</b>

**TEKNIK KOMPUTER FAKULTAS TEKNIK**  
**UNIVERSITAS INDONESIA**  
**2019/2020**

## KATA PENGANTAR

Puji syukur kami panjatkan kehadiran Tuhan Yang Maha Esa yang telah melimpahkan rahmat dan hidayah-Nya sehingga kami dapat menyelesaikan laporan proyek Sistem Embedded yang berjudul “Monitoring Kualitas Udara” dengan tepat waktu dan tanpa halangan yang berarti. Salah satu hal yang menjadi latar belakang pembuatan laporan ini adalah untuk menjadi bahan pembelajaran dan pemahaman terhadap proyek yang telah kami buat. Oleh karena itu, dengan laporan ini diharapkan bisa memperdalam pemahaman tentang cara kerja dari alat yang telah kami buat. Dalam pembuatan laporan dan proyek ini, kami mendapat banyak bimbingan, masukan dan juga saran dari berbagai pihak. Oleh karena itu, kami mengucapkan terima kasih kepada pihak-pihak terkait yang sudah membantu penyelesaian laporan dan proyek kami. Pihak-pihak tersebut antara lain :

1. I Gde Dharma Nugraha S.T., M.T. sebagai dosen kelas Sistem Embedded 1-02.
2. Fransiskus Astha Ekadiyanto S.T., M.Sc. sebagai dosen kelas Sistem Embedded 1-01.
3. Teman-teman dari kelompok lain sebagai pemberi saran dan masukan.
4. Semua pihak terkait yang telah memberikan dukungan terhadap pembuatan laporan proyek dan alat yang menjadi tujuan proyek ini.

Semoga laporan proyek “Monitoring Kualitas Udara” bisa bermanfaat bagi pembaca dan dapat dipergunakan sebagaimana mestinya. Kami menyadari masih banyak kekurangan dari laporan proyek ini. Oleh karena itu, kami membutuhkan kritik, saran, dan masukan demi perbaikan untuk kedepannya.

Depok, 20 Desember 2019

Kelompok

## DAFTAR ISI

KATA PENGANTAR .....	1
DAFTAR ISI.....	2
BAB I PENDAHULUAN.....	3
1.1. Latar Belakang .....	3
1.2. Rumusan Masalah.....	4
1.3. Tujuan .....	4
BAB II DASAR TEORI .....	5
2.1. Node MCU ESP8266.....	5
2.2. MQ135 .....	7
2.3. DHT11 .....	11
2.4. Spesifikasi Platform Thinger .....	16
2.5. UML (Unified Modelling Language) .....	17
2.6. Bahasa Pemrograman dan Software Compiler .....	21
2.7. Parameter Kesehatan Udara .....	22
BAB III PEMBAHASAN DAN PERANCANGAN .....	24
3.1. Deskripsi Proyek .....	24
3.2. UML (Unified Modelling Language) Proyek .....	25
3.3. Schematic / Wiring Alat.....	28
3.4. Fitur-fitur Pada Platform yang Dimanfaatkan .....	29
3.5. Code / Sketch Perangkat Monitoring Kualitas Udara.....	32
BAB IV PENUTUP .....	35
4.1. Saran .....	35
4.2. Kesimpulan .....	35
DAFTAR PUSTAKA .....	36
LAMPIRAN.....	37

# **BAB I**

## **PENDAHULUAN**

### **1.1. Latar Belakang**

Udara merupakan salah satu komponen yang sangat penting bagi makhluk hidup, karena udara merupakan sumber kehidupan bagi makhluk hidup yang ada di dunia ini. Udara tersebut perlu dijaga dan dipelihara kualitasnya agar tidak memberikan dampak negatif bagi kesehatan manusia, sehingga dapat mendukung kehidupan makhluk hidup secara optimal. Namun, dewasa ini kualitas udara telah menurun di banyak tempat, karena terjadinya pencemaran udara atau polusi yang disebabkan oleh aktivitas manusia seperti aktivitas transportasi, industri, perumahan, pembakaran sampah, dan aktivitas lainnya yang memberikan dampak yang buruk bagi kualitas udara. Salah satu tempat yang tidak terlepas dari pencemaran udara yang disebabkan oleh kegiatan makhluk hidup adalah lingkungan kampus seperti lingkungan Fakultas Teknik (FT). Pada lingkungan kampus seperti FT ini tidak terlepas dari aktivitas yang dapat mencemari udara seperti aktivitas menggunakan kendaraan bermotor, merokok tidak pada tempatnya, asap dari kantin, percobaan di laboratorium yang berdampak pada kualitas udara, dan aktivitas lainnya.

Pencemaran udara atau polusi ini menjadi masalah yang sangat penting bagi kehidupan makhluk hidup. Oleh karena itu, diperlukannya suatu alat yang dapat memonitoring atau memantau kadar polusi udara disuatu tempat secara realtime dan dapat diakses dimanapun serta dapat memberikan peringatan jika kadar polusi udara telah melebihi batas kesehatan (keamanan) yang telah ditentukan. Batas kesehatan tersebut berdasarkan parameter yang telah ditentukan oleh badan kesehatan.

Alat yang digunakan untuk memantau kadar polusi ini akan mengimplementasikan teknologi IoT (Internet of Things) karena penginformasian data secara online dan realtime dapat memberikan akses yang luas bagi banyak orang dengan informasi yang akurat. Selain itu, juga digunakannya sensor gas MQ135 yang merupakan sensor yang peka terhadap partikel-partikel yang terkandung dalam udara. Pada alat ini juga akan diberikan sensor-sensor tambahan yang juga berkaitan dengan pemantuan kesehatan udara seperti penggunaan DHT11 yang merupakan sensor kelembapan dan suhu.

## 1.2. Rumusan Masalah

Berdasarkan latar belakang dan permasalahan yang ada, maka diperoleh rumusan masalah sebagai berikut :

- a. Bagaimana spesifikasi alat yang dibutuhkan untuk merancang suatu perangkat yang dapat mendeteksi kadar partikel pada suatu gas atau udara melalui sensor-sensor yang akan digunakan ?
- b. Bagaimana cara kerja dari perangkat yang dapat menentukan tingkat atau kategori polusi udara di suatu tempat ?
- c. Apa saja pin-pin yang digunakan dan dihubungkan sehingga perangkat dapat bekerja secara optimal dan baik?
- d. Bagaimana cara melakukan transfer data ke suatu server platform agar dapat dipantau secara online dan realtime ?
- e. Bagaimana code atau sketch yang sesuai sehingga perangkat dapat bekerja dengan baik dan sesuai dengan tujuan pembuatan?

## 1.3. Tujuan

- a. Mengetahui dan memahami spesifikasi platform dan alat yang dibutuhkan untuk merancang suatu perangkat yang dapat mendeteksi kadar partikel pada suatu gas atau udara melalui sensor-sensor yang akan digunakan
- b. Mengetahui cara kerja dari perangkat yang dapat menentukan tingkat atau kategori polusi udara di suatu tempat.
- c. Mengetahui dan memahami pin-pin mana saja yang digunakan dan dihubungkan sehingga perangkat dapat bekerja secara optimal dan baik.
- d. Mengetahui dan memahami cara transfer data ke suatu server platform sehingga dapat dipantau secara online dan realtime.
- e. Memahami code atau sketch yang telah dibuat sesuai dengan tujuan dan perangkat berfungsi dengan baik.

## **BAB II**

### **DASAR TEORI**

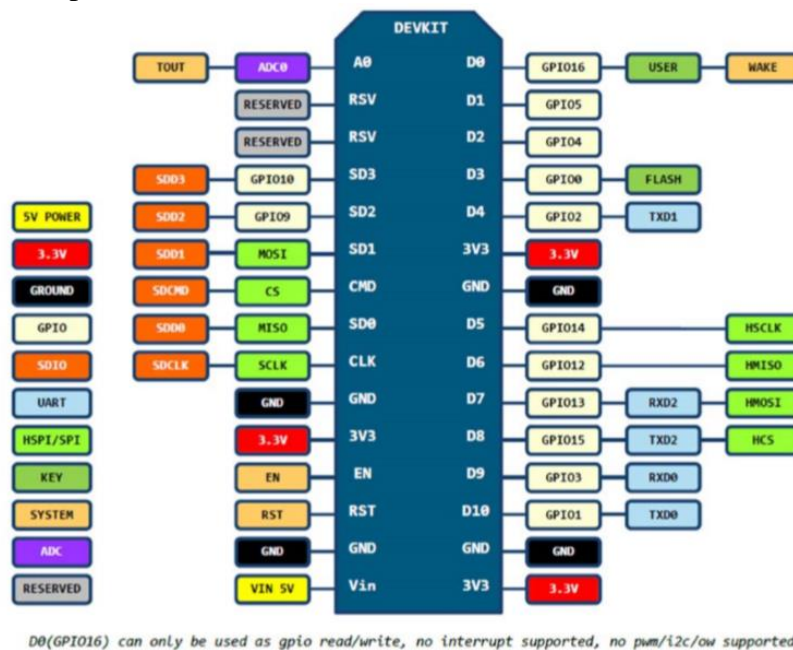
#### **2.1. Node MCU ESP8266**

ESP8266 adalah suatu microcontroller yang dibuat oleh Espressif Systems. ESP8266 ini memiliki komponen yang dapat terhubung ke jaringan WIFI sehingga perangkat ini merupakan solusi penghubung suatu microcontroller ke WIFI dan juga dapat menjalankan aplikasi mandiri. Perangkat ini juga dilengkapi dengan konektor USB dan juga pin in-out. Adanya USB tersebut membantu user untuk dapat menghubungkan ESP8266 ke perangkat PC atau laptop milik user sehingga user dapat melakukan flash terhadap perangkat tersebut. Untuk melakukan upload Code ke NodeMCU ini dapat menggunakan software IDE.

##### **2.1.1. Spesifikasi Alat**

- Voltage : 3.3V.
- Current consumption : 10uA~170mA.
- Flash memory attachable : 16MB max (512K normal).
- Processor : Tensilica L106 32-bit.
- RAM : 32K + 80K.
- GPIOs : 17 (multiplexed with other functions).
- Analog to Digital : 1 input with 1024 step resolution.
- Processor speed: 80~160MHz.
- Integrated TCP/IP protocol stack.
- Wi-Fi Direct (P2P), soft-AP.
- +19.5dBm output power in 802.11b mode
- 802.11 support: b/g/n.
- Maximum concurrent TCP connections: 5

### 2.1.2. Pin pada ESP8266



### 2.1.3. NodeMCU GPIO

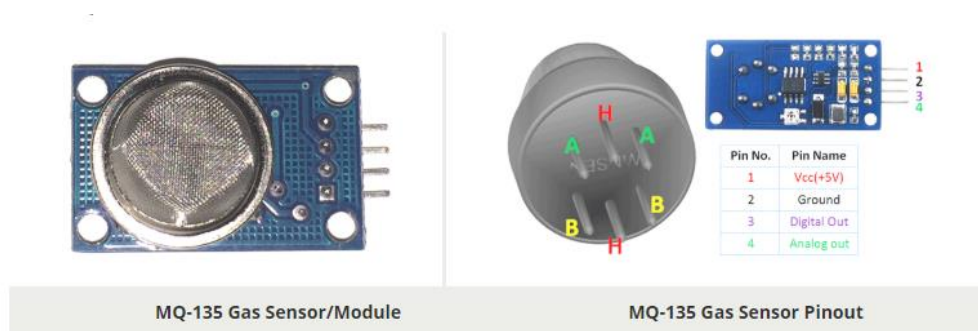
Adanya GPIO (General Purpose Input/Output) memungkinkan user untuk dapat mengakses pin ESP8266 karena untuk mengakses semua pin pada ESP8266 dapat menggunakan command GPIO dimana nomor GPIO tersebut berdasarkan pada indeks I/O ESP8266. Sebagai contoh untuk dapat mengakses atau terkoneksi dengan pin 'D1' pada ESP8266 maka pada command program dapat menggunakan command '5' (GPIO5) karena pin 'D1' ini dipetakan ke pin GPIO internal 5. Berikut tabel pemetaannya :

Nodemcu dev kit	ESP8266 Pin
D0	GPIO16
D1	GPIO5
D2	GPIO4
D3	GPIO0
D4	GPIO2
D5	GPIO14
D6	GPIO12
D7	GPIO13
D8	GPIO15

D9	GPIO3
D10	GPIO1
D11	GPIO9
D12	GPIO10

Pada pin D0 atau GPIO16 hanya dapat digunakan untuk read dan write, sedangkan pada pin lainnya tidak hanya read dan write saja tetapi juga dapat digunakan untuk PWM/I2C.

## 2.2. MQ135



### 2.2.1. Pin Konfigurasi Untuk modul

Pin number	Pin name	Deskripsi
1	Vcc	Digunakan untuk menyalakan sensor. Umumnya tegangan yang dibutuhkan sebesar +5V
2	Ground	Digunakan untuk menghubungkan modul ke system ground
3	Digital Out	Digunakan untuk mendapatkan output digital dari pin ini, dengan menetapkan nilai threshold menggunakan potensiometer
4	Analog Out	Pin ini menghasilkan tegangan analog 0-5V berdasarkan intensitas gas



#### Untuk Sensor

Pin number	Pin name	Deskripsi
1	H-pins	Dari dua pin H, satu pin terhubung ke supply dan yang lainnya ke ground
2	A-pins	Pin A dan pin B dapat dipertukarkan. Pin ini akan diikatkan ke tegangan suplai
3	B-pins	Pin A dan pin B dapat dipertukarkan. Satu pin akan bertindak sebagai output sementara yang lainnya akan menuju ground

#### 2.2.2. Fitur pada MQ-135

- Mendeteksi dalam ruang lingkup yang luas
- Mempunyai respons cepat dan Sensitivitas tinggi
- Stabil dan daya hidup tinggi
- Beroperasi pada tegangan +5 volt
- Mendeteksi/mengukur NH<sub>3</sub>, NO<sub>x</sub>, alcohol, Benzene, asap, CO<sub>2</sub> dll
- Output tegangan analog 0V ke 5V
- Output tegangan digital 0V ke 5V
- Dapat digunakan sebagai digital atau analog sensor
- Sensitivitas dari pin digital dapat divariasikan menggunakan potensiometer.

#### 2.2.3. Sensor dan modul MQ-135

Seri sensor gas MQ adalah yang paling murah dan umum digunakan. MQ135 dapat digunakan sebagai sebuah modul atau hanya sebagai sensor saja. Jika anda mencoba untuk hanya mendeteksi (tidak mengukur PPM) keberadaan gas, maka anda dapat membelinya sebagai modul karena dilengkapi dengan komparator op-amp dan pin digital output. Tetapi jika anda berencana untuk mengukur PPM gas, disarankan untuk membeli sensor tanpa modul.

#### 2.2.4. Tempat penggunaan MQ135 Gas sensor

Sensor gas MQ135 digunakan dalam peralatan control kualitas udara dan cocok untuk mendeteksi atau mengukur NH<sub>3</sub>, NO<sub>x</sub>, Alkohol, Benzene, Asap, CO<sub>2</sub>. Modul sensor MQ-135 dilengkapi dengan pin digital yang membuat sensor ini beroperasi bahkan tanpa menggunakan mikrokontroller dan hal ini sangat berguna ketika anda hanya mencoba untuk mendeteksi suatu gas tertentu. Jika anda perlu mengukur PPM pada gas, maka pin analog harus digunakan. Pin

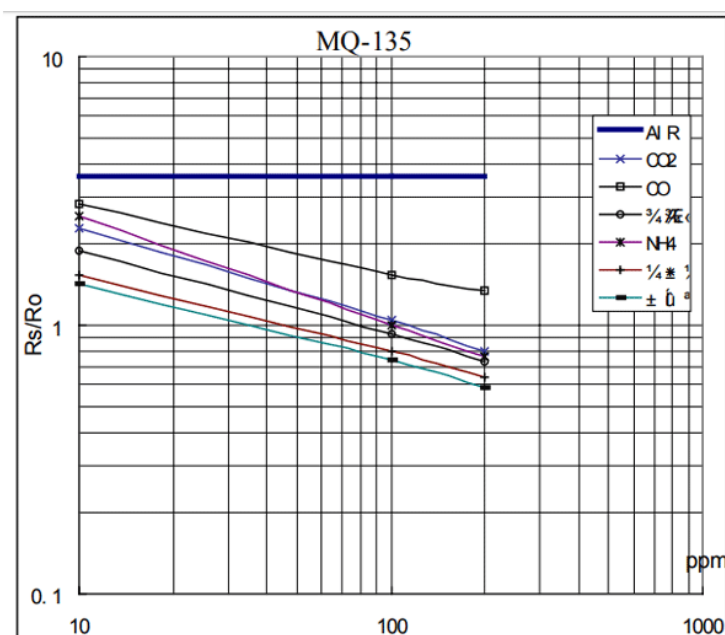
analog adalah TTL yang digerakkan dan bekerja pada 5V dan dapat digunakan dengan mikrokontroller umum.

#### 2.2.5. Cara Menggunakan MQ135 Untuk Mendeteksi Gas

Anda dapat menggunakan pin digital atau pin analog untuk melakukan hal ini. Cukup dengan mengaktifkan modul dengan 5V dan anda akan melihat lampu LED pada modul akan menyala ketika tidak ada gas terdeteksi namun hal ini terjadi karena sensor masih berada dalam keadaan pre-heating sebelum sensor benar benar bekerja. LED akan tetap dimatikan yang berarti pin digital output akan menjadi 0V. Ketika sensor gas mendeteksi adanya gas, output LED akan menjadi high bersama dengan pin digital (5V), jika tidak menggunakan potensiometer. Anda juga dapat menggunakan pin analog dalam hal yang sama. Nilai pin analog akan terbaca di kisaran 0 – 5 V dengan menggunakan mikrokontroller, nilai ini akan berbanding lurus dengan konsentrasi gas yang dideteksi sensor. Anda dapat bereksperimen dengan nilai nilai ini dan periksa bagaimana sensor bereaksi terhadap berbagai konsentrasi gas.

#### 2.2.6. Cara menggunakan MQ135 untuk mengukur PPM

Sensor gas MQ135 menggunakan SnO<sub>2</sub> yang memiliki resistansi lebih tinggi di udara jernih sebagai bahan sensor gas. Ketika ada peningkatan gas pencemar, resistansi dari sensor gas menurun seiring dengan hal itu. Untuk mengukur PPM menggunakan MQ135 kita perlu melihat grafik PPM ( $R_s/R_o$ ) v/s yang diambil dari datasheet MQ135.



Gambar diatas menunjukan karakteristik sensitivitas khas MQ-134 untuk beberapa gas, Dengan Temp : 20, Kelembapan: 65%, konsentrasi O2 21% dan RL 20k $\Omega$ , dimana

Ro : resistensi sensor pada 100ppm NH3 pada udara bersih

Rs : resistensi sensor pada berbagai konsentrasi

Nilai Ro adalah nilai resistansi di udara segar dan nilai Rs adalah nilai resistansi dalam konsentrasi gas. Pertama, Anda harus mengkalibrasi sensor dengan mencari nilai Ro pada udara segar dan kemudian nilai tersebut digunakan untuk menemukan Rs menggunakan formula dibawah ini :

**Resistance of sensor(Rs):**  $Rs = (V_c / V_{RL} - 1) \times R_L$

Setelah menghitung Rs dan Ro, anda dapat menemukan rasio dan kemudia menggunakan grafik yang ditunjukkan diatas, dan dapat menghitung nilai PPM yang setara untuk gas tersebut.

#### 2.2.7. Spesifikasi

- Standard work condition

Symbol	Parameter name	Technical condition	Remarks
V <sub>c</sub>	Circuit voltage	5V $\pm$ 0.1	AC OR DC
V <sub>H</sub>	Heating voltage	5V $\pm$ 0.1	AC OR DC
R <sub>L</sub>	Load resistance	can adjust	
R <sub>H</sub>	Heater resistance	33 $\Omega$ $\pm$ 5%	Room Tem
P <sub>H</sub>	Heating consumption	less than 800mw	

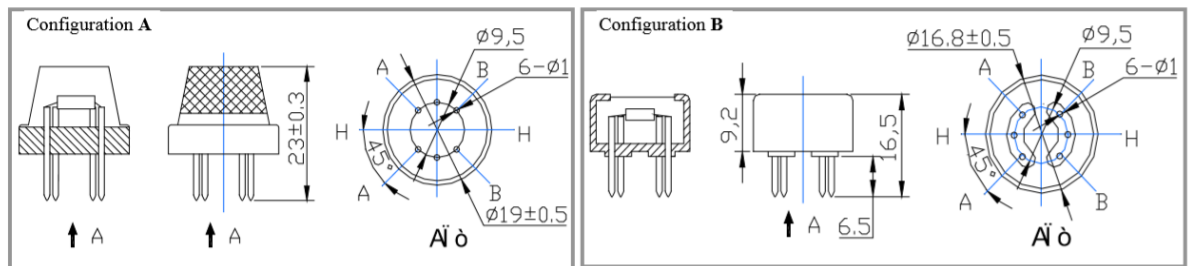
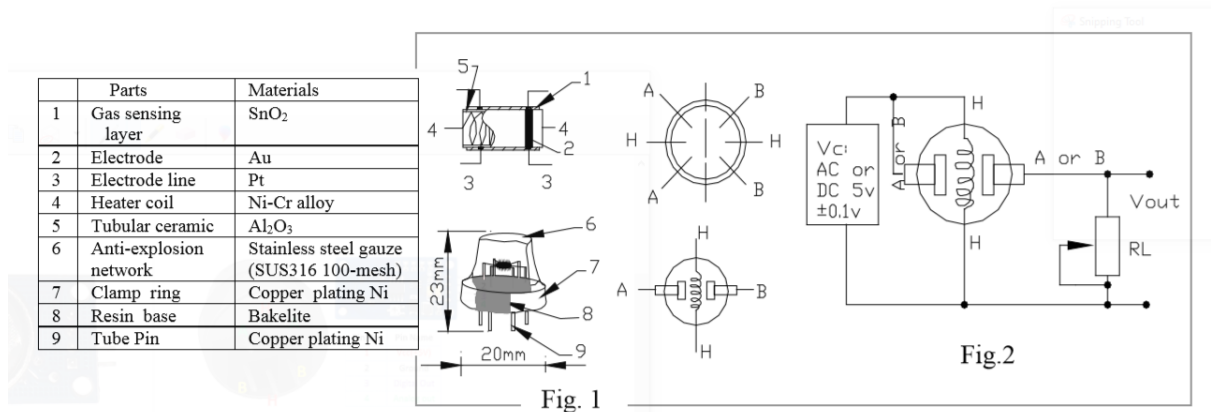
- Environment Condition

Symbol	Parameter name	Technical condition	Remarks
Tao	Using Tem	-10 $\square$ -45 $\square$	
Tas	Storage Tem	-20 $\square$ -70 $\square$	
R <sub>H</sub>	Related humidity	less than 95%Rh	
O <sub>2</sub>	Oxygen concentration	21%(standard condition)Oxygen concentration can affect sensitivity	minimum value is over 2%

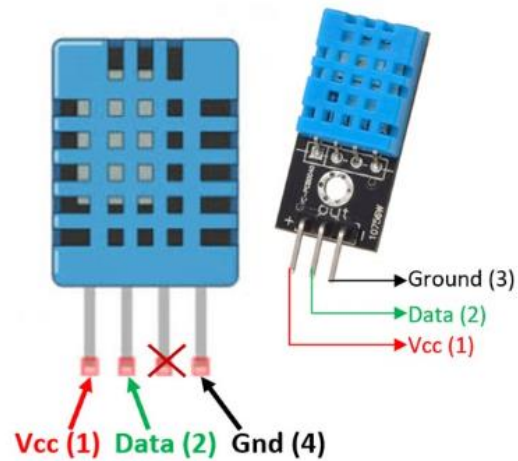
- Sensitivity Charateristic

Symbol	Parameter name	Technical parameter	Ramark 2
Rs	Sensing Resistance	30KΩ-200KΩ (100ppm NH <sub>3</sub> )	Detecting concentration scope 10ppm-300ppm NH <sub>3</sub> 10ppm-1000ppm Benzene 10ppm-300ppm Alcohol
α (200/50) NH <sub>3</sub>	Concentration Slope rate	≤0.65	
Standard Detecting Condition	Temp: 20±2℃ Vc:5V±0.1 Humidity: 65%±5% Vh: 5V±0.1		
Preheat time	Over 24 hour		

- Structure and configuration, basic measuring circuit

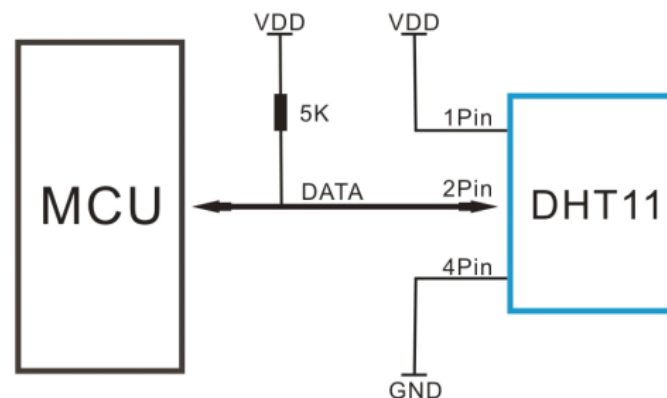


## 2.3. DHT11



No	Pin Name	Deskripsi
	For Sensor	

1	Vcc	Power supply dengan tegangan 3.5V hingga 5.5V
2	Data	Nilai output dari Temperature dan Humidity dengan data serial
3	NC	Tidak ada koneksi sehingga tidak digunakan
4	Ground	Terhubung ke ground rangkaian
<b>For module</b>		
1	Vcc	Power supply dengan tegangan 3.5V hingga 5.5V
2	Data	Nilai output dari Temperature dan Humidity dengan data serial
3	Ground	Terhubung ke ground rangkaian



Ketika kabel penghubung antara MCU dan DHT11 lebih pendek dari 20 meter maka disarankan menggunakan resistor pull-up 5K, sebaliknya ketika kabel penghubung antara MCU dan DHT11 lebih panjang dari 20 meter, penggunaan resistor pull-up perlu disesuaikan dengan kebutuhan.

### 2.3.1. Power dan Pin

Ketika daya disuplai ke sensor, tidak disarankan untuk mengirim apa pun instruksi ke sensor dalam waktu satu detik agar dapat melewati status tidak stabil. Satu kapasitor bernilai 100nF dapat ditambahkan di antara VDD dan GND untuk power filtering.

### 2.3.2. Spesifikasi

Parameters	Conditions	Minimum	Typical	Maximum
<b>Humidity</b>				
Resolution		1%RH	1%RH	1%RH
			8 Bit	
Repeatability			± 1%RH	
Accuracy	25°C		± 4%RH	
	0-50°C			± 5%RH
Interchangeability	Fully Interchangeable			
Measurement Range	0°C	30%RH		90%RH
	25°C	20%RH		90%RH
	50°C	20%RH		80%RH
Response Time (Seconds)	1/e(63%)25°C, 1m/s Air	6 S	10 S	15 S
Hysteresis			± 1%RH	
Long-Term Stability	Typical		± 1%RH/year	
Temperature				
Resolution		1°C	1°C	1°C
		8 Bit	8 Bit	8 Bit
Repeatability			± 1°C	
Accuracy		± 1°C		± 2°C
Measurement Range		0°C		50°C
Response Time (Seconds)	1/e(63%)	6 S		30 S

### 2.3.3. Karakteristik Elektrik

	Conditions	Minimum	Typical	Maximum
Power Supply	DC	3V	5V	5.5V
Current Supply	Measuring	0.5mA		2.5mA
	Average	0.2mA		1mA
	Standby	100uA		150uA
Sampling period	Second	1		

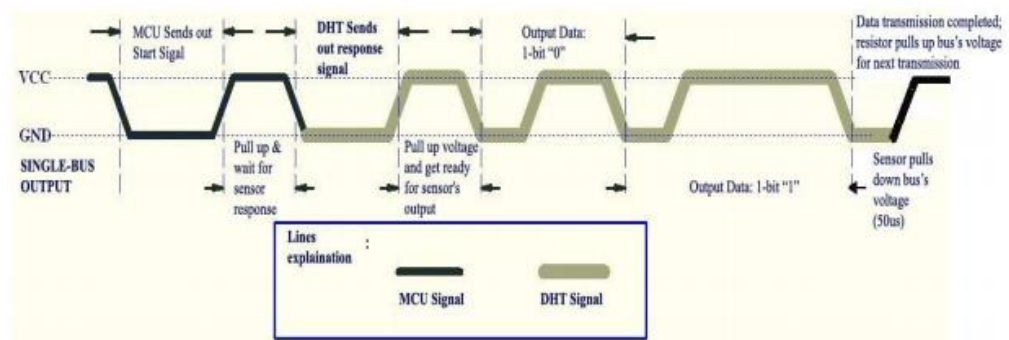
### 2.3.4. Komunikasi MCU dan DHT11 (Serial Interfcae: Single-Wire Two-Way)

Format data bus tunggal digunakan untuk komunikasi dan sinkronisasi antara sensor MCU dan DHT11. Satu proses komunikasi membutuhkan waktu sekitar 4ms. Data terdiri dari bagian desimal dan integral. Transmisi data lengkap adalah 40bit, dan sensor mengirimkan bit data yang lebih tinggi terlebih dahulu. Format data: data RH integral 8bit + data RH desimal 8bit + data T integral 8bit + data T desimal 8bit + jumlah cek 8bit Jika transmisi data benar,

check-sum harus menjadi 8bit terakhir dari "data RH 8bit integral + data RH desimal 8bit + data T integral 8bit + data T desimal 8 bit".

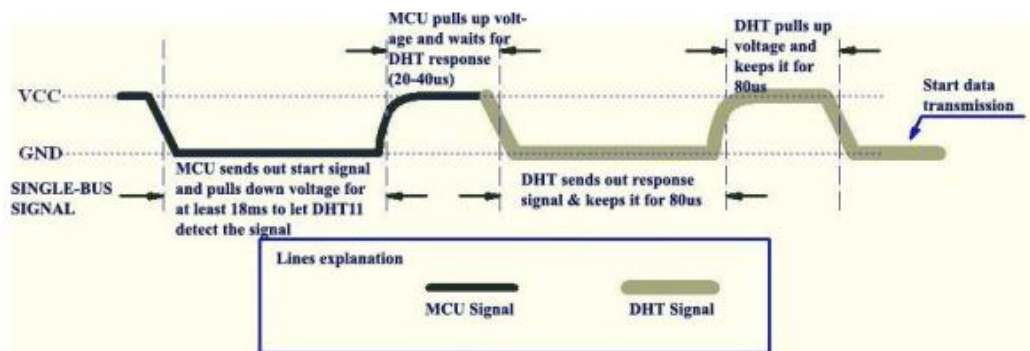
### 2.3.5. Keseluruhan Proses Komunikasi

Ketika MCU mengirimkan sinyal mulai, DHT11 berubah dari mode konsumsi daya rendah ke mode berjalan, menunggu MCU menyelesaikan sinyal mulai. Setelah selesai, DHT11 mengirimkan sinyal respons data 40-bit yang mencakup informasi kelembaban dan suhu relatif ke MCU. Pengguna dapat memilih untuk mengumpulkan (membaca) beberapa data. Tanpa sinyal start dari MCU, DHT11 tidak akan memberikan sinyal respons ke MCU. Setelah data dikumpulkan, DHT11 akan berubah ke mode konsumsi daya rendah hingga menerima sinyal awal dari MCU lagi.



### 2.3.6. MCU Mengirim Sinyal Start ke DHT

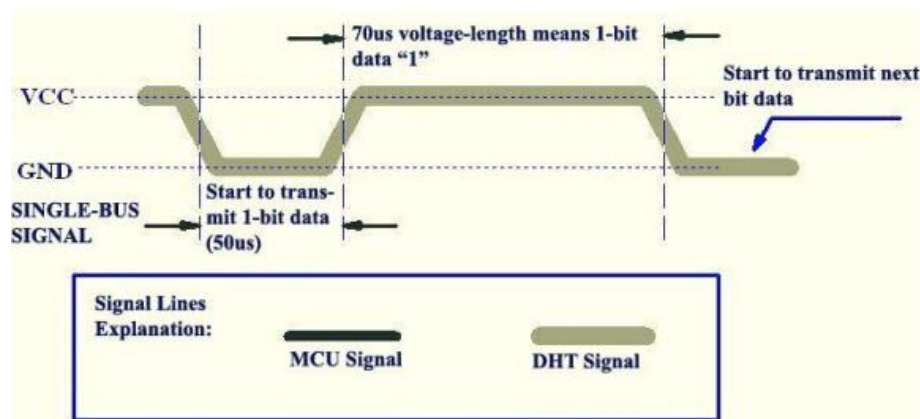
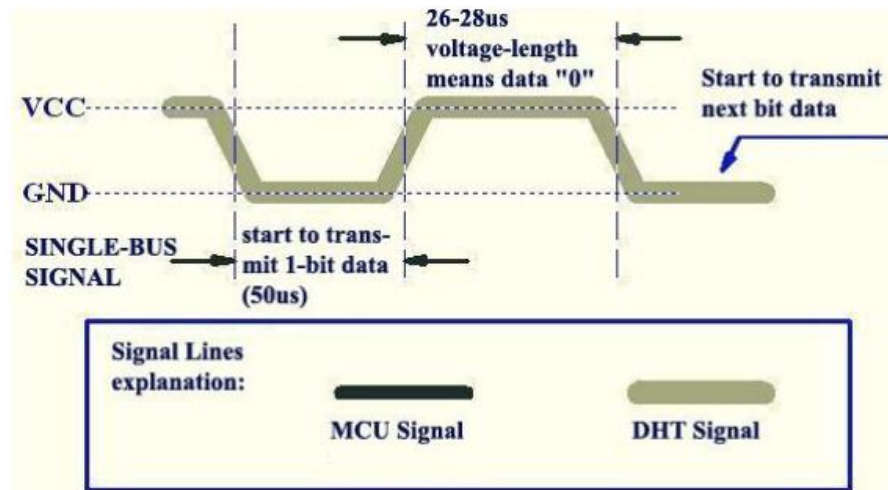
Data Status bebas bus tunggal berada pada level tegangan tinggi. Ketika komunikasi antara MCU dan DHT11 dimulai, program MCU akan mengatur level tegangan bus data tunggal dari tinggi ke rendah dan proses ini harus memakan waktu setidaknya 18 ms untuk memastikan deteksi DHT dari sinyal MCU, maka MCU akan menarik tegangan dan menunggu 20 -40µs untuk respons DHT.



### 2.3.7. DHT Response ke MCU

Begitu DHT mendeteksi sinyal start, itu akan mengirimkan sinyal respons level tegangan rendah, yang bertahan 80us. Kemudian program DHT menetapkan level tegangan bus data Tunggal dari rendah ke tinggi dan menyimpannya untuk 80 us untuk persiapan DHT untuk mengirim data. Ketika DATA Single-Bus berada pada level tegangan rendah, ini berarti bahwa DHT mengirimkan respons sinyal. Begitu DHT mengirimkan sinyal respons, ia menarik tegangan dan menyimpannya untuk 80 kita dan bersiap untuk transmisi data.

Ketika DHT mengirim data ke MCU, setiap bit data dimulai dengan level tegangan rendah 50us dan panjang sinyal level tegangan tinggi berikut menentukan apakah bit data adalah "0" atau "1" (seperti gambar di bawah).

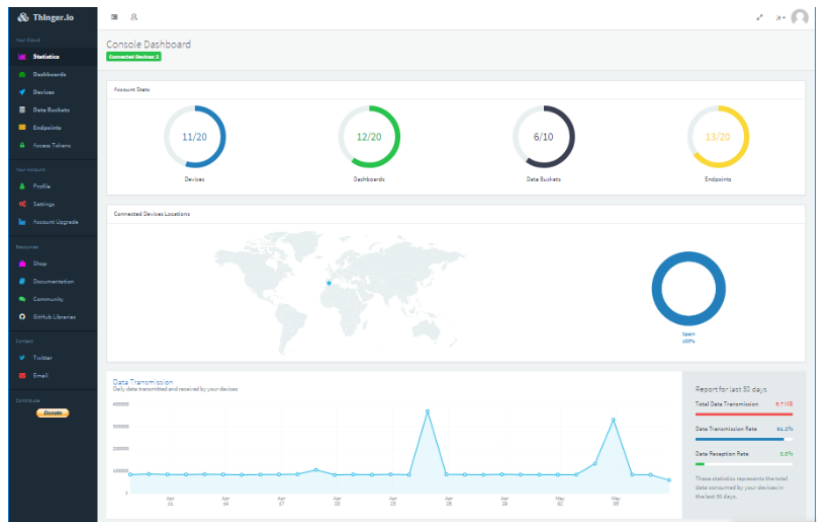


Jika sinyal respons dari DHT selalu berada pada level tegangan tinggi, ini menunjukkan bahwa DHT tidak merespons dengan baik dan periksa koneksi. Ketika bit data terakhir ditransmisikan, DHT11 menarik turun level



tegangan dan menyimpannya untuk 50us. Kemudian tegangan Single-Bus akan ditarik oleh resistor untuk mengaturnya kembali ke status bebas.

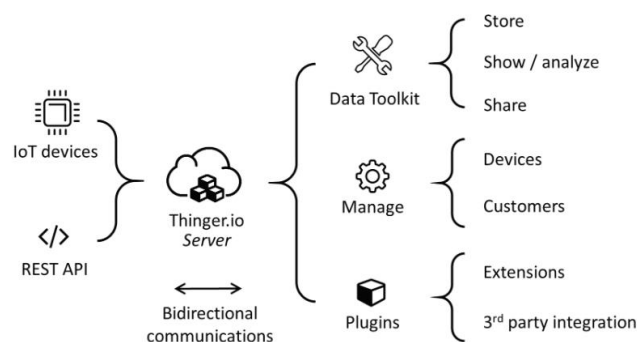
## 2.4. Spesifikasi Platform Thinger



Thinger merupakan suatu platform Open-Source dari Internet of Things (IoT) cloud yang memberikan semua tool yang dibutuhkan untuk membuat suatu prototype, scale, dan mengelola produk-produk yang akan dihubungkan melalui suatu server melalui langkah-langkah yang sederhana. Tujuan dari dibuatnya platform ini adalah untuk membantu menyederhanakan pengembangan proyek IOT dan membuat sesuatu yang dapat diakses oleh seluruh dunia melalui internet.

- Free IoT Platform
- Simple dan Powerful
- Hardware agnostic
- Open-Source
- Customizable

### 2.4.1. Fitur utama Platform Thinger



- Menghubungkan banyak perangkat.
- Komunikasi efisien secara bidireksional.
- Penyimpanan data perangkat.
- Menampilkan data secara real-time.
- Cloud Computing.
- Terintegrasi dengan pihak ketiga.

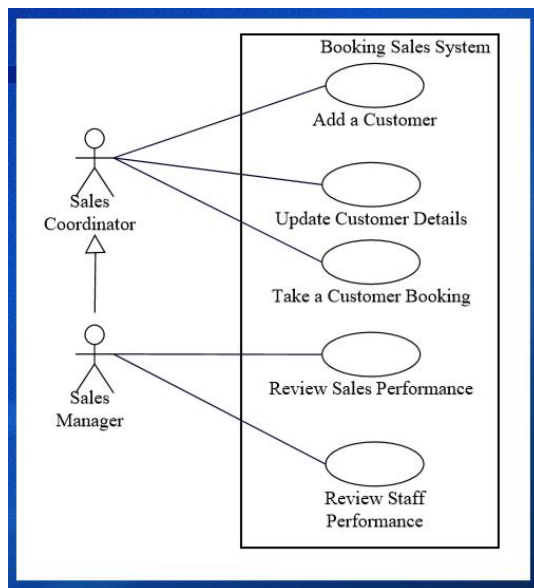
## 2.5. UML (Unified Modelling Language)

UML (Unified Modelling Language) merupakan suatu bahasa pemodelan yang terdiri atas rangkaian diagram yang bertujuan untuk membantu pengembangan suatu sistem baik dari sisi perangkat keras maupun perangkat lunaknya. Jadi, UML ini bertujuan untuk membantu pengembangan suatu sistem sehingga mudah dipahami dan dipelajari serta memberikan gambaran sistem secara garis besar. Berikut hal-hal yang direpresentasikan oleh UML :

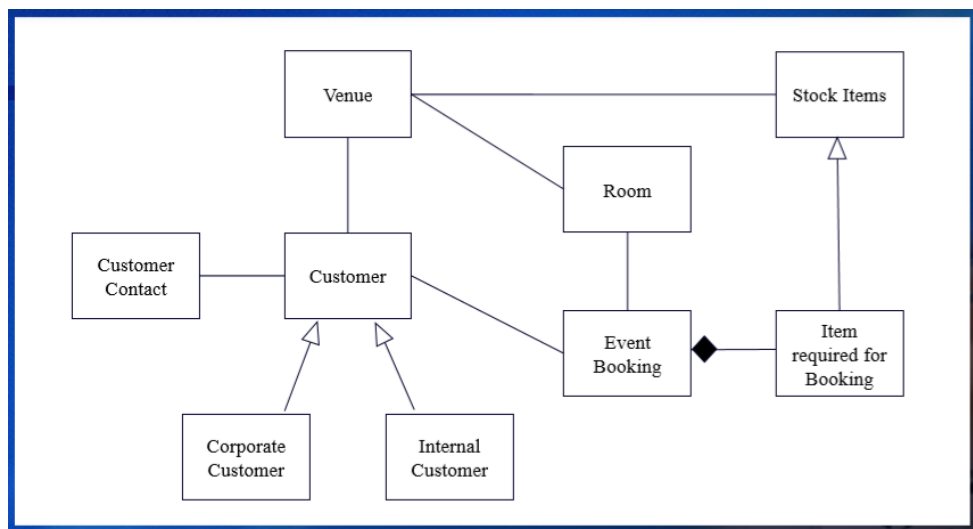
- Specification
- Architecture design
- Simulation and Testing
- Visualization
- Documentation
- Construction

UML diagram terdiri dari beberapa jenis yang masing-masing jenis tersebut berbeda-beda dan menunjukkan aspek sistem yang dibuat. Penggunaan UML juga memungkinkan untuk membuat suatu diagram dengan tampilan khusus dengan membatasi elemen diagram untuk tujuan tertentu pada waktu tertentu. Berikut merupakan beberapa jenis diagram UML :

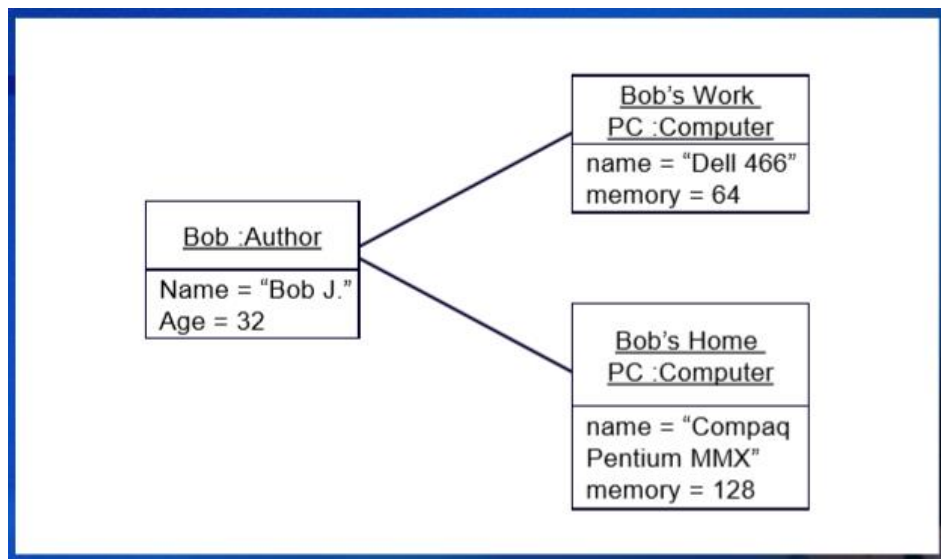
- User Case Diagrams



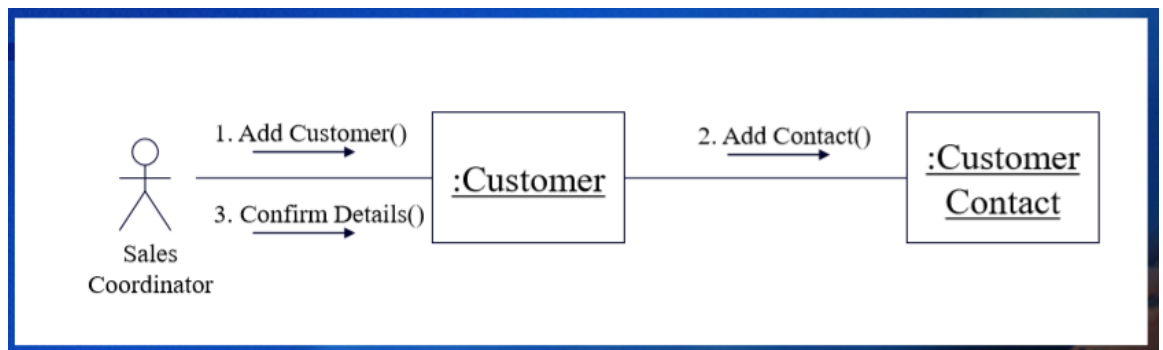
- Class Diagrams



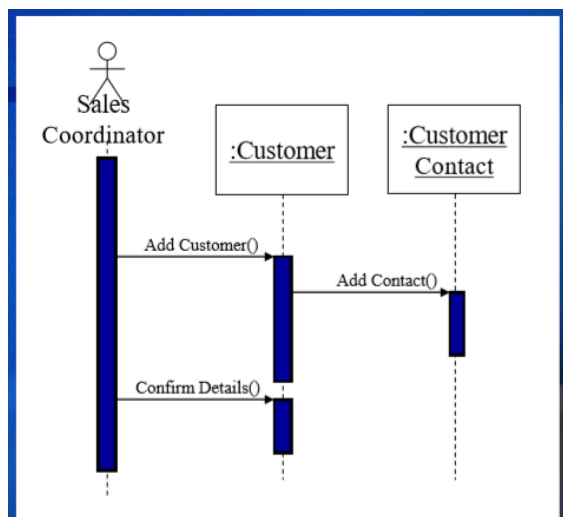
- Object Diagrams



- Collaboration Diagrams



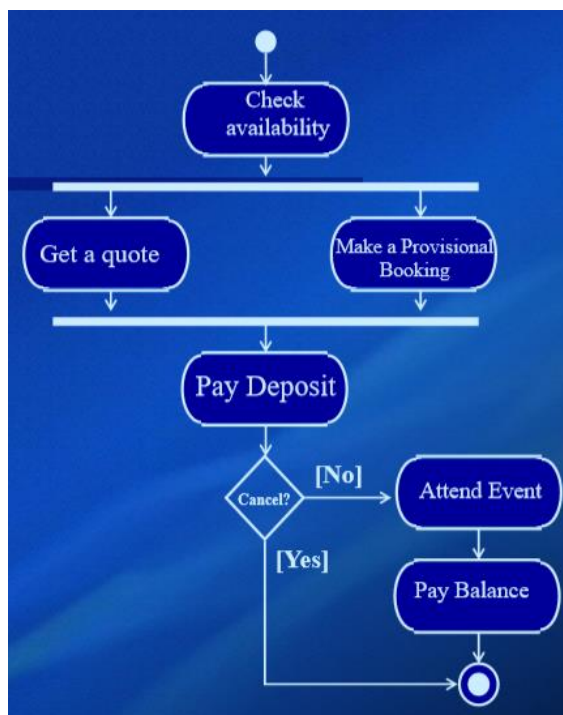
- Sequence Diagrams



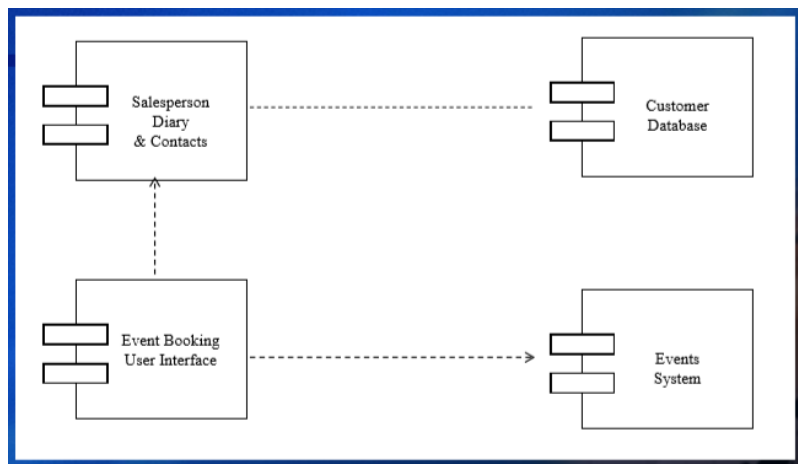
- State Diagrams



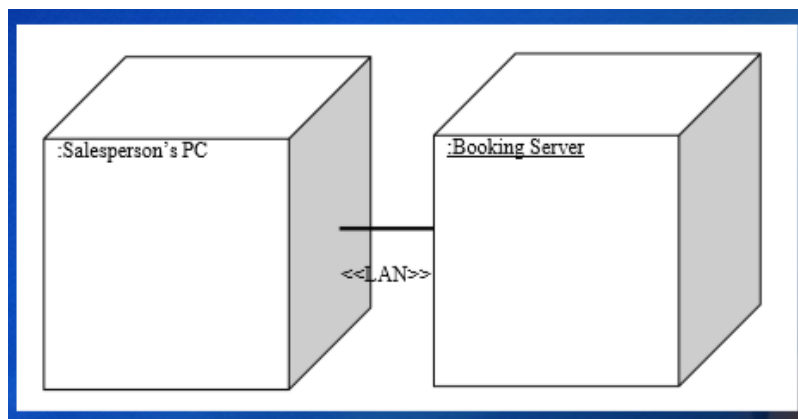
- Activity Diagrams



- Component Diagrams



- Deployment Diagrams



## 2.6. Bahasa Pemrograman dan Software Compiler

Bahasa pemrograman yang digunakan pada perangkat pemantau kualitas udara pada proyek ini adalah bahasa pemrograman Arduino dengan menggunakan software Arduino IDE (Integrated Development Environment). Software ini dapat digunakan sebagai media untuk melakukan pemrograman baik di Arduino maupun di Nodemcu seperti esp8266 yang digunakan pada proyek ini. Software ini berupa text editor untuk melakukan pembuatan code, melakukan edit, melakukan validasi terhadap code yang dibuat, dan untuk melakukan upload code ke board perangkat esp8266. Karena pada esp8266 ini menggunakan pemrograman Arduino maka source code yang digunakan diberi nama Sketch Arduino.

### 2.6.1. Sketch Arduino

Pada sketch Arduino bahasa yang digunakan sama dengan bahasa C/C++, tetapi dibagi menjadi 3 bagian utama yaitu structure, values, dan function.

- a. Structure, pada bagian ini berisi fungsi setup() dan loop
  - o Setup ()  
Fungsi ini hanya dijalankan sekali ketika board dinyalakan atau direset dan digunakan sebagai tempat inisialisasi pin mode, variabel, dan lainnya.
  - o Loop()  
Fungsi yang diletakkan dan dieksekusi setelah fungsi Setup() serta dijalankan secara berulang-ulang dan digunakan sebagai pengontrol board yang digunakan.
- b. Values  
Pada bagian ini berisi variable dan konstanta.
- c. Function  
Function merupakan suatu segmentasi code yang berisi tugas-tugas tertentu dan terdefinisi. Function tersebut dibutuhkan ketika programmer akan melakukan tindakan yang sama beberapa kali dalam suatu program. Selain itu function disebut juga dengan modular code.

## 2.7. Parameter Kesehatan Udara

Berikut merupakan parameter kesehatan udara berdasarkan tingkat konsentrasi CO<sub>2</sub> dalam PPM, suhu dalam Celcius (C) , dan kelembapan udara dalam persen (%).

- CO<sub>2</sub> dalam PPM



- Kelembapan dan Suhu

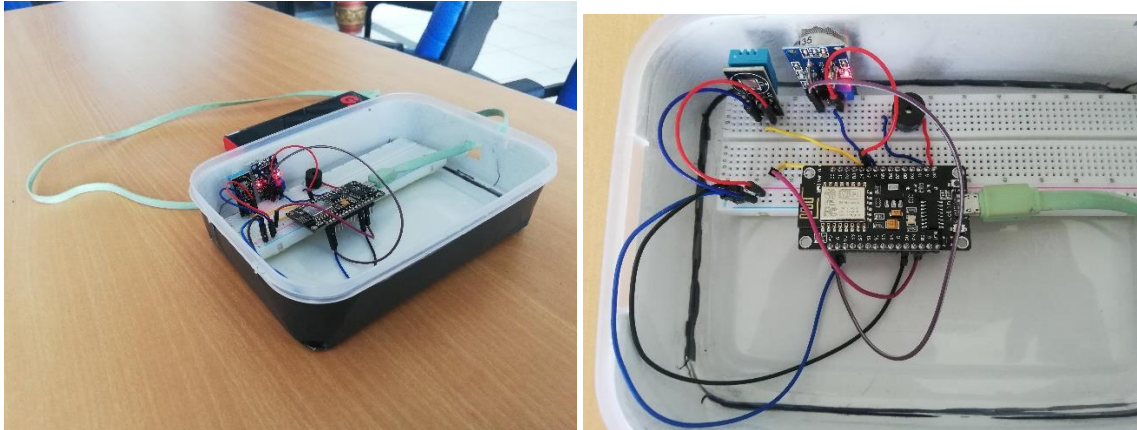
Parameter	Rentang Kualitas Suhu dan Kelembapan dalam Suatu Ruangan
Suhu udara	22,5 °C - 25,5 °C
Kelembaban udara	$\leq 70 \%$



## BAB III

### PEMBAHASAN DAN PERANCANGAN

#### 3.1. Deskripsi Proyek



Proyek pembuatan alat atau perangkat pemantau kualitas udara merupakan suatu perangkat yang digunakan untuk memantau kualitas udara pada suatu tempat yang telah ditentukan. Alat ini akan menampilkan hasil pemantauan berupa data-data secara realtime melalui platform thinger.io yang dapat diakses oleh siapapun. Data-data tersebut berupa tingkat suhu udara dalam CO<sub>2</sub>, tingkat kelembapan udara dalam persen (%), dan menampilkan kadar konsentrasi partikel gas berbahaya yang terkandung dalam udara seperti CO<sub>2</sub>. Selain menampilkan data-data tersebut, perangkat ini juga akan memberikan notifikasi apabila kualitas udara memasuki kategori berbahaya seperti membunyikan buzzer.

Dalam pembuatan perangkat ini membutuhkan beberapa alat, diantaranya esp 8266, DHT11, dan MQ135. Alat-alat tersebut memiliki fungsinya masing-masing terhadap alat yang dibuat, yaitu :

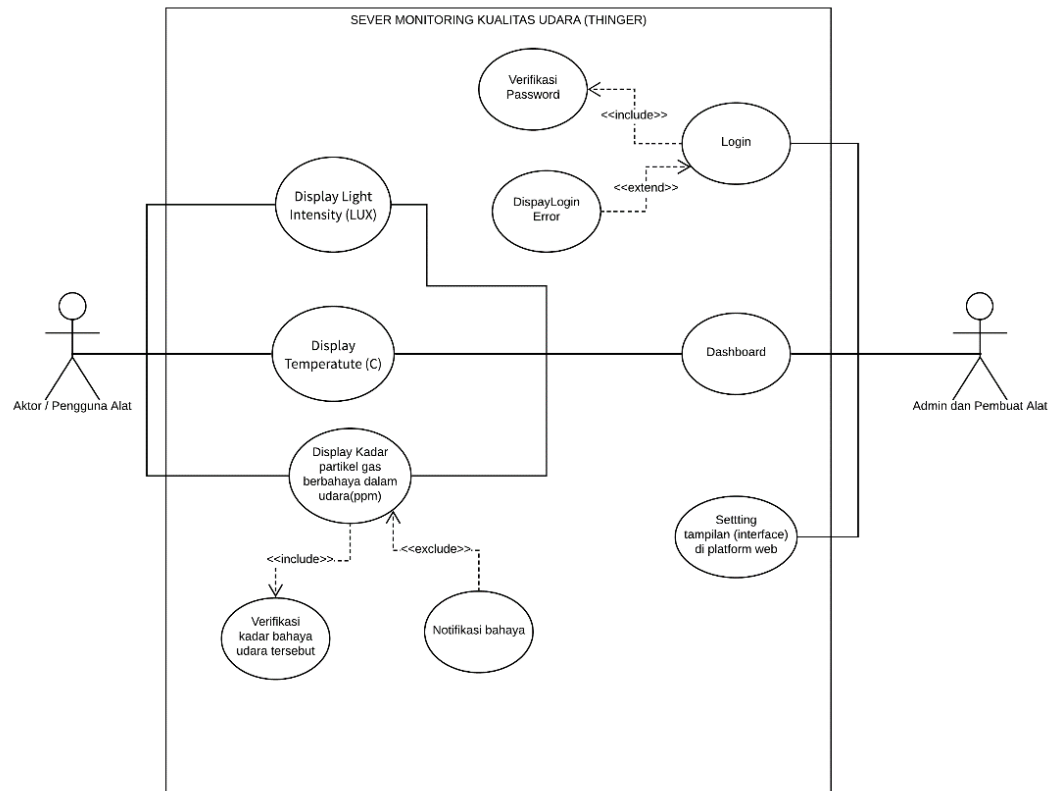
- Esp8266 : sebagai mikrontroller yang dapat terhubung dengan WIFI dan internet sehingga dapat melakukan transfer data-data yang dipantau ke platform thinger.io.
- DHT11 : alat yang digunakan untuk mengukur suhu dan kelembapan .
- MQ135 : alat yang digunakan untuk mengukur kadar partikel gas berbahaya dalam udara, sehingga dalam proyek ini digunakan sebagai alat yang bertujuan untuk mengetahui tingkat kualitas udara.

Alat/perangkat ini diletakkan di Gedung **MRPQ lantai 4** sebagai tempat untuk dipantau tingkat kualitas udaranya.

### 3.2. UML (Unified Modelling Language) Proyek

Berikut beberapa diagram UML berdasarkan proyek pemantauan/monitoring kualitas udara yang dibuat :

#### 3.2.1. Use Case Diagram



Diatas adalah diagram use case berdasarkan proyek pemantuan/monitoring kualitas udara yang kami buat. Pada diagram ini digambarkan bahwa terdapat 2 jenis peran yang terlibat yaitu peran aktor/pengguna alat dan peran admin/pembuat alat. Berikut penjelasan tentang masing-masing perannya :

- Pada sisi admin/pembuat alat dapat melakukan login pada website thinger melalui akunnya untuk melakukan pemantuan data-data yang diperoleh seperti data intensitas cahaya (LUX), temperature (C), serta kadar partikel gas dalam udara. Semua data-data tersebut dapat dipantu atau dilihat melalui menu dashboard. Admin/pembuat alat dapat melakukan setting/pengaturan tampilan (User Interface), seperti pengaturan tampilan pada dashboard untuk data-data apa saja yang ingin ditampilkan, bentuk tampilanya, dan lainnya.

- Pada sisi aktor/pengguna alat hanya dapat melihat data-data hasil pemantauan kualitas udara melalui dashboard, seperti data intensitas cahaya (LUX), temperature (C), dan kadar partikel gas dalam udara.

Pada diagram tersebut juga diketahui bahwa akan ada notifikasi peringatan apabila kadar gas berbahaya dalam udara telah mencapai level yang berbahaya. Notifikasi tersebut dapat diketahui oleh admin/pembuat alat dan juga user/pengguna alat.

### 3.2.2. Sequential Diagram

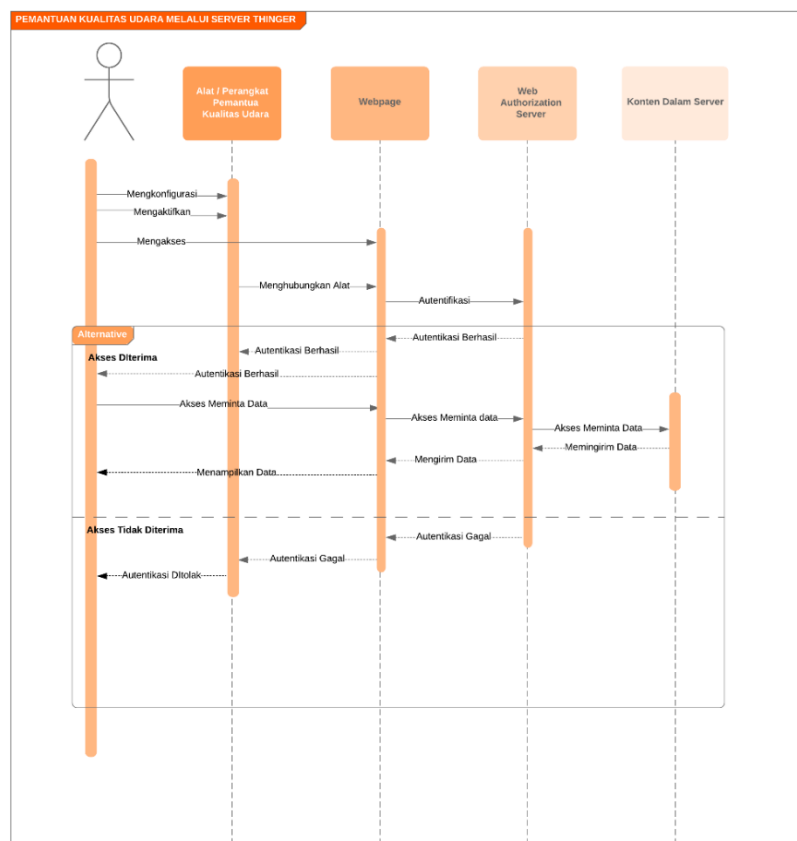


Diagram sequensial menggambarkan hubungan antar komponen berdasarkan proses yang berjalan secara sequensial. Diatas adalah diagram sequensial dari alat/perangkat pemantau/monitoring kualitas udara. Diagram diatas menginformasikan bagaimana suatu alat dapat terkoneksi dengan webpage dan bagaimana admin alat pemantau kualitas udara memperoleh informasi berupa data-data hasil pemantauan dari server. Berikut proses sequensial yang terjadi :

- Proses 1

Admin melakukan konfigurasi terhadap alat/perangkat pemantau kualitas udara dan kemudian mengaktifkannya.

- Proses 2

Admin mengakses webpage platform yaitu thinger.io, kemudian login. Perangkat/alat tersebut juga harus dihubungkan ke webpage dengan memasukkan konfigurasi-konfigurasi yang diperlukan seperti konfigurasi token yang diminta.

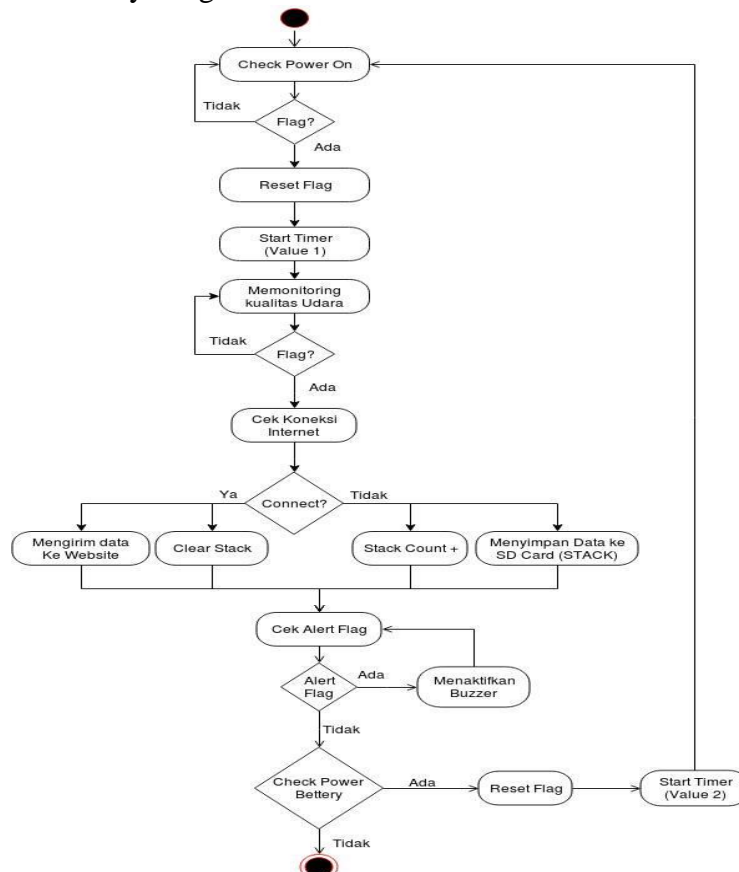
- Proses 3

Dilakukannya proses autentikasi pada Web Authorization Server.

- Proses 4

Pada proses 4 ini terjadi 2 kemungkinan yaitu proses autentikasi berhasil atau tidak berhasil. Jika proses autentikasi berhasil maka data berupa informasi hasil pemantauan kualitas udara akan dapat diakses oleh Admin/pembuat alat dan juga user/pengguna alat. Namun, jika proses autentikasi tidak berhasil maka data berupa informasi hasil pemantauan kualitas udara tidak dapat diakses.

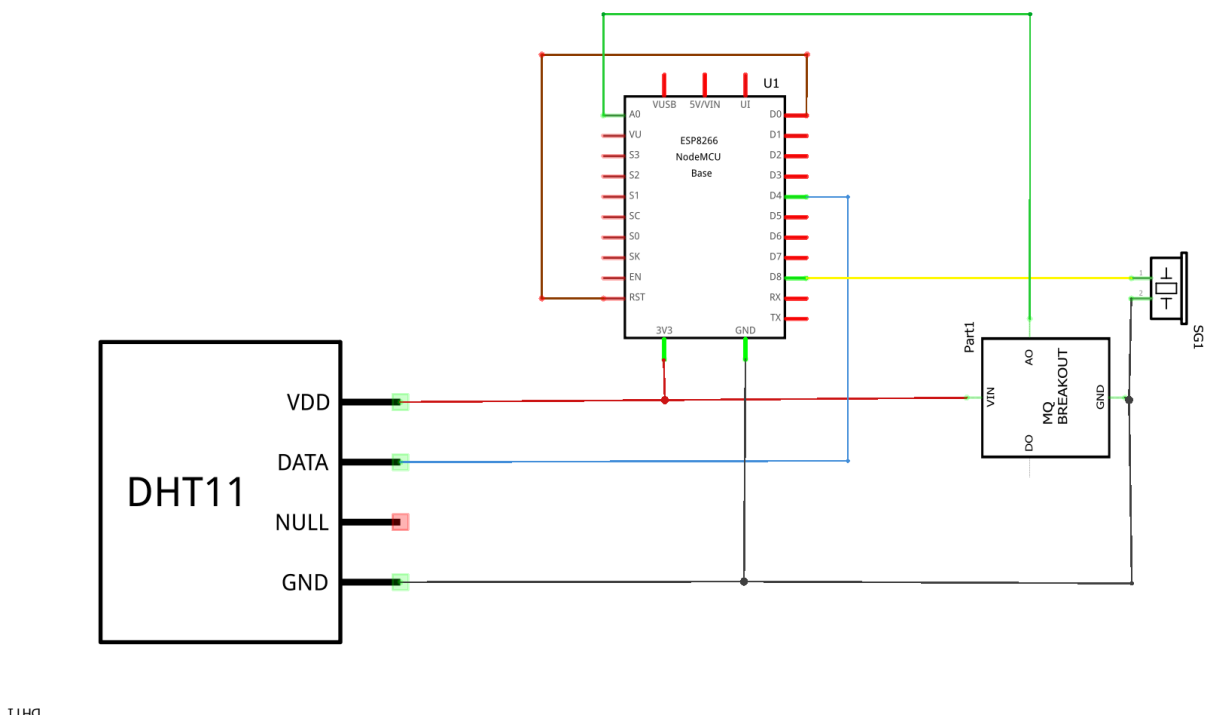
### 3.2.3. Activity Diagram



Activity diagram atau activity behaviaoral yaitu diagram yang menggambarkan alur kegiatan pada suatu system. Diagram ini berfokus pada kondisi bagaimana alur dan sequence pada system.

Pada gambar di atas, terlihat workflow system diawali dengan mengecek kondisi power, lalu masuk ke decision state flag jika tidak ada maka kembali mengecek power, jika ada maka reset flag (hibernate mode off). Selanjutnya, akan memulai timer lalu monitoring kondisi udara selama flag tidak ada. Setelah ada flag, maka dilanjutkan menegcek koneksi internet jika tidak ada maka data akan disimpan di SD Card dan Stack Count diincrement, sebaliknya jika ada maka stack count akan di clear dan semua data dikirim ke server. Lalu, mengecek alert flag, jika ada maka akan buzzer akan aktif selama flag ada, sebaliknya jika tidak ada maka akan mengecek power battery jika tidak ada (habis) akan off/berakhir, sebaliknya jika ada maka akan mereset flag, lalu memulai timer (masuk mode hibernasi), dan loop.

### 3.3. Schematic / Wiring Diagram Alat



DHT11

fritzing

- Kabel merah : VCC dengan nilai 3,3 Volt atau 5 Volt
- Kabel hitam : Ground
- Warna lain : pin input output

- Device yang digunakan :

esp8266, MQ135, DHT11, Buzzer, dan beberapa kabel jumper yang dibutuhkan baik jenis Female maupun Male.

Tabel dibawah merupakan penjelasan dari beberapa pin penting yang saling dihubungkan.

PERANGKAT 1 - Pin	PERANGKAT 2 - Pin	Penjelasan
ES8266 – A0	MQ135 – A0	Pin A0 MQ135 ini berfungsi untuk menghasilkan tegangan analog 0-5V berdasarkan intensitas gas CO <sub>2</sub> pada ruangan tersebut, sehingga output dapat berupa ppm.
ES8266 – A0	DHT11 – OUT/DATA	Untuk mendapatkan nilai output suhu dan kelembapan.
ES8266 – RST	ES8266 – D0	Pin D0 atau GPIO16 hanya dapat digunakan untuk read dan write, sehingga dapat dimanfaatkan untuk membuat fitur deep sleep dan wake up otomatis. Pada mode deep sleep ini item-item seperti Wi-Fi, System Clock, dan CPU akan mati (OFF). Namun, item RTC (Real Time Clock) akan tetap hidup (ON) sehingga dapat mengaktifkan kembali perangkat secara otomatis berdasarkan waktu yang telah ditentukan.
ES8266 – D8	Buzzer – pin positif	Untuk membunyikan buzzer berdasarkan keadaan tertentu.

### 3.4. Fitur-fitur Pada Platform yang Dimanfaatkan

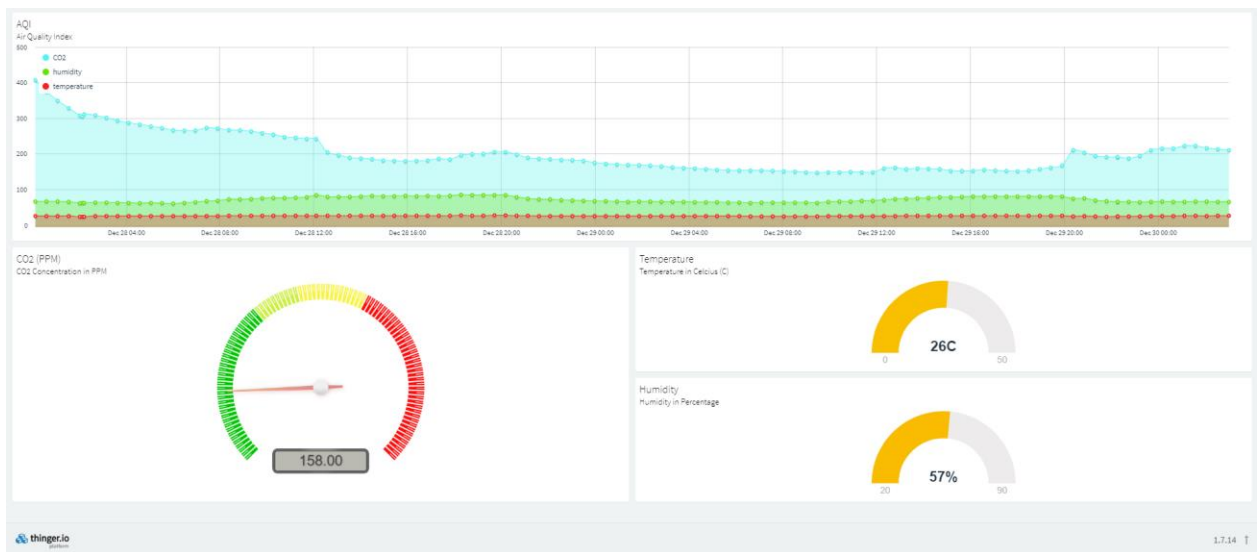
Pada proyek embedded sistem ini yang berkaitan dengan pemantaun kualitas udara disekitar lingkungan Fakultas Teknik, mengimplementasikan beberapa fitur yang ada pada platform Thinger.io. Berikut beberapa fitur yang diimplementasikan :

- Fitur pemantauan atau monitoring kualitas udara secara realtime dan online.
- Data-data hasil pemantuan dapat diakses oleh siapapun.

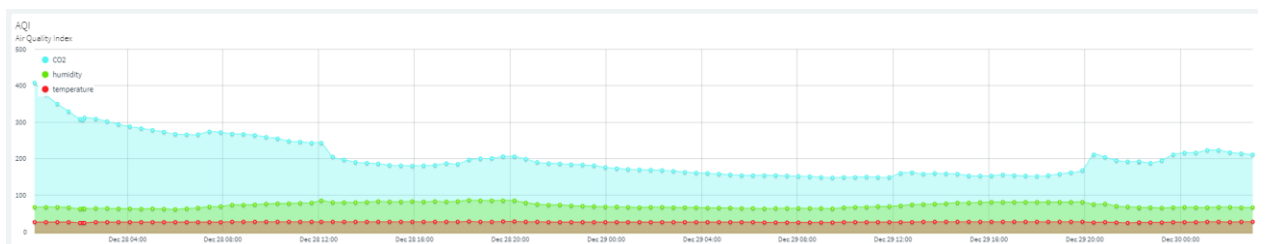
- Data-data yang dipantau yaitu tingkat kelembapan, tingkat suhu dan konsentrasi partikel gas berbahaya yang terkandung dalam udara seperti CO<sub>2</sub>.
- Adanya notifikasi jika kualitas udara masuk kedalam kategori berbahaya.

#### 3.4.1. Akses Dashboard Perangkat Monitoring Kualitas Udara

- Website : [bit.ly/AirMonitor](http://bit.ly/AirMonitor)
- Tampilan keseluruhan dashboard

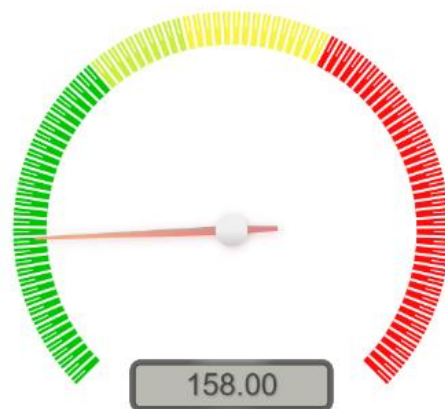


- Grafik pemantauan kualitas CO<sub>2</sub>, kelembapan, dan suhu berdasarkan periode tertentu



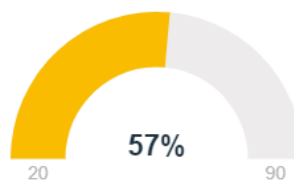
- Pemantauan Real-time CO2

CO2 (PPM)  
CO2 Concentration in PPM



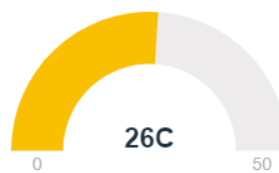
- Kelembapan

Humidity  
Humidity in Percentage



- Suhu

Temperature  
Temperature in Celcius (C)





### 3.5. Code / Sketch Perangkat Monitoring Kualitas Udara

```
#define _DISABLE_TLS_

#include <ESP8266WiFi.h>

#include <ThingySmartConfig.h>

#include <Wire.h>

#include <MQ135.h>

#include <dht.h>

#define _DEBUG_

/* PIN D0 = GPIO16 & D8 = GPIO15 */

static const uint8_t D0 = 16;

static const uint8_t D8 = 15;

//WiFi credentials

#define SSID "NEST UI 2018-powered by ICON+"

#define PASSWORD ""

//thingy.io credentials

#define USERNAME "mkamal"

#define DEVICE_ID "1111"

#define DEVICE_CREDENTIAL "mkamal"

ThingySmartConfig thing(USERNAME, DEVICE_ID, DEVICE_CREDENTIAL);

#define RLOAD 22.0

#define RZERO 879.13

float ppmCO2;

void co2Sensor();

int sensorValue;

MQ135 gasSensor = MQ135(A0);

//#define dht_type DHT11

#define dht_pin 2

dht DHT;

//DHT dht(dht_pin, dht_type);
```

```

float humidity;

float temperature;

void dhtSensor();

void alert();

const int buzzer = D8;

void setup() {
    // pinMode(buzzer, OUTPUT);

    Serial.begin(9600);

    Serial.setTimeout(2000);

    while (!Serial) { }

    Serial.println("Device Activated.");

    /* lightSensor */

    Wire.begin();

    DHT.read11(dht_pin);

    analogRead(0);

    WiFi.begin(SSID, PASSWORD);

    if (WiFi.status() != WL_CONNECTED) {
        delay(10000);
        Serial.print(".");
    }

    Serial.println("Device Connected!");

    thing["AQI"] >> [](pson & out) {
        out["CO2"] = analogRead(0);

        out["temperature"] = DHT.temperature;

        out["humidity"] = DHT.humidity;
    };

    co2Sensor();

    dhtSensor();

    thing.handle();

    thing.write_bucket("1", "AQI");

    delay(15000);

    WiFi.disconnect();

    if (sensorValue > 600 || temperature > 45 || humidity >= 90){
        Serial.println("Danger");

        alert();
    }
}

```

```
void loop() {  
    //Dikosongkan  
}  
  
void co2Sensor() {  
    sensorValue = analogRead(0);    // read analog input pin 0  
    Serial.print("AirQua=");  
    Serial.print(sensorValue, DEC); // prints the value read  
    Serial.println(" PPM");  
}  
  
void dhtSensor() {  
    DHT.read11(dht_pin);  
    humidity = DHT.humidity;  
    Serial.print("Humidity : ");  
    Serial.println(humidity);  
    temperature = DHT.temperature;  
    Serial.print("Temperature : ");  
    Serial.println(temperature);  
}  
  
void alert(){  
    tone(buzzer, 900);  
    delay (1000);  
    noTone(buzzer);  
    delay (500);  
    tone(buzzer, 900);  
    delay (1000);  
    noTone(buzzer);  
}
```

## **BAB IV**

### **PENUTUP**

#### **4.1. Saran**

- Membuat suatu alat atau perangkat yang berguna bagi kehidupan manusia terutama dibidang yang berkaitan dengan kesehatan.
- Memanfaatkan segala sumber yang ada untuk membantu dalam pengerjaan proyek.
- Lebih memahami tentang esp8266 dan sensor-sensor yang digunakan pada proyek ini.

#### **4.2. Kesimpulan**

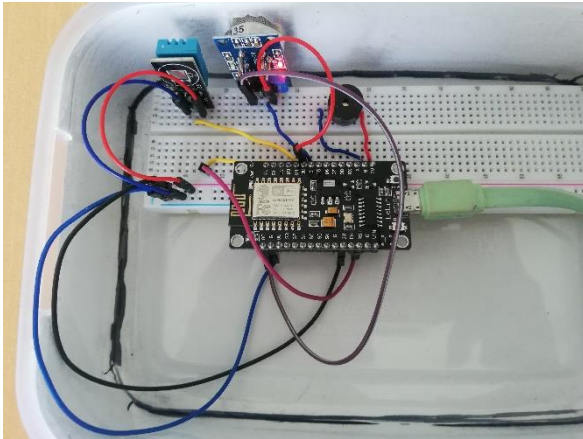
Proyek pembuatan perangkat pemantauan atau monitoring kualitas udara sangat dibutuhkan saat ini, karena tingginya polusi udara dilingkungan sekitar. Oleh karena itu proyek kami membuat suatu perangkat atau alat yang mampu mengukur kualitas udara secara realtime melalui suatu platform thinger.io. Perangkat yang dibuat ini memanfaatkan teknologi IoT (Internet of Things), sehingga data-data yang didapatkan dapat diakses kapanpun dan dimanapun.

Perangkat yang dibuat ini memanfaatkan esp8266 yang digunakan sebagai mikrokontroller yang dapat terhubung ke internet dan sensor-sensor yang dibutuhkan DHT11 dan MQ135. Masing-masing dari sensor-sensor tersebut memiliki fungsinya masing masing, seperti DHT11 untuk memantau tingkat kelembapan dan suhu udara, serta MQ135 sebagai penentu kualitas udara berdasarkan tingkat konsentrasi dan jenis partikel gas yang terkandung dalam udara.

## DAFTAR PUSTAKA

- User Manual V1.2 ESP8266 NodeMCU WiFi Devkit, Handson Technology  
[[www.handsontec.com](http://www.handsontec.com)]
- User Manual DHT11 Humidity & Temperature Sensor, D-Robotics UK  
[[www.droboticsonline.com](http://www.droboticsonline.com)]
- "MQ-135 - Gas Sensor for Air Quality", components101, 27 February 2018, [Online]  
Tersedia :<https://components101.com/sensors/mq135-gas-sensor-for-air-quality>  
[Diakses : 21 Desember 2019]
- Craig Larman - Applying UML and Patterns\_ An Introduction to Object-Oriented Analysis and Design and the Unified Process (2001, Prentice Hall PTR)K. Elissa, "Title of paper if known," unpublished.
- "Kualitas Udara", BADAN METEOROLOGI, KLIMATOLOGI, DAN GEOFISIKA, 2 Januari 2020, [Online]  
Tersedia :<https://www.bmkg.go.id/kualitas-udara/informasi-partikulat-pm10.bmkg> [ Diakses : 27 Desember 2019]
- Kurniawan, Agusta "PENGUKURAN PARAMETER KUALITAS UDARA (CO, NO2, SO2, O3 DAN PM10) DI BUKIT KOTOTABANG BERBASIS ISPU", Jurnal Tekno Sains, vol. 7, no. 1, pp. 1-82, 22 Desember 2017.

## LAMPIRAN



Alat / Perangkat



Lokasi/tempat alat diletakkan



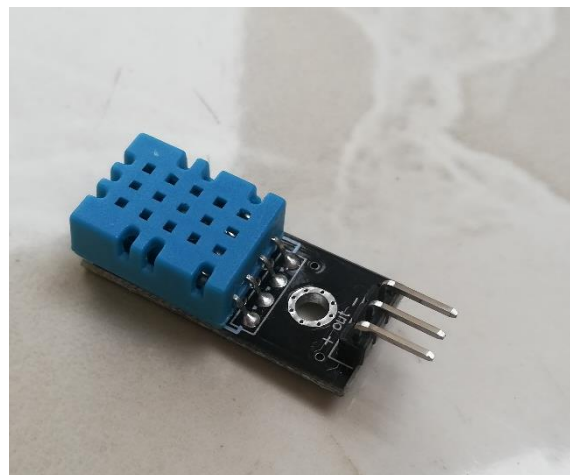
Alat / Perangkat



Esp8266



MQ135



DHT11