

BAB III

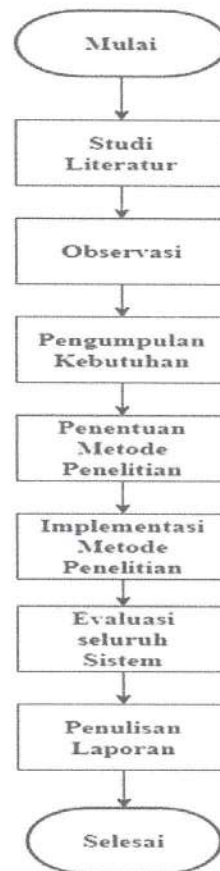
METODOLOGI PENELITIAN

3.1 Subyek Dan Obyek Penelitian

Yang dimaksud dengan subyek penelitian adalah orang, benda, atau tempat yang diamati dalam rangka pembuatan sebagai sasaran. Subyek dari penelitian ini adalah untuk memonitoring kualitas udara dari lingkungan sekitar. Dan yang dimaksud dengan obyek penelitian adalah hal yang menjadi sasaran penelitian. Obyek pada penelitian ini adalah bagaimana cara membuat perangkat dan sitem Alat Monitoring Polusi Udara Berbasis *Internet of Things* (IOT).

3.2 Diagram Alur Penelitian

Pada metodologi penelitian, terdapat tahapan yang dilakukan peneliti sehingga dapat menyelesaikan penelitian. Dalam tahapan penelitian ini disusun secara sistematis guna memudahkan peneliti dalam mencapai tujuan penelitian. Penelitian ini diawali dari studi literatur, observasi, pengumpulan kebutuhan, menentukan metode penelitian, implementasi metode penelitian, evaluasi seluruh sistem, dan penulisan laporan. Adapun gambar dari diagram penelitian dapat di lihat di bawah ini:



Gambar 3.1 Diagram Alur Penelitian

3.2.1 Studi Literatur

Pada tahap ini, peneliti melakukan pengumpulan data yang berhubungan dengan monitoring polusi udara, pembuatan aplikasi, *Database*, dan IoT. Data bisa diperoleh dari jurnal, artikel maupun buku. Tujuan dari studi literatur ini untuk memperkuat permasalahan yang diangkat untuk penelitian serta menjadi dasar untuk melakukan pengembangan berikutnya.

3.2.2 Observasi

Observasi adalah kegiatan yang dilakukan dengan mengamati secara langsung kondisi suatu tempat yang akan dijadikan target dalam penelitian. Observasi yang dilakukan peneliti awalnya berada di sekitar tempat tinggal peneliti yang berada di Kalimantan Selatan,

tetapi karena sekarang posisi peneliti sudah berada di Purwokerto maka target tempat yang digunakan sebagai lokasi penelitian akan di sesuaikan seperti di jalanan yang ada di pada kawasan di sekitar kawasan kos Samudra di jalan Sidodadi Purwokerto Kidul, Jawa Tengah.

3.2.3 Pengumpulan Kebutuhan

Dalam merancang sebuah sistem monitoring kadar polusi udara, peneliti menggunakan spesifikasi perangkat lunak dan perangkat keras sebagai berikut:

1. Spesifikasi perangkat lunak

Analisis perangkat lunak yang digunakan dalam penelitian ini terdiri dari perangkat lunak yang akan digunakan dalam membuat sebuah sistem dari alat monitoring polusi udara guna memberikan informasi akan keadaan dari kualitas udara kepada masyarakat sekitar.

Perangkat lunak yang digunakan untuk membangun sistem adalah sebagai berikut:

A. Arduino IDE

Arduino IDE, pada penelitian ini digunakan untuk membuat sebuah program guna memeberikan intruksi kepada perangkat mikrokontroler dan juga perangkat sensor dan modul yang digunakan sehingga bisa beroperasi sesuai kebutuhan.

B. Android Studio

Android Studio, pada penelitian ini digunakan untuk membuat sebuah aplikasi android guna menampilkan data yang di kirimkan oleh *Database hosting*, yang mana data tersebut berasal dari perangkat monitoring kadar polusi udara.

C. *phpMyAdmin*

phpMyAdmin, digunakan untuk membuat sebuah *Database* penyimpanan data yang menyimpan data yang dikirimkan oleh perangkat NodeMCU, kemudian data tersebut akan dikirim sebagai

informasi ke pada aplikasi yang berada di perangkat *smartphone* sebagai bentuk notifikasi.

2. Spesifikasi perangkat keras

Spesifikasi perangkat keras terdiri dari alat yang akan digunakan dalam membuat perangkat monitoring suhu dan kadar polusi udara. yang digunakan dalam membuat perangkat monitor adalah sebagai berikut:

A. Laptop

Laptop digunakan dalam perancangan dan pembuatan alat untuk membuat program menggunakan *software Android IDE* untuk NodeMCU, juga digunakan untuk membuat aplikasi android beserta *Databasenya* menggunakan aplikasi *Android Studio* dan *phpMyAdmin*. Dalam pembuatannya spesifikasi laptop yang digunakan yaitu prosesor Inter® Core™ i3-4010U, dengan RAM 4GB, VGA Nvidia Geforce 740M dan menggunakan *operation sistem windows 10 64bit*.

Tabel 3.1 Spesifikasi Laptop

Perangkat	Spesifikasi
Prosesor	Inter® Core™ i3-4010U
RAM	4GB
VGA	NVIDIA GEFORCE 740M
HDD	500GB
Display	14 inch

B. NodeMCU

NodeMCU, pada penelitian ini berfungsi sebagai penerima dan pengirim informasi data dari sensor ke aplikasi. Sebenarnya terdapat beberapa mikrokontroler yang dapat digunakan dalam pembuatan alat penelitian ini, seperti Arduino Uno dan Raspberry Pi, tapi apabila ingin menggunakan konsep yang mudah dan tidak memakan terlalu banyak biaya, NodeMCU adalah pilihan yang cocok. Pada

perangkat Arduino untuk terhubung ke Internet diperlukan perangkat tambahan modul wifi, sedangkan NodeMCU sudah memiliki modul WIFI ESP8266 yang tertanam langsung di perangkatnya. Raspberry Pi mempunyai spesifikasi yang lebih unggul dibandingkan dengan NodeMCU tetapi memiliki harga yang lebih mahal.

Tabel 3.2 Tabel Spesifikasi NodeMCU

Perangkat	Spesifikasi
Ukuran <i>board</i>	57mm x 30mm
Tegangan <i>Input</i>	3.3 - 5V
I/O	13 Pin
Kanal PWM	10 Kanal
10bit ADC Pin	1 Pin
<i>Flash memory</i>	4 MB
<i>Clock speed</i>	40/26/24 MHz
Wifi	IEEE 802.11 b/g/n
Frekuensi	2.4 GHZ – 22.5 GHZ
USB <i>Port</i>	Micro USB
USB <i>To Serial Converter</i>	CH340G

C. Sensor MQ-7

MQ-7 adalah sensor gas yang dipakai dalam peralatan untuk mendeteksi gas karbon monoksida (CO). Sensor MQ-7 mempunyai fitur sensitivitas yang tinggi terhadap karbon monoksida (CO). sensor ini menggunakan catu daya heater: 5V AC/DC dan menggunakan catu daya rangkaian: 5VDC, jarak pengukuran yang ampuh untuk alat ini adalah 20-2000ppm untuk mengukur gas karbon monoksida.

Tabel 3.3 Tabel Spesifikasi Sensor MQ-7

Perangkat	Spesifikasi
VC / (tegangan rangkaian)	$5V \pm 0.1$
VH (H)/ tegangan pemanas (high)	$5V \pm 0.1$
VH (L)/ tegangan pemanas (low)	1.4 ± 0.1
RL / relasi beban	Dapat disesuaikan
RH / resistensi pemanas	$33\Omega \pm 5\%$
TH (H)/waktu pemanasan (high)	60 ± 1 second
TH (L)/waktu pemanasan (low)	90 ± 1 second
PH konsumsi pemanasan	Sekitar 350mW

D. DHT11

DHT11 merupakan sensor digital yang dapat mengukur suhu dan kelembaban udara di sekitarnya. Sensor ini memiliki stabilitas yang baik serta fitur kalibrasi yang akurat. DHT11 ini termasuk sensor yang memiliki kualitas terbaik, dinilai dari respon, pembaca data yang cepat dan kemampuan anti-interference. Sensor ini cocok dipakai untuk aplikasi pengukuran suhu dan kelembaban.

Tabel 3.4 Tabel Spesifikasi DHT11

Perangkat	Spesifikasi
Tegangan <i>Input</i>	3,3 – 5V
<i>Range</i> Suhu	0 – 50°C
<i>Range</i> Kelembaban	20 – 90% RH
Akurasi	$2^{\circ}C \pm (\text{temperature}) \pm 5^{\circ}C$ RH (humidity)

E. LCD 16 X 2

LCD 16x2 (Liquid Crystal Display) digunakan sebagai media penyampaian informasi pada perangkat penelitian. LCD adalah sebuah modul penampilan data yang menggunakan kristal cair

sebagai bahan penampil data yang akan berupa tulisan atau pun gambar. LCD 16x2 juga memiliki beberapa fitur diantaranya terdiri dari 16 kolom dan 2 baris, dilengkapi dengan back light, memiliki 192 karakter tersimpan dan dapat dialamati dengan mode 4bit dan 8 bit.

Tabel 3.5 Tabel pin pada LCD 16x2

Pin	Keterangan
GND	Catu daya 0Vdc
VCC	Catu daya positif
Constrate	Untuk kontras tulisan pada LCD
RS / Register Select	High = berfungsi mengirim data Low = berfungsi mengirim intruksi
R/W (Read / Write)	High = berfungsi mengirim data Low = berfungsi mengirim intruksi Disambungkan ke LOW untuk mengirimkan data ke layer
E (ENABLE)	Berfungsi mengontrol ke LCD Ketika memiliki nilai LOW, LCD tidak dapat diakses
D0 – D7	Data bus 0-7
Backlight+	Disambung ke VCC untuk menyalakan lampu latar
Backlight-	Disambung ke VCC untuk menyalakan lampu latar

Tabel 3.6 Spesifikasi LCD

Perangkat	Spesifikasi
<i>Display format</i>	16 karakter x 4 lines
Tegangan suplai	5V
<i>Back lit</i>	Blue with white char color
<i>Pcb Size</i>	60mm99mm
<i>Contras Adjust</i>	<i>potentiometer</i>
<i>Backlight Adjust</i>	<i>jumper</i>

F. *Smartphone*

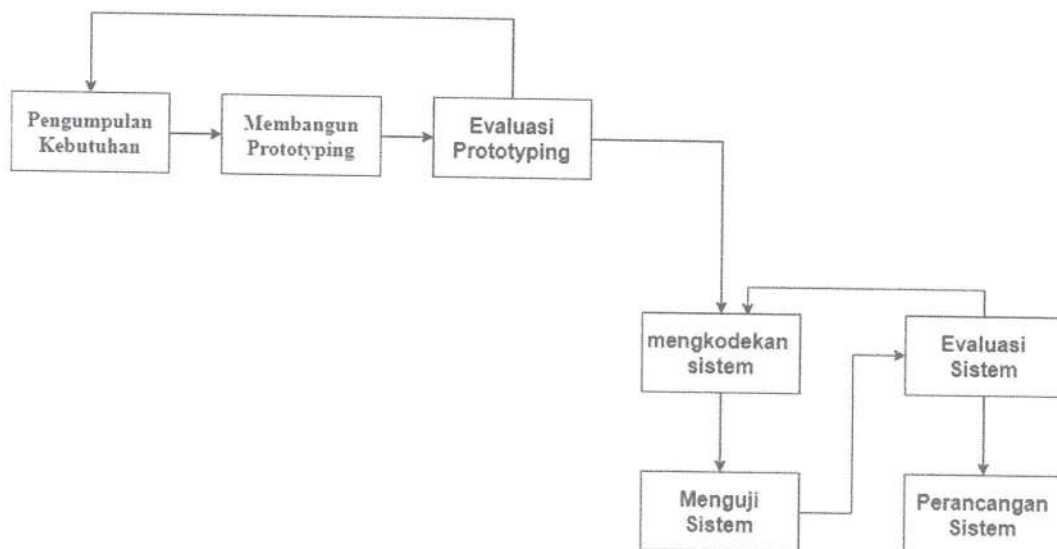
Smartphone dalam penelitian ini digunakan sebagai media monitoring kadar polusi udara melalui aplikasi yang dibuat dari Android Studio.

Tabel 3.7 Spesifikasi *Smartphone*

Perangkat	Spesifikasi
Nama Perangkat	Realme 3 Pro
Versi <i>Android</i>	10
RAM	4GB
Prosesor	Qualqom SDM710
<i>Storage</i>	64GB

3.2.4 Menentukan Metode Penelitian

Metode yang digunakan oleh peneliti dalam penelitian ini adalah metode *Prototyping* untuk membantu peneliti dalam mengerjakan penelitian ini. Adapun alur yang digunakan dalam metode ini untuk membantu penelitian ini adalah sebagai berikut.



Gambar 3.2 Alur Diagram Metode *Prototyping*

A. Pengumpulan Kebutuhan

Langkah pertama dalam metode *Prototyping* adalah mengidentifikasi seluruh perangkat dan permasalahan. Tahapan pengumpulan yang sangat penting adalah analisis dan mengidentifikasi kebutuhan secara garis besar dari sistem. Setelah itu akan diketahui langkah apa dan permasalahan yang akan dibuat dan dipecahkan.

B. Membangun Prototyping

Langkah selanjutnya adalah membangun *prototipe* yang berfokus pada penyajian kepada *user*. Misalnya seperti membuat *Input* dan *Output* dari hasil sistem.

C. Evaluasi Prototyping

Sebelum melanjutkan ke Langkah selanjutnya, Langkah ini bersifat wajib yaitu memeriksa Kembali langkah 1 dan 2 karena ini adalah penentu keberhasilan dan proses yang penting. Ketika Langkah ke 1 dan 2 ada yang kurang dan salah, maka kedepannya akan sulit untuk melanjutkan ke langkah selanjutnya.

D. Mengkodekan Sistem

Pada Langkah ini kita melakukan proses koding, mulai dari program untuk perangkat mikrokontroler maupun program untuk aplikasi pada android.

E. Menguji Sistem

Setelah pengkodean atau pengkodingan tentu akan dilakukan sebuah *testing*. Pengujian yang dilakukan pada metode ini adalah menggunakan *Black box* sebagai skenario penguji perangkat lunak untuk pengguna pada sistem monitoring kadar polusi udara.

F. Evaluasi Sistem

Mengevaluasi dari semua Langkah yang pernah dilakukan, apakah sudah sesuai dengan kebutuhan atau belum. Jika masih ada yang perlu di revisi maka dapat mengulangi ke tahap 1 dan 2.

G. Penggunaan Sistem

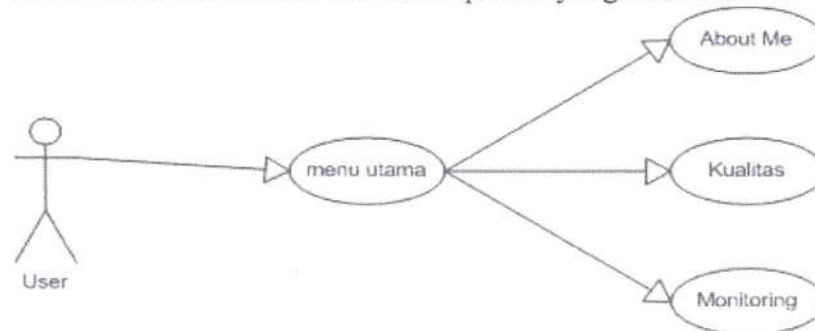
Apabila Perancangan alat dan sistem yang dibuat sudah selesai maka selanjutnya akan di serahkan kepada masyarakat.

3.2.5 Implementasi Metode Penelitian

Pada penelitian ini akan membuat sebuah alat yang digunakan untuk mendeteksi suhu dan gas karbon monoksida (CO) dengan menggunakan *Internet of Things*. Ada beberapa hal yang akan dibuat dalam penelitian ini yaitu membangun perancangan perangkat *hardware* yang berfungsi mengumpulkan data informasi monitoring, dan *software* yang fungsinya sebagai media penyimpanan data dan menampilkan informasi.

A. Use Case Diagram

Di bawah ini adalah *use case* dari aplikasi yang akan dibuat:



Gambar 3.3 Use case Program

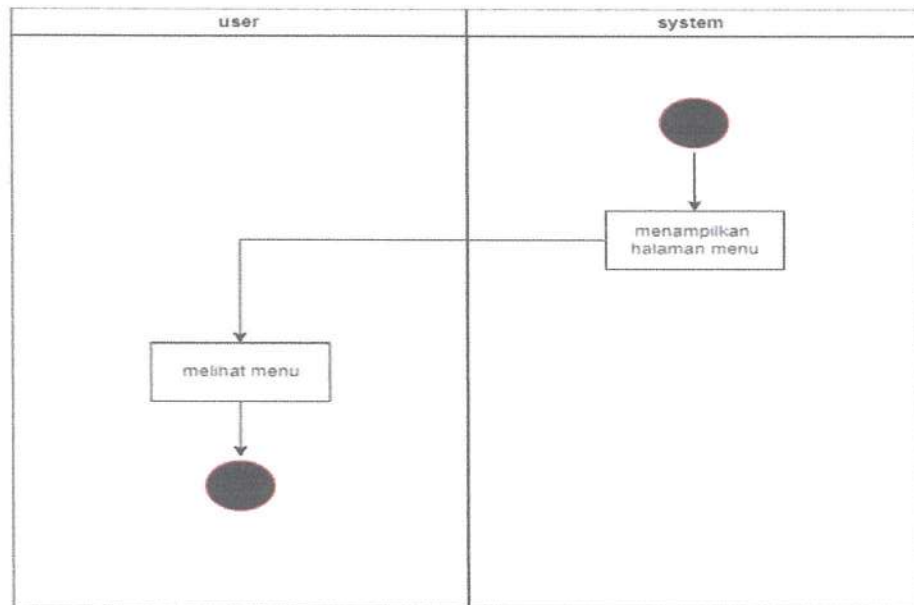
Alur dari *use case* di atas adalah sebagai berikut, *user* atau pengguna mengakses aplikasi dan membuka menu utama, didalam menu utama akan terdapat 3 menu yaitu:

- a. *About Me*, untuk menampilkan informasi tentang pengembangan aplikasi, dan tentang data diri peneliti.
- b. *Kualitas*, untuk menampilkan data kualitas udara yang menjadi patokan dari kadar polusi udara yang akan di monitor.
- c. *Monitoring*, adalah halaman yang menampilkan informasi tentang hasil *monitoring* yang didapat oleh perangkat berupa suhu, dan karbon monoksida (CO), dan juga menampilkan anjuran apa yang

harus dilakukan yang berhubungan tentang kondisi kualitas udara yang ditampilkan.

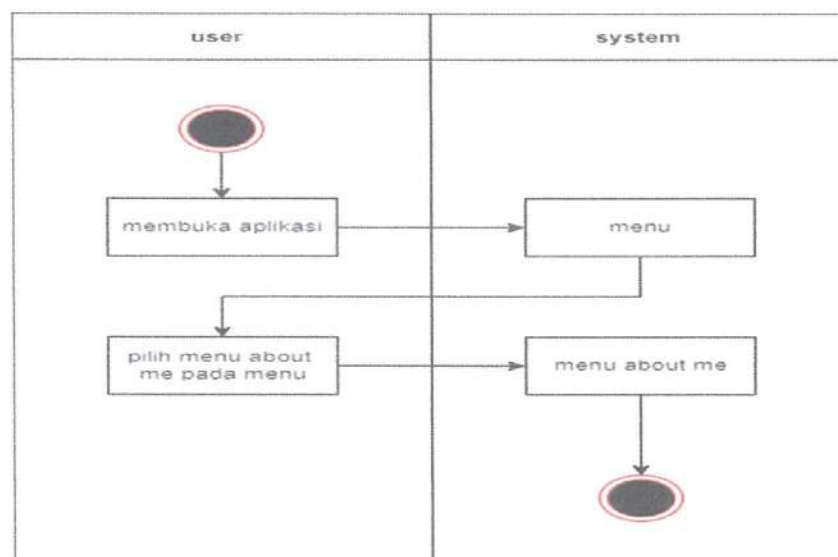
B. Activity Diagram

Berikut ini adalah alur *Activity* diagram dari aplikasi yang akan dibuat.



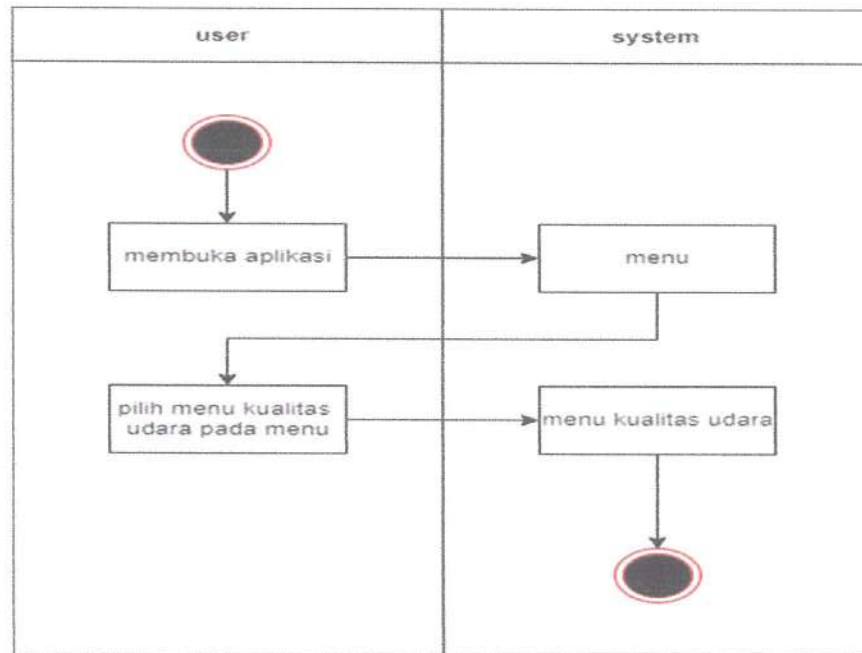
Gambar 3.4 *Activity* melihat menu

Pada gambar 3.4 menggambarkan aktivitas *user* Ketika ingin melihat tampilan menu.



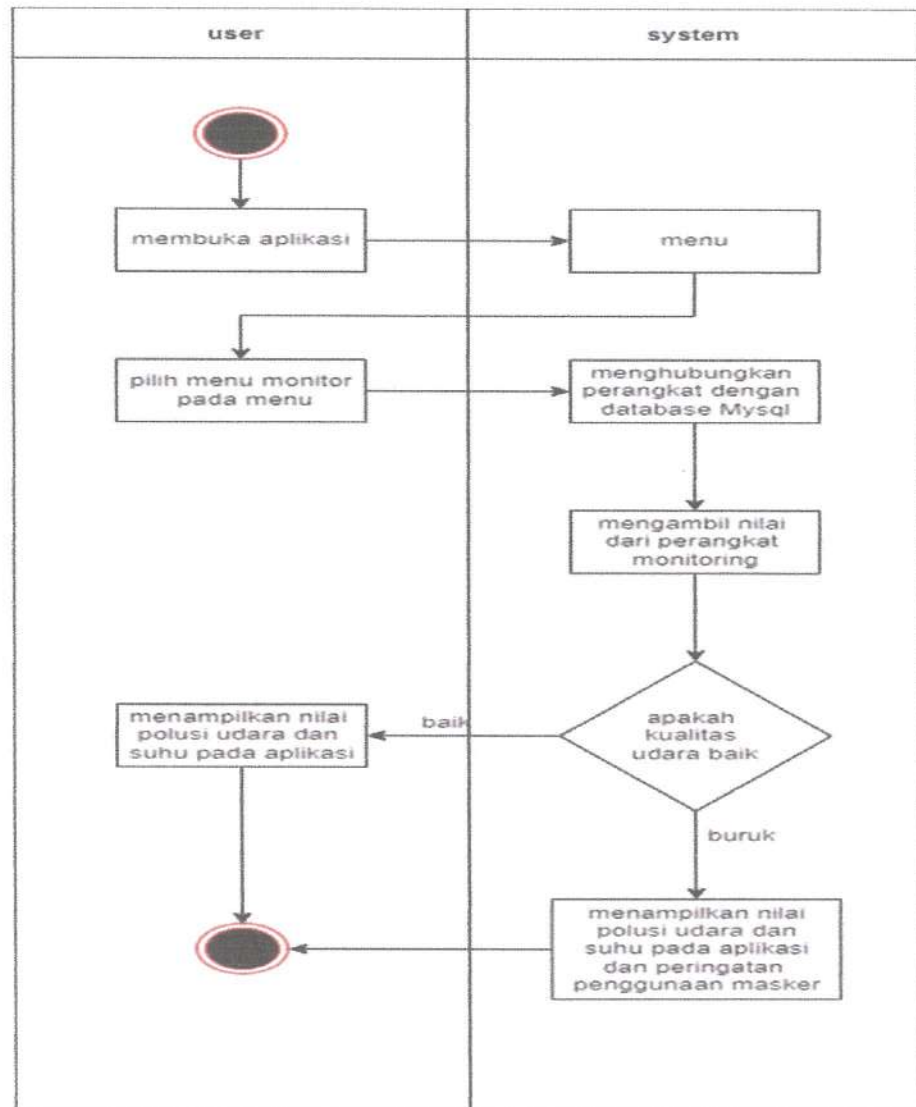
Gambar 3.5 Activity melihat halaman about me

Pada gambar 3.5 menggambarkan aktivitas *user* apabila ingin melihat halaman about me pada menu.



Gambar 3.6 Activity melihat halaman kualitas udara

Pada gambar 3.6 menggambarkan aktivitas *user* apabila ingin melihat halaman kualitas udara pada menu.

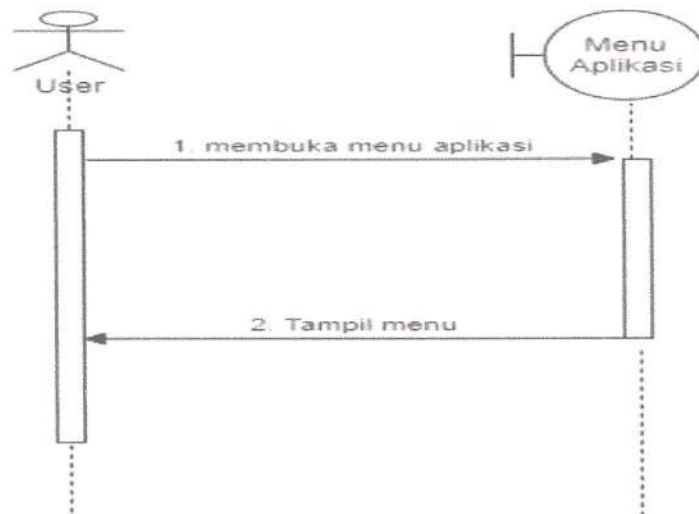


Gambar 3.7 Activity melihat halaman monitor

Pada gambar 3.7 menggambarkan aktivitas *user* apabila ingin melihat halaman kualitas udara pada menu.

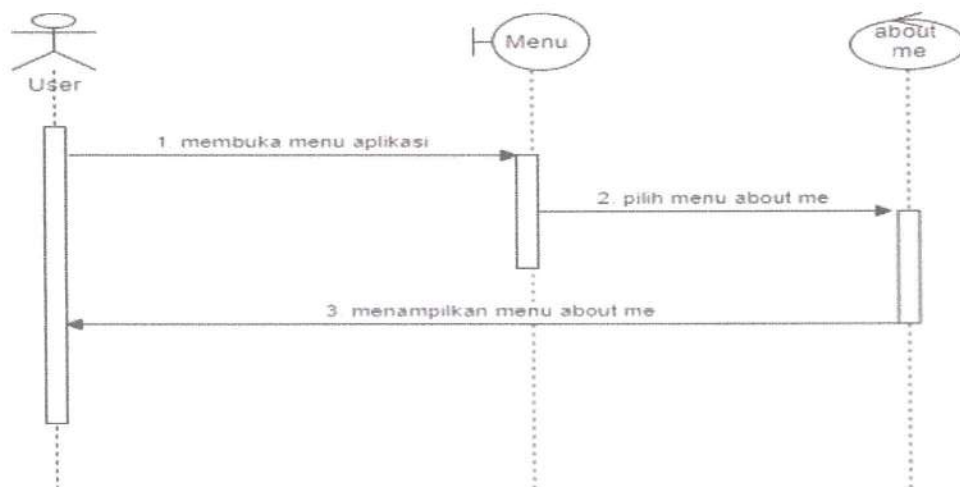
C. Sequence Diagram

Berikut ini adalah rancangan *sequence* diagram untuk aplikasi monitoring



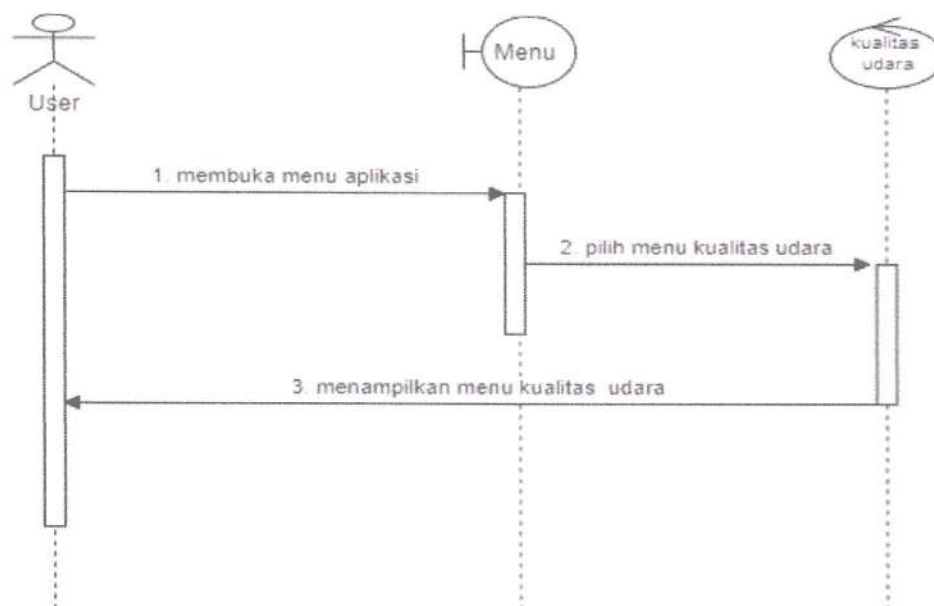
Gambar 3.8 *sequence* diagram menu

Pada gambar 3.8 *user* akan membuka aplikasi monitoring dan aplikasi akan langsung menampilkan halaman menu kepada *user*.



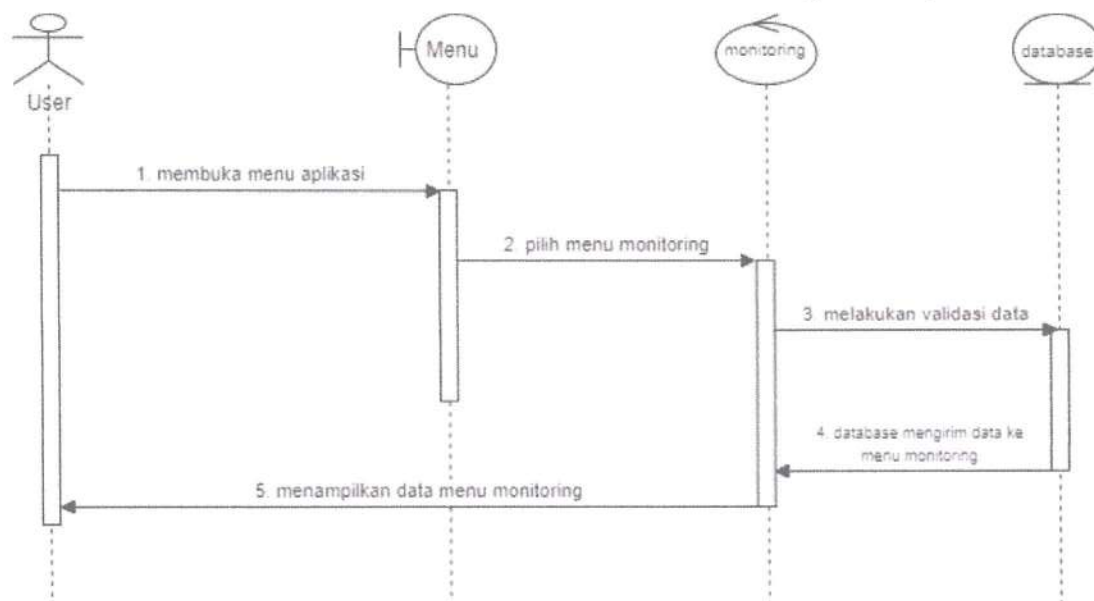
Gambar 3.9 *sequence* diagram *about me*

Pada gambar 3.9 *user* akan membuka aplikasi dan akan langsung menampilkan halaman menu kepada *user*, lalu *user* memilih menu *about me* dan menu *about me* akan di tampilkan kepada *user*.



Gambar 3.10 *sequence* diagram kualitas udara

Pada gambar 3.10 *user* akan membuka aplikasi dan akan langsung menampilkan halaman menu kepada *user*, lalu *user* memilih menu kualitas udara dan menu kualitas udara akan di tampilkan kepada *user*.



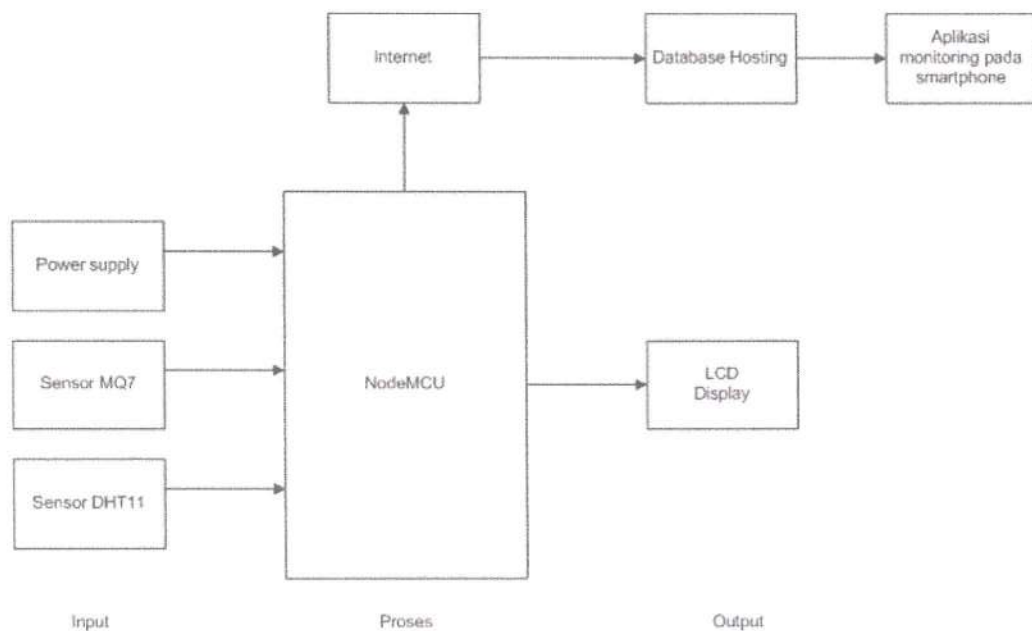
Gambar 3.11 *sequence* diagram monitor

Pada gambar 3.11 *user* akan membuka aplikasi dan akan langsung menampilkan halaman menu kepada *user*, lalu *user* memilih menu

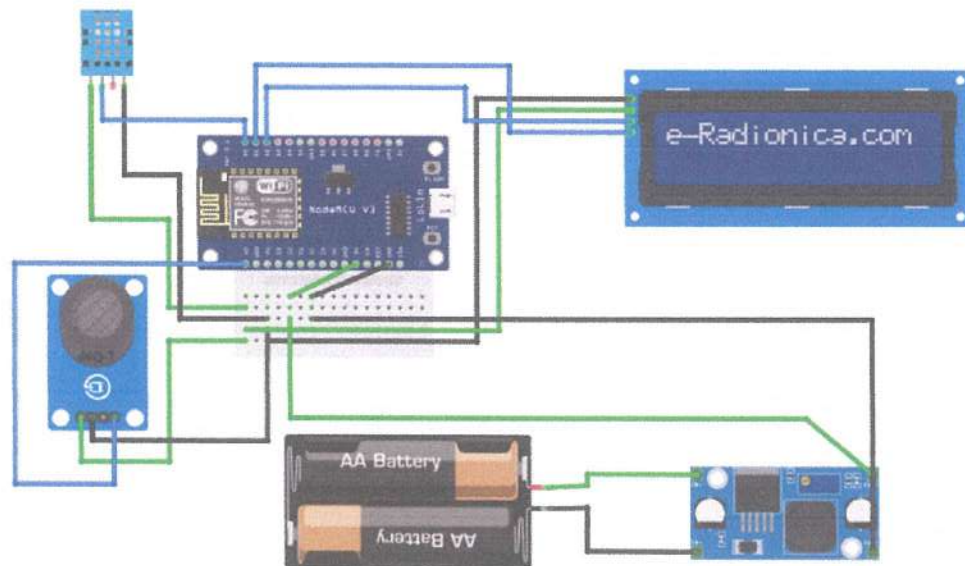
monitor dan menu monitor akan menampilkan data yang diambil dari *Database* dan di tampilkan kepada *user*.

D. Perancangan Hardware

Pada penelitian ini akan merancang sebuah alat yang dapat memonitor suhu dan gas karbon monoksida (CO) untuk mengukur kadar polusi udara dengan menggunakan *Internet of Things*. Pada pembuatan alat ini menggunakan perangkat mikrokontroler NodeMCU yang dirangkai dengan dua sensor yaitu MQ-7 yang berfungsi mendeteksi gas karbon monoksida (CO), dan DHT11 yang berfungsi mendeteksi suhu dan kelembaban udara. Alat ini nantinya akan mengirimkan informasi data kepada pengguna melalui aplikasi android. Di bawah ini adalah rancangan alat yang akan dibuat:



Gambar 3.12 Blok Diagram Rancangan Alat



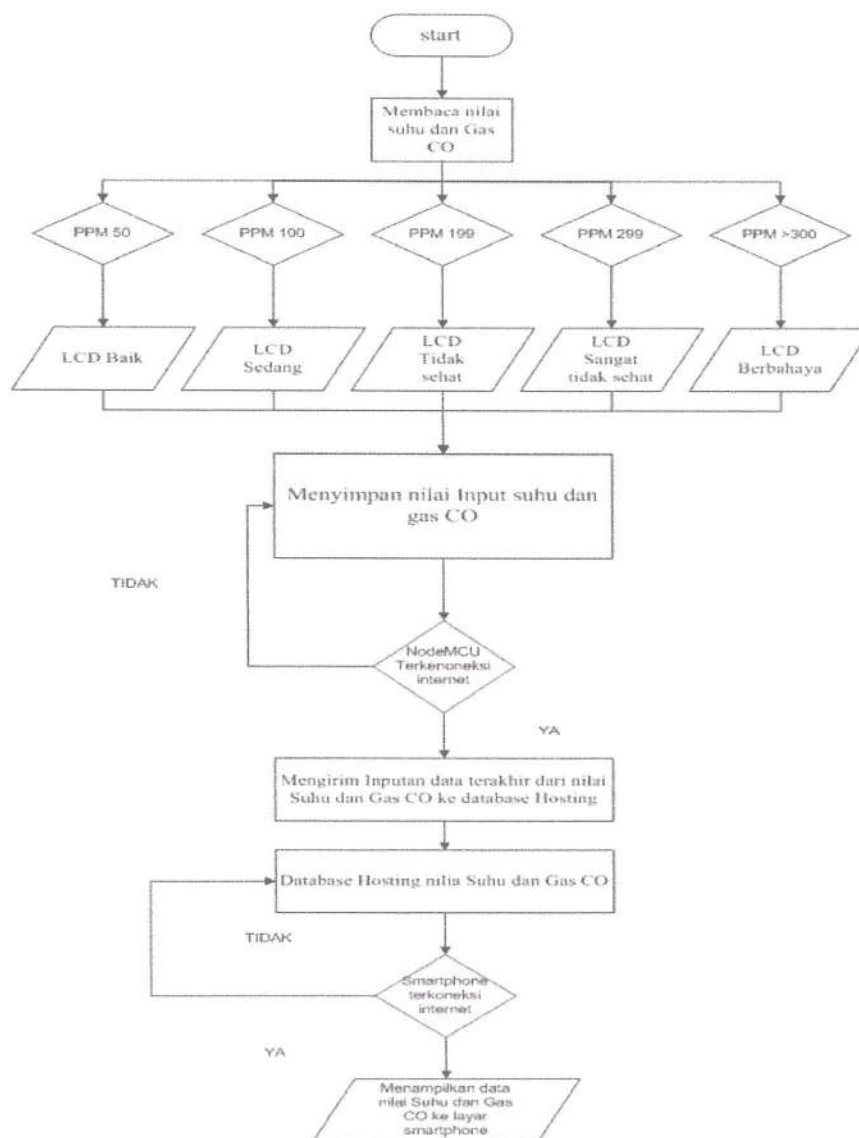
Gambar 3.13 Rancangan Alat

Pada **Gambar 3.13** merupakan perancangan alat dari sistem monitoring suhu dan karbon monoksida (CO) untuk mengukur kadar polusi berbasis *Internet of Things* yang akan dibuat. Pada rangkaian perangkat di atas terdiri dari mikrokontroler NodeMCU yang berada di tengah berfungsi sebagai penerima data yang didapat oleh sensor yang kemudian data tersebut akan dikirimkan ke layar LCD 16 X 2 dan *Database hosting* untuk menampilkan data informasi tentang keadaan kualitas udara. Lalu ada perangkat MQ-7 yang berada di bagian sisi kiri bawah yang berfungsi mendeteksi gas karbon monoksida (CO), dan kemudian data yang diperoleh akan dikirimkan ke perangkat NodeMCU. Lalu pada bagian sisi pojok kiri atas terdapat DHT11 yang berfungsi mendeteksi kelembaban udara dan kemudian data akan dikirim ke perangkat NodeMCU. Lalu pada sisi pada bagian kanan terdapat perangkat LCD 16x2 yang berfungsi untuk menampilkan langsung data informasi yang dikumpulkan oleh perangkat sensor yang disimpan lalu kemudian dikirim oleh NodeMCU.

Tabel 3.8 Penggunaan Pin Pada Rancangan Alat

Pin NodeMCU	Pin DHT11
Pin 3V3	Pin VCC
Pin GND	Pin GND
Pin D0	Pin Data
Pin NodeMCU	Pin MQ-7
Pin 3V3	Pin VCC
Pin GND	Pin GND
Pin A0	Pin A0
Pin NodeMCU	Pin LCD 16x2
Pin 3V3	Pin VCC
Pin GND	Pin GND
Pin D2	Pin SDA
Pin D1	Pin SCL

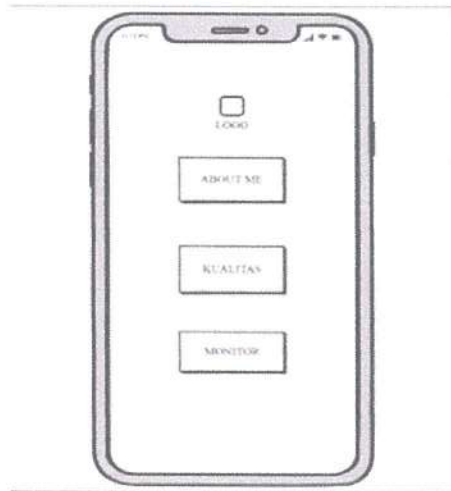
Langkah-langkah dari cara kerja sistem monitoring suhu dan karbon monoksida (CO) dapat dilihat sebagai berikut:



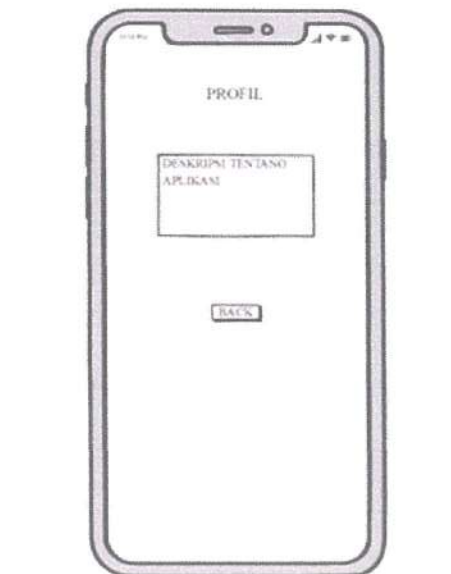
Gambar 3.14 Flowchart Alur Kerja Sistem

E. Perancangan *User Interface*

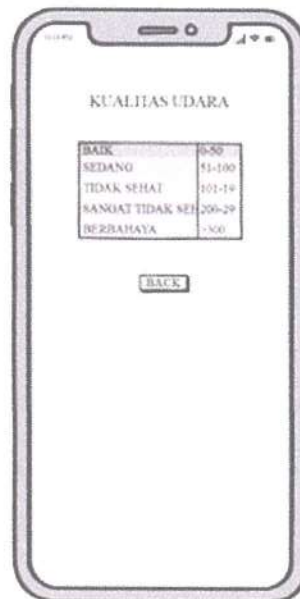
Perangkat monitoring kadar polusi udara nantinya akan mengirimkan data ke *Database* hosting, lalu selanjutnya data itu kemudian dari *Database* tersebut akan dikirimkan ke perangkat *Android* sebagai media monitoring. Pembuatan aplikasi monitoring ini menggunakan aplikasi *Android Studio*. Di bawah ini adalah desain dari tampilan halaman *Android* yang akan dibuat:



Gambar 3.15 Tampilan Utama



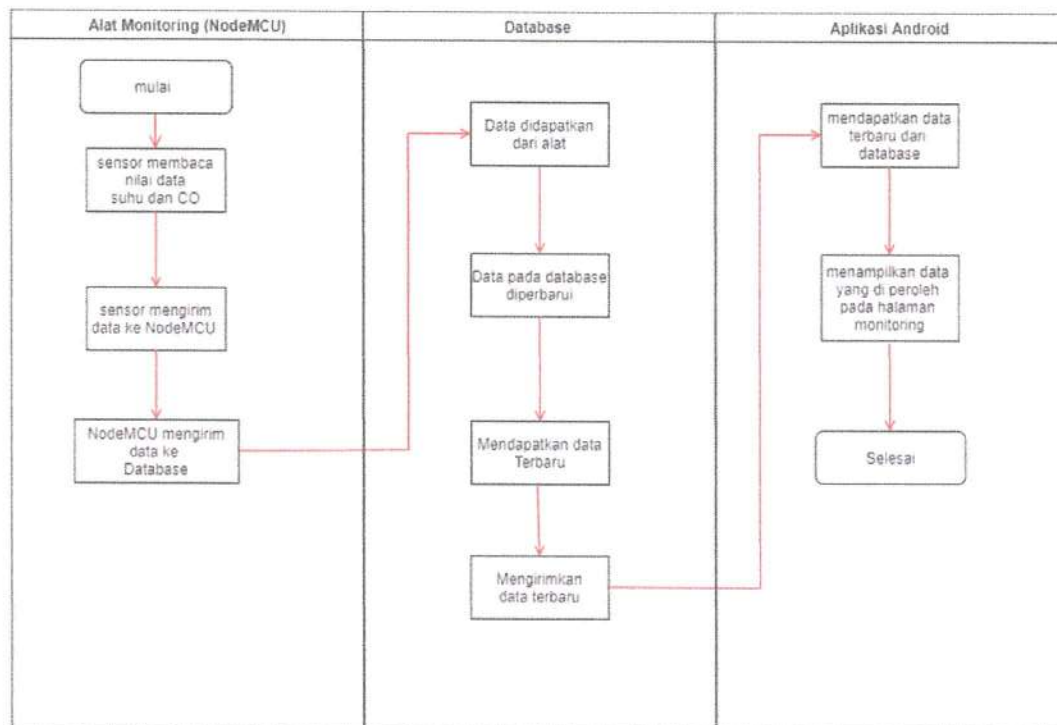
Gambar 3.16 Halaman About Me



Gambar 3.17 Halaman Kualitas udara



Gambar 3.18 Halaman Monitor



Gambar 3.19 Alur Data Dari *Hardware* ke *Mobile App*

Pada gambar 3.19 merupakan proses untuk menampilkan data yang diperoleh dari alat monitoring ke aplikasi android. Sensor MQ-7 dan DHT11 akan mengirimkan data ke NodeMCU. Data ini dikirimkan dari NodeMCU ke *Database* melalui jaringan internet, dan aplikasi android akan menampilkan data yang diperoleh dari *Database* ke halaman monitoring.

F. Pengkodean Sistem

Pada tahap ini rancangan yang sudah dibuat akan diubah kedalam bahasa pemrograman, yaitu pada pembuatan aplikasi *android* dan juga pemrograman pada mikrokontroler. Pengkodean aplikasi *android* dilakukan dengan menggunakan aplikasi Android Studio. Berikut ini adalah cuplikan dari proses pengkodean agar aplikasi android dapat menampilkan data dari *Database*.

```
private static final String TAG = Hasil.class.getSimpleName();
private String url_select = "https://apeda.renoagilsaputra.my.id/select.php";
public static final String TAG_ID = "id_hasil";
```

```

public static final String TAG_ppm = "ppm";
public static final String TAG_co = "co";
public static final String TAG_status = "status";

private static final String TAG_SUCCESS = "success";
private static final String TAG_MESSAGE = "message";

String tag_json_obj = "json_obj_req";

```

Potongan kode di atas adalah untuk mendefinisikan url *Database* dan mendefinisikan data.

```

// membuat request JSON
JSONArrayRequest jArr = new JSONArrayRequest(url_select, new
Response.Listener<JSONArray>() {
    @Override
    public void onResponse(JSONArray response) {
        Log.d(TAG, response.toString());

        // Parsing json
        for (int i = 0; i < response.length(); i++) {
            try {
                JSONObject obj = response.getJSONObject(i);
                Data item = new Data();
                item.setId(obj.getString(TAG_ID));
                item.setppm(obj.getString(TAG_ppm));
                item.setco(obj.getString(TAG_co));
                item.setStatus(obj.getString(TAG_status));

                // menambah item ke array
                itemList.add(item);
            } catch (JSONException e) {
                e.printStackTrace();
            }
        }

        // notifikasi adanya perubahan data pada adapter
        adapter.notifyDataSetChanged();
        swipe.setRefreshing(false);
    }
}, new Response.ErrorListener() {

```

```

@Override
public void onErrorResponse(VolleyError error) {
    VolleyLog.d(TAG, "Error: " + error.getMessage());
    swipe.setRefreshing(false);
}
});
// menambah request ke request queue
AppController.getInstance().addToRequestQueue(jArr);
}

```

Potongan kode di atas adalah berfungsi untuk menampilkan listview pada *Database*.

Berikutnya adalah kodingan yang terdapat pada NodeMCU ESP8266, dan proses pemrograman menggunakan Arduino IDE.

```

void loop() {
    suhu = dht.readTemperature();
    sensorvalue = analogRead(pin_sensor_MQ);
    vr1 = sensorvalue*5.00/1024;
    rs = ( 5.00 * rl / vr1 ) - rl;
    ppm = 100 * pow(rs / ro,-1.53); // ppm = 100 * ((rs/ro)^-1.53);
    delay(1000);
}

```

Potongan kode di atas berfungsi untuk mencari nilai data yang akan dibaca oleh sensor MQ-7 dan DHT11 dan kemudian di proses oleh NodeMCU, kemudian data yang didapat oleh sensor akan ditampilkan pada LCD kode di bawah ini.

```

void tampilan() {
    lcd.setCursor(0, 0);
    lcd.print(String(suhu) + "C");
    lcd.setCursor(8, 0);
    lcd.print("CO:");
    lcd.setCursor(12, 0);
}

```



```
lcd.print(ppm);
if (tampil == 1) {
  lcd.setCursor(0, 1);
  lcd.print("Baik      ");
}
else if (tampil == 2) {
  lcd.setCursor(0, 1);
  lcd.print("sedang     ");
}
else if (tampil == 3) {
  lcd.setCursor(0, 1);
  lcd.print("Tidak sehat! ");
}
else if (tampil == 4) {
  lcd.setCursor(0, 1);
  lcd.print("Sangat tdk sehat ");
}
else if (tampil == 5) {
  lcd.setCursor(0, 1);
  lcd.print("Berbahaya!   ");
}
}

void tugas(){
  if(ppm >= 299){
    tampil=5;
  }
  else if(ppm > 199){
    tampil=4;
  }
  else if(ppm > 99){
    tampil=3;
```

```

}
else if(ppm >50){
  tampil=2;
}
else {
  tampil=1;
}
}
}

```

G. Pengujian Sistem

Pada tahap ini dilakukan pengujian sistem aplikasi *monitoring*. Kegiatan ini bertujuan untuk mengetahui kekurangan dari sistem aplikasi, pengujian yang dilakukan pada aplikasi sistem monitoring kadar polusi udara ini menggunakan *Black box testing*.

Pengujian yang dilakukan pada aplikasi adalah pengujian fungsional, dan apa yang akan diuji adalah sebagai berikut:

Tabel 3.9 Pengujian Fungsional Sistem

Kasus Uji	Detail Pengujian	Hasil Pengujian
<i>Front end</i> alat monitoring polusi udara	Menampilkan tombol monitor	
	Menampilkan tombol kriteria udara	
	Menampilkan tombol <i>about me</i>	
	Menampilkan logo sistem	
<i>Back end</i> alat monitoring polusi udara	Masuk ke halaman monitor	
	Mengambil data pada <i>Database</i>	
	Menampilkan data pada halaman monitor	
	Masuk ke halaman kriteria udara	
	Masuk ke halaman <i>about me</i>	

H. Pengujian Perangkat Monitoring

Pengujian ini dilakukan untuk mencoba apakah semua komponen dapat berjalan dengan baik dan lancar. Adapun pengujian yang dilakukan meliputi:

a. Pengujian NodeMCU dan Sensor

Pengujian ini dilakukan untuk menguji apakah NodeMCU dan sensor dapat membaca nilai data kualitas udara dan mengirimkan data tersebut ke *Database* yang telah dibuat.

Tabel 3.10 Pengujian Fungsi Alat

Kelas uji	Bagian Uji	Keterangan
NodeMCU dan sensor	Membaca nilai data dari kualitas udara	
	Mengirim data kedalam <i>Database</i>	

b. Pengujian sensor gas MQ-7

Pengujian sensor MQ-7 akan menggunakan pengujian kalibrasi, yaitu dengan membandingkan hasil ukur dari alat sensor MQ-7 dengan CO meter. Hal ini dilakukan dengan cara memaparkan sensor langsung ke objek yang dapat menghasilkan gas polusi udara karbon monoksida seperti asap pembakaran obat nyamuk, asap rokok, dan juga asap knalpot kendaraan bermotor. Pengujian dilakukan sebanyak 30 kali dengan waktu interval adalah selama 5 menit disetiap pengujian. Selain itu, tingkat keberhasilan dilakukan sesuai dengan rumus [8]:

$$Error (\%) = \left(\frac{(\text{sensor MQ7} - \text{CO Meter})}{\text{sensor MQ7}} \right) \times 100\% \quad (3.1)$$

$$\text{Rata-rata Error} = \frac{\text{jumlah Error}}{\text{banyaknya data}} \quad (3.2)$$

Tabel 3.11 Pengujian Sensor MQ7

No	Kondisi	Pengujian	Sensor MQ-7 (PPM)	CO Meter (PPM)	Error
1	Tanpa pengasapan	1			
		2			
		3			
		4			
		5			
		6			
		7			
		8			
		9			
		10			
2	Asap obat nyamuk	1			
		2			
		3			
		4			
		5			
		6			
		7			
		8			
		9			
		10			
3	Asap rokok	1			
		2			
		3			
		4			
		5			
		6			
		7			
		8			
		9			
		10			
4	Asap knalpot	1			
		2			
		3			
		4			
		5			
		6			
		7			
		8			
		9			
		10			

c. Pengujian DHT11

Pengujian sensor suhu DHT11 akan menggunakan pengujian kalibrasi, yaitu dengan membandingkan hasil ukur dari alat sensor DHT11 dengan *hygrometer*. Pengukuran dan pengujian nilai higrometer dilakukan sebanyak 10 kali. Selain itu, tingkat keberhasilan dilakukan sesuai dengan rumus [22]:

$$\%Berhasil = \frac{\bar{s}-x}{\bar{s}} \times 100\% \quad (3.3)$$

$$\%Gagal = \frac{x}{n} \times 100\% \quad (3.4)$$

Keterangan:

\bar{s} = nilai rata – rata sensor

x = Nominal beban yang diukur.

Tabel 3.12 Pengujian DHT11

Pembacaan Hygrometer	Pembacaan sensor DHT11			Rata-rata pembacaan sensor DHT11	Persen keberhasilan (%)	Persen kesalahan (%)
	Pengujian 1	Pengujian 2	Pengujian 3			
Rata-rata						

3.2.6 Evaluasi Keseluruhan Sistem

Pada tahap ini akan dilakukan proses pengumpulan data yang akan dianalisis. Data yang diperoleh berasal dari hasil uji seluruh perangkat dan sistem yang sudah dibuat sebelumnya. Apabila hasil yang diperoleh dari pengujian perancangan alat moitoring suhu dan karbon monoksida (CO) untuk mengukur kadar polusi udara berbasis *Internet of Things* (IoT) berjalan dengan baik maka penelitian ini bisa dianggap berhasil, sedangkan apabila masih terdapat kekurangan atau *error* di salah satu rancangan penelitian maka, harus dilakukan evaluasi ulang terhadap kekurangan tersebut untuk dapat diperbaiki.

3.2.7 Penulisan Laporan

Penulisan laporan dilakukan pada saat awal akan melakukan penelitian hingga penelitian telah sukses dilakukan. Laporan penelitian ini berisi tentang hasil dari penelitian yang sudah dilakukan. Tujuan yang ingin dituju peneliti dari penulisan laporan ini adalah sebagai media informasi seputar penelitian yang sudah dibuat oleh peneliti. Informasi yang disampaikan dalam laporan ini diharapkan dapat berguna bagi masyarakat.