

DAFTAR ISI

HALAMAN Sampul	i
HALAMAN Judul.....	ii
HALAMAN Persetujuan	iii
HALAMAN Pengesahan	vi
HALAMAN Pernyataan Keaslian Penelitian	v
HALAMAN Persembahan	vi
HALAMAN Motto	vii
Kata Pengantar	viii
DAFTAR ISI	i
DAFTAR Tabel	iii
DAFTAR Gambar	iv
DAFTAR Lampiran.....	xiii
INTISARI.....	vix
<i>ABSTRACT</i>	xv
BAB I PENDAHULUAN	
A. Latar Belakang Masalah	1
B. Rumusan Masalah	5
C. Batasan Masalah	5
D. Tujuan Penelitian	6
E. Manfaat Penelitian	6
BAB II TINJAUAN PUSTAKA	
A. Landasan Teori.....	7
B. Penelitian Sebelumnya	28
BAB III METODE PENELITIAN	
A. Tempat dan Waktu Penelitian	31
B. Metode Pengumpulan Data.....	32
C. Konsep Penelitian.....	32

BAB IV HASIL DAN PEMBAHASAN

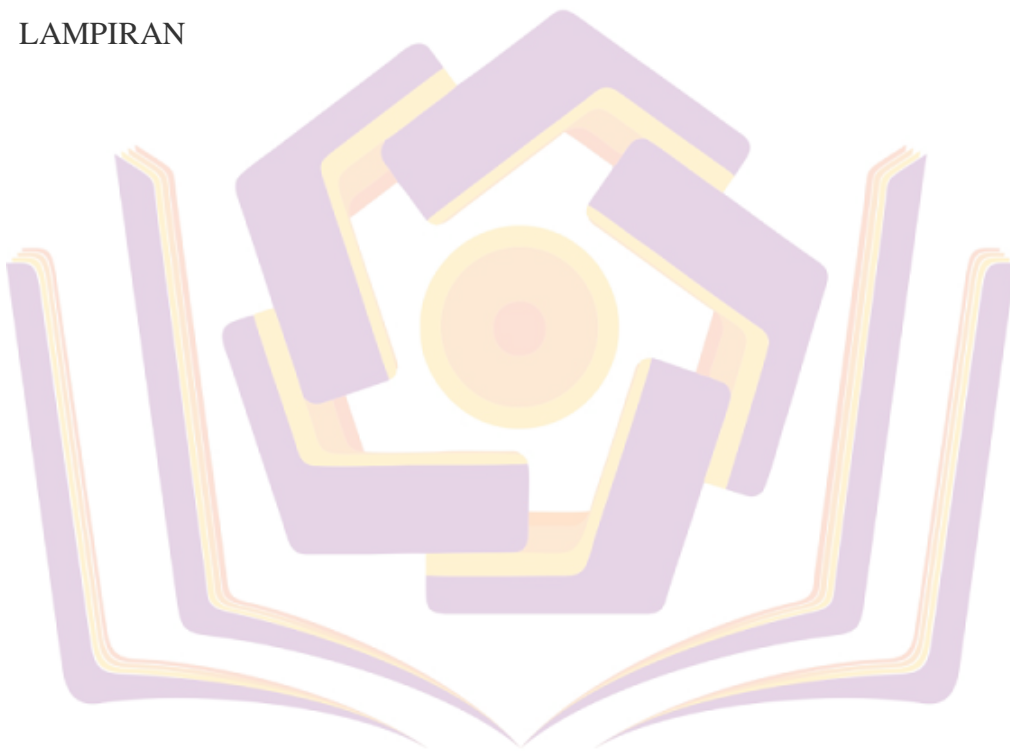
A. Gambar Objek Penelitian..... 37
B. Analisis Hasil 38

BAB V PENUTUP 69

A. Kesimpulan 69
B. Saran 70

DAFTAR PUSTAKA

LAMPIRAN



DAFTAR TABEL

Tabel 2.1. Spesifikasi Board Arduino	20
Tabel 2.2. Spesifikasi <i>Dust Sensor</i>	21
Tabel 2.3. Spesifikasi <i>SD Card</i>	23
Tabel 2.4. Spesifikasi <i>Real Time Clock</i>	25
Tabel 2.5. Spesifikasi <i>SD Card Module</i>	28



DAFTAR GAMBAR

Gambar 2.1. <i>Arduino Uno R3</i>	20
Gambar 2.2. <i>Dust Sensor</i>	23
Gambar 2.3. <i>Sd Card</i>	24
Gambar 2.4. <i>Real Time Clock</i>	25
Gambar 2.5. <i>Kabel Jumper</i>	26
Gambar 2.6. <i>Adaptor</i>	27
Gambar 2.7. <i>SD Card Module</i>	28
Gambar 4.1. Kerangka Berfikir	32
Gambar 4.2. Lokasi Puskesmas 1 Tambak pada Maps.....	38
Gambar 4.3. <i>Flowchart</i> Program.....	42
Gambar 2.2. Perancangan Sistem <i>Monitoring</i>	43
Gambar 2.2. Sketsa Prototipe alat.....	47
Gambar 2.2. Alat Tampak dari Belakang.....	48
Gambar 2.2. Alat Tampak dari Atas	49
Gambar 2.2. Alat Tampak dari Kiri.....	49
Gambar 2.2. Alat Tampak dari Kanan	50
Gambar 2.2. <i>Interface</i> pada Android sebelum revisi	50
Gambar 2.2. <i>Interface</i> pada Android Sebelum menghitung nilai	51
Gambar 2.2. <i>Interface</i> pada Android Setelah mulai menghitung	52
Gambar 2.2. <i>Interface</i> pada Android pengujian debu tanah	63
Gambar 2.2. <i>Interface</i> pada Android pengujian gas buang kendaraan	64

DAFTAR LAMPIRAN

Lampiran 1. Hasil wawancara

Lampiran 2. Wawancara dengan bapak dr. Harry Widiyatmoko

Lampiran 3. Hasil Kerjasama dengan pihak Puskesmas 1 Tambak

Lampiran 4. Kartu Bimbingan Skripsi





BAB I

PENDAHULUAN

A. Latar Belakang Masalah

Pada era *modern* dan globalisasi dengan kemajuan teknologi transportasi, pertambangan, pabrik yang memproduksi berbagai kebutuhan bagi individu dan organisasi di dunia seakan terasa lebih cepat dan singkat. Disamping kemajuan teknologi transportasi, pertambangan, dan pabrik penyumbang terbesar polusi udara adalah dari bidang transportasi. Penggunaan bidang transportasi yang dapat mempermudah masyarakat karena mempersingkat waktu tempuh dari tempat satu ke tempat yang lain, dapat mengubah arah tujuan, murah dan menjangkau berbagai tempat, dan bisa langsung ke tempat tujuan. Namun dampak lain yang ditimbulkan dari penggunaan Transportasi ini berbahaya bagi kesehatan pernapasan dan sangat berbahaya jika dihirup secara langsung. Partikel yang terdapat pada debu tanah yang kering karena terkena angin kendaraan yang lewat sangat mengganggu terhadap perumahan di sekitar dan pejalan kaki terutama bagi bidang kesehatan seperti klinik dan puskesmas yang seharusnya jauh dari debu dan asap yang berterbangan. Salah satu penyebab pencemaran udara adalah meningkatnya jumlah kendaraan di Indonesia. Jumlah kendaraan di Indonesia tahun 2016 mencapai 124.215 juta unit, naik 10-15 %. *World Health Organization* (WHO) menyatakan bahwa pencemaran udara merupakan faktor risiko gangguan kesehatan terbesar di dunia,

diperkirakan data tahun 2016 sekitar 6,5 juta orang meninggal tiap tahun akibat paparan polusi udara. Faktor yang berhubungan dengan konsentrasi pencemar udara adalah jumlah kendaraan, suhu udara, kelembaban udara, kecepatan angin, hujan, dan topografi. Berdasarkan regulasi Keputusan Menteri Negara Lingkungan Hidup Nomor 45 Tahun 1997, Standar Pencemar Udara di Indonesia (ISPU), dihitung salah satunya dari konsentrasi PM₁₀. PM₁₀ adalah debu partikulat yang mengapung di udara yang memiliki ukuran <10 mikrogram, sekira 1/10 ukuran helai rambut. Sejauh ini MenLHK dan Pemda DKI, yang juga didukung BMKG, telah mengukur kualitas udara di Jakarta dengan memasang alat pengukur konsentrasi PM₁₀ dan PM_{2.5} (Thirafi, 2019). Akibat dampak yang ditimbulkan dari penggunaan transportasi yang berlebihan berbahaya bagi kesehatan seperti penyakit *Particular Matter* (PM), *Ozone* (O₃), Karbon monoksida (CO). Dampak yang ditimbulkan dari *Particular matter* atau PM adalah kumpulan partikel padat atau cair yang ditemukan di udara. Komponen utama dari PM adalah sulfat, nitrat, amonia, natrium klorida, karbon hitam, mineral debu, dan air. Dampak yang ditimbulkan dari *Ozone* yang berlebihan di udara dapat melemahkan fungsi paru-paru. Akibatnya akan menyebabkan masalah pernapasan, memicu gejala asma kambuh, dan juga menyebabkan penyakit paru-paru. Karbon monoksida adalah salah satu gas beracun penyebab pencemaran udara. Dampak yang ditimbulkan dari Gas Karbon Monoksida ini memang tidak berwarna, tidak berbau, bahkan tidak mengiritasi kulit dan mata. Namun, menghirup karbon monoksida

dalam jumlah banyak sangatlah berbahaya sehingga berisiko buruk bagi kesehatan tubuh.

Beberapa penelitian sebelumnya untuk melakukan proses *monitoring* kualitas udara ini telah dilakukan oleh beberapa perguruan Tinggi maupun swasta. Referensi penulisan ini, terdapat 2 penelitian terkait dengan *monitoring* kualitas udara. Penelitian pertama dilakukan oleh Agus Sulistiyo dan Suryono yang telah menghasilkan penelitian dengan judul. Dalam penelitian tersebut bertujuan untuk *memonitoring* kualitas udara dengan menggunakan model penerimaan data konsentrasi debu jarak jauh secara *realtime* dengan menghasilkan *output* PWM (*Pulse Width Modulation*). Sensor ini terhubung dengan sistem minimum mikrokontroler. Mikrokontroler yang digunakan yaitu ATmega8535 dengan fitur USART (*Universal Synchronous and Asynchronous Serial Receiver and Transmitter*) sebagai mode komunikasi serial. Protokol RS-232 digunakan untuk menghubungkan data komunikasi serial ke komputer *Remote Terminal Unit* (RTU) yang terletak dilokasi pengambilan data. Data yang diperoleh dikirim ke *personal computer* (PC) *Control Terminal Unit* (CTU) di stasiun pemantauan dengan sistem telemetri menggunakan jaringan WiFi dan disimpan ke dalam basis data (Sulistiyo, 2016). Penelitian yang kedua dilakukan oleh Irma Dita Kurniawati, Ulfa Nurullita, dan Mufbakhuddin yang telah menghasilkan penelitian berjudul. Dalam penelitian tersebut bertujuan untuk mengetahui hubungan jumlah kendaraan dan kondisi iklim dengan konsentrasi CO, mengukur SO₂ dan NO₂ di terminal Mangkang dan

Terminal Penggaron Semarang. Dan pada penelitian tersebut menggunakan metode penelitian analitik observasional dengan pendekatan *cross sectional*. Objek dan sample penelitian yaitu udara di lokasi penelitian diukur dalam kurun waktu 1 jam. Variabel bebas dalam penelitian ini adalah jumlah kendaraan, suhu udara, kelembapan udara, dan kecepatan angin, sedangkan variabel terkait adalah konsentrasi karbon monoksida. Analisis data menggunakan *korelasi person* dan *rank spearman* (Kurniawati, Nurullita, 2017).

Berdasarkan latar belakang diatas maka penulis tertarik mengangkat skripsi dengan judul. Penelitian ini membuat sebuah alat *monitoring* kualitas udara menggunakan mikrokontroler *Arduino* sebagai pengendali *sensor* debu atau *dust sensor*. Dengan adanya alat pemantau kualitas udara diharapkan akan bisa memantau kualitas udara khususnya Jalan Raya Tambak.

A. Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan diatas, maka dapat dirumuskan suatu permasalahan untuk melakukan monitoring kualitas udara, pemantauan jarak jauh yang dapat mengetahui tingkat kepekatan karbon monoksida dan debu yang dapat memicu penyakit ISPA. Dengan adanya alat berbasis mikrokontroler yang dapat digunakan untuk memantau kapanpun dan dimanapun untuk mengidentifikasi kepekatan debu dan

karbon monoksida pada tiap waktu yang selanjutnya dapat disimpulkan untuk puncak dan rata-rata kepekatanya.

B. Batasan Penelitian

Agar dalam pembuatan alat ini dapat mencapai sasaran dan tujuan yang diharapkan, maka penelitian membahas permasalahan polusi yang ada di masyarakat sebagai berikut:

1. Alat ini masih Prototipe untuk *memonitoring* kualitas udara
2. Alat ini menggunakan *Adaptor* untuk terhubung dengan arus listrik
3. Pengeditan dan pembuatan menggunakan Arduino IDE

C. Tujuan Penelitian

Tujuan dari penelitian adalah membuat alat *monitoring* kualitas udara berbasis mikrokontroler yang dapat mempermudah untuk mengetahui kepekatan debu dan karbon monoksida terutama di daerah Tambak, Banyumas. Sehingga alat ini mampu dijadikan alternatif dalam membantu permasalahan polusi udara yang ada di daerah Tambak.

D. Manfaat Penelitian

Adapun manfaat penelitian yang diharapkan oleh penulis adalah sebagai berikut:

1. Manfaat Teoritik

a. Alat *monitoring* udara berbasis mikrokontroler Arduino dapat digunakan sebagai salah satu alat yang dapat membantu permasalahan memantau kualitas udara dalam masyarakat khususnya daerah Tambak.

b. Dapat dijadikan sebagai referensi karya tulis yang dapat dikembangkan pada penelitian selanjutnya.

2. Manfaat Aplikatif

Manfaat bagi para masyarakat khususnya daerah Tambak dengan adanya alat *monitoring* kualitas udara dapat dijadikan sebagai alternatif dalam bidang pencemaran udara pada lingkungan jalan raya Tambak.

BAB II

TINJAUAN PUSTAKA

A. Landasan Teori

1. Kota Tambak

Kota Tambak merupakan wilayah Kecamatan Banyumas yang paling timur dan berbatasan langsung dengan Kabupaten Kebumen. Mayoritas pekerjaan adalah wirausahawan dan PNS, Namun sebagian mudanya pergi merantau ke kota-kota seperti Jakarta, Bandung, dan kota besar lainnya. Pada pertaniannya Cukup berpotensi karena rata-rata persawahan warga tersebar disekeliling desa. Kondisi geografisnya subur dan berbukit-bukit karena diapit oleh dua sugai kecil, yaitu sungai ijo dan sungai manggis. Untuk fasilitas kesehatan tersedia 2 Puskesmas yang berada pada pemukiman warga dan pinggiran Jalan Raya.

2. Andorid

Android merupakan suatu sistem operasi yang berbasis Linux yang digunakan untuk perangkat *mobile* layar sentuh misalnya telepon pintar dan tablet. Sistem operasi ini awalnya dikembangkan oleh *Android, Inc.*, dengan dukungan finansial dari *Google* dan pada tahun 2005 secara resmi membeli perusahaan ini. Secara resmi, android ini diluncurkan pada tahun 2007 dan ponsel pertama android dijual pada tahun 2007 (Wijayanto, 2018).

Kini android terbukti menjadi sistem operasi yang paling banyak digunakan. Jika kita memilih sistem operasi ini maka kita tidak rugi baik sebagai pengguna maupun sebagai developer (pembuat program). Memilih sistem operasi ini bisa dikatakan tepat karena android berada dibawah Google dan pasti sistem operasi ini akan dikembangkan secara terus menerus. Selain itu developer juga bisa secara bebas membuat aplikasi berbasis android karena bersifat open source (Wijayanto, 2018).

Sistem operasi Android terdiri dari beberapa versi. Setiap versi Android terbaru memiliki nama unik tersendiri dan memiliki beberapa jenis kelebihan mulai dari tampilan hingga optimasi keamanan. Berikut merupakan versi versi android yang sudah ada hingga sekarang (Wijayanto, 2018):

- a) Android versi 1.0 dirilis tanggal 23 September 2008.
- b) Android versi 1.1 dirilis tanggal 9 Februari 2009.
- c) Android versi 1.5 Cupcake dirilis tanggal 30 April 2009.

Merupakan versi pertama yang diluncurkan secara komersil dan juga pertama dinamai berdasarkan nama kue. Android mulai menambahkan fitur fitur yang sudah lazim ditemukan saat ini seperti widget, Auto Rotate, dan dukungan keyboard virtual dari pihak ketiga. Dalam tahap ini bisa dikatakan kalau Google masih mencoba untuk memasukkan fitur fitur dasar.

- d) Android versi 1.6 *Donut* dirilis tanggal 15 September 2009.

Pada Android *Donut*, smartphone mendapatkan beberapa fitur tambahan seperti dukungan CDMA, mesin teks ke suara, dan indikator penggunaan baterai. Hal ini membuat versi Android *Donut* lebih sebagai update mini.

e) Android versi 2.0 *Eclair* dirilis tanggal 26 Oktober 2009.

Ada perubahan yang cukup banyak dibandingkan versi Android awal. Pada update ini Google memberikan dukungan dan fitur baru yang utamanya pada aplikasi kamera dengan menambahkan dukungan flash, fokus, hingga efek warna. Pada versi ini Google juga menambahkan Live Wallpaper.

f) Android versi 2.2 *Froyo*, dirilis tanggal 10 Mei 2010

Pada versi ini Android mengalami optimalisasi performa yang cukup signifikan dan ditambah dengan dukungan USB *Thetering*, Wi-Fi, *Hotspot*, dan *push notification* yang sangat berguna hingga saat ini. Selain itu pada versi ini kita bisa memindahkan aplikasi ke penyimpanan *eksternal* jika memungkinkan.

g) Android versi 2.3 *Gingerbread* tergolong paling sukses.

Dirilis tanggal 6 Desember 2010.

h) Android versi 3.0 *Honeycomb* memiliki tampilan mewah dan kinerja paling baik. Dirilis tanggal 22 Februari 2011.

Sistem operasi satu ini dikhususkan untuk perangkat tablet. Pada *Honeycomb* ini beberapa fitur berguna masih

bertahan hingga sekarang seperti *System Bar*, dukungan prosesor *multi-core*, hingga layar *Home* yang dapat dikustomisasi.

i) Android versi 4.0 *Ice Cream Sandwich* sudah mendukung *Flash Player*, dirilis tanggal 19 Oktober 2011.

Google mempersatukan perangkat smartphone dan tablet serta menambahkan tampilan antarmuka yang lebih minimalis. Salah satu fitur baru yang cukup menarik pada Android *Beam* yang memungkinkan transfer data secara tepat menggunakan NFC.

j) Android versi 4.1 *Jelly Bean* memiliki kelebihan di baterai, navigasi gesture, dan kamera. Dirilis tanggal 9 Juli 2012.

Versi ini lebih fokus pada peningkatan kinerja dan keamanan. Sebagian besar peningkatan yang ada pada versi Android ini terjadi pada sistem Android itu sendiri. Namun ada fitur baru yaitu dukungan resolusi *UHD 4K*.

k) Android versi 4.4 *Kitkat* dirilis tanggal 31 Oktober 2013.

Fokus terhadap peningkatan *user experience* dan performa khususnya pada perangkat kelas bawah. Android 4.4 *Kitkat* memberikan batas minimal RAM 512 MB sehingga perangkat yang memiliki RAM dibawah itu digolongkan sebagai perangkat RAM rendah. Fitur terbarunya adalah *screen recording*, *System Wide settings* untuk *closed captioning*, dan lain sebagainya.

l) Android versi 5.0 *Lollipop* dirilis pada 17 Oktober 2014.

Merupakan yang pertama kali mengusung desain material yang masih menjadi tren hingga saat ini.

m) Android versi 6.0 *Marshmallow* dirilis tanggal 28 Mei 2015.

Semakin memperkuat apa yang telah dilakukan oleh Google selama ini melalui Android *Lollipop*. Melalui *system operasi* terbaru ini, Google membawa fitur canggih yang sangat berguna seperti:

- *Support USB Type-C.*
- *Support Fingerprint*
- Daya tahan baterai lebih lama.
- Tambahan fungsi “*Google Now*” yang tidak sekedar melayani perintah suara.

- *Support* sistem pembayaran dengan *Android Pay* yang berkolaborasi dengan *Fingerprint Authentication* sehingga terjamin keamanannya.

n) Android versi *Nougat* dirilis tanggal 22 Agustus 2016.

Memberikan peningkatan terhadap pengalaman penggunaan agar lebih mudah. Fitur dukungan *multi window* menjadi perhatian utama sehingga pengguna dapat menggunakan beberapa aplikasi secara bersamaan

o) Android versi 8.0 *Oreo* dirilis resmi pada tanggal 21 Agustus 2017.

- Android O lebih Fokus pada kecepatan dan efisiensi.
- Kecepatan *Boot up* 2X lebih cepat.
- *Mode Picture in picture* lebih fleksibel.
- Aplikasi yang berjalan di latar belakang lebih

menghemat baterai.

- Baterai lebih tahan lama.

p) Android versi 9.0 *Pie* dirilis tanggal 6 Agustus 2018.

3. Debu

Menurut Kamus Besar Bahasa Indonesia debu adalah sebuah partikel kecil dengan diameter kurang dari 500 mikrometer bisa berasal dari tanah yang kering dan tertiuip angin, sehingga partikel-partikelnya berterbanan di udara (KBBI).

Debu yang berterbangan tersebut dapat berpengaruh terhadap timbulnya penyakit atau gangguan pada saluran pernapasan. Faktor-faktor debu yang meliputi ukuran partikel, bentuk, konsentrasi, daya larut dan sifat kimiawi (Wulandari et al., 2015).

1. Jenis-jenis debu

Jenis-jenis debu dibagi menjadi 3 (Ayu, 2013) yaitu:

a. Debu Partikel

Sumber dari debu partikel ini antara lain debu domestik maupun dari industri, gas buang kendaraan, peleburan timah hitam, dan pabrik batrai.

b. *Benzen*

Sumber dari partikel ini antara lain kendaraan dan daerah Industri.

c. Partikel Polutan

Sumber dari partikel ini antara lain berasal dari daerah yang kurang bersih lingkunganya.

2. Pengaruh fisiologi dari debu

Pengaruh fisiologi dari debu dibagi menjadi 2 (Siswanto, 2019), yaitu:

- a. Pada debu-debu yang menerap pada lokasi saluran pernapasan atas akan memberikan reaksi iritasi (secara ringan) dengan akibat penyakit yang akan ditimbulkan berupa *pharyngitis*. Tetapi bila debu-debu itu dapat mudah dikeluarkan pada reaksi iritasi itu, maka tidak perlu menimbulkan *pharyngitis*.
- b. Untuk debu-debu yang sudah berada dalam jaringan paru-paru maka di sini kemungkinan dapat difagositir oleh makrofaag, atau mengalami filtrasi lewat dinding alveoler masuk dalam saluran dan selanjutnya diselesaikan dalam Reticulo endethelial system pada kelenjer-kelenjer getah bening.

4. Adaptor

Secara umum Adaptor adalah rangkaian elektronika yang berfungsi untuk mengubah tegangan AC (arus bolak-balik) yang tinggi menjadi tegangan DC (arus searah) yang lebih rendah (Loveri, 2017).

5. *Breadboard*

Adalah dasar konstruksi sebuah sirkuit elektronik dan merupakan prototype dari suatu rangkaian elektronik. Istilah ini sering merujuk pada jenis papan tempat merangkai komponen, dimana papan ini tidak memerlukan proses menyolder. Karena papan ini tidak

memerlukan solder sehingga dapat digunakan kembali, dan dengan demikian dapat digunakan untuk prototype sementara serta membantu dalam bereksperimen desain sirkuit elektronika (Mulyana & Kharisman, 2015).

6. *Pengertian Prototype*

Prototype adalah metode proses pembuatan sistem yang dibuat secara terstruktur dan memiliki beberapa tahap-tahap yang harus dilalui pada pembuatannya, namun jika tahap final dinyatakan bahwa sistem yang telah dibuat belum sempurna atau masih memiliki kekurangan, maka sistem akan dievaluasi kembali dan akan melalui proses dari awal (Lugina 2015)

A. Tahapan-tahapan dalam model prototype (Lugina 2015) adalah sebagai berikut :

1) Pengumpulan Kebutuhan

Pada tahap pengumpulan kebutuhan, Pelanggan dan pengembang bersama-sama mendefinisikan format dan kebutuhan keseluruhan perangkat lunak, mengidentifikasi semua kebutuhan, dan garis besar sistem yang akan dibuat.

2) Membangun *Prototyping*

Pada tahap pembangunan *prototyping*, pelanggan dan pembuat sistem bersama-sama membuat format *input* maupun *output* yang akan dihasilkan oleh sistem yang dibuat.

3) Evaluasi *Prototyping*

Selanjutnya, setelah tahap pembangunan *prototyping*, Pelanggan dan pengembang bersama-sama mendefinisikan format dan kebutuhan keseluruhan perangkat lunak, mengidentifikasi semua kebutuhan, dan garis besar sistem yang akan dibuat.

4) Mengkodekan *System*

Dalam tahap ini *prototyping* yang sudah disepakati diterjemahkan ke dalam bahasa pemrograman yang sesuai.

5) Menguji *System*

Pada tahap pengujian *system*, koding yang telah dibuat sebelumnya akan diuji apakah dapat berjalan dengan baik ataupun masih ada bagian-bagian yang perlu diperbaiki atau apakah masih ada bagian yang belum sesuai dengan keinginan pelanggan.

B. Keunggulan dan kekurangan *Prototyping* (Lugina 2015)

- Keunggulan

1. Komunikasi akan terjalin baik antara pengembang dan pelanggan.
2. Pengembang dapat bekerja lebih baik dalam menentukan kebutuhan setiap pelanggannya.
3. Pelanggan berperan aktif dalam proses pengembangan sistem.
4. Lebih menghemat waktu dalam pengembangan sistem.
5. Penerapan menjadi lebih mudah karena pemakai mengetahui apa yang diharapkannya

- Kelemahan

Pelanggan kadang tidak melihat atau menyadari bahwa perangkat lunak yang ada belum mencantumkan kualitas perangkat lunak secara keseluruhan dan juga belum

7. Pengertian Mikrokontroler (Arni, 2018)

Central Processing Unit (CPU) adalah sebuah pusat komputer yang merupakan perangkat keras dari komputer itu sendiri yang bertugas atau berfungsi sebagai penerima dan pelaksana suatu perintah dan data yang ada dari sebuah perangkat lunak. Sedangkan Mikrokontroler merupakan suatu bagian elektronik yang dalam wujudnya seperti *Integrated Circuit (IC)* yang dapat berguna sebagai komputer tetapi dalam ukuran yang kecil atau minim. Kata Mikrokontroler ini ialah gabungan dari dua kata yaitu Mikro dan *Controler*. Mikro berarti yang sangat kecil, sedangkan kontroler berarti sebuah pengendali. Jika kata Mikrokontroler ini menjadi atau berarti suatu unit pengendali yang dalam ukuran yang sangat kecil.

8. Perangkat Keras yang digunakan

Pada Proses Pembuatan dan perancangan alat, perangkat keras (*Hardware*) yang diperlukan antara lain sebagai berikut:

a. Arduino Uno

Arduino Uno adalah papan mikrokontroler berdasarkan ATmega328P. Ini memiliki 14 pin *input / output* digital (6 di antaranya dapat digunakan sebagai *output* PWM), 6 *input* analog, *resonator* keramik 16 MHz (CSTCE16M0V53-R0), koneksi USB, colokan listrik, *header* ICSP, dan tombol *reset* (Arduino, 2020a).

Dalam Arduino Uno yang digunakan dalam penelitian dapat dideskripsikan spesifikasinya dilihat pada tabel 1.1:

Tabel 2.1 Spesifikasi *Board* Arduino Uno R3

No	Mikrokontroler	Uno
1	Tegangan Sumber	5V
2	Input Batasan Tegangan	6-20V
3	Pin <i>Digital</i> I/O	14 (6 PWM output)
4	Pin <i>Digital</i> I/O PWM	6
5	Pin Input <i>Analog</i>	6
6	Arus DC per pin I/O	20mA
7	Arus DC untuk Pin 3.3V	50mA

(Sumber : (Hidayat, 2017))



Gambar 2.1 Arduino Uno R3

(Sumber : Arduino, 2020)

b. Dust Sensor

Sensor debu dengan penginderaan optic sistem.

Diode pemancar inframerah (IRED) dan fototransistor diatur secara diagonal ke dalam perangkat ini. Dapat mendeteksi cahaya debu yang dipantulkan di udara. *Dust Sensor* juga efektif untuk mendeteksi partikel yang sangat halus seperti asap rokok. Selain itu dapat membedakan asap dari debu rumah oleh pola pulsa tegangan *output* (Optical & Sensor, 2006).

Dalam *Dust Sensor* yang digunakan dalam penelitian dapat dideskripsikan spesifikasinya pada tabel 2.2 berikut:

Tabel 2.2 Spesifikasi *dust sensor*

No	Mikrokontroler	<i>Dust Sensor</i>
1	V led	5V (150 ohm resistor)
2	LED-GND	GND
3	LED	Digital pin 2
4	S-GND	GND
5	V0	Analog pin 0
6	Vcc	5V

(Sumber : Hidayat, 2010)



Gambar 2.2 *Dust sensor*
(Sumber : Hidayat, 2010)

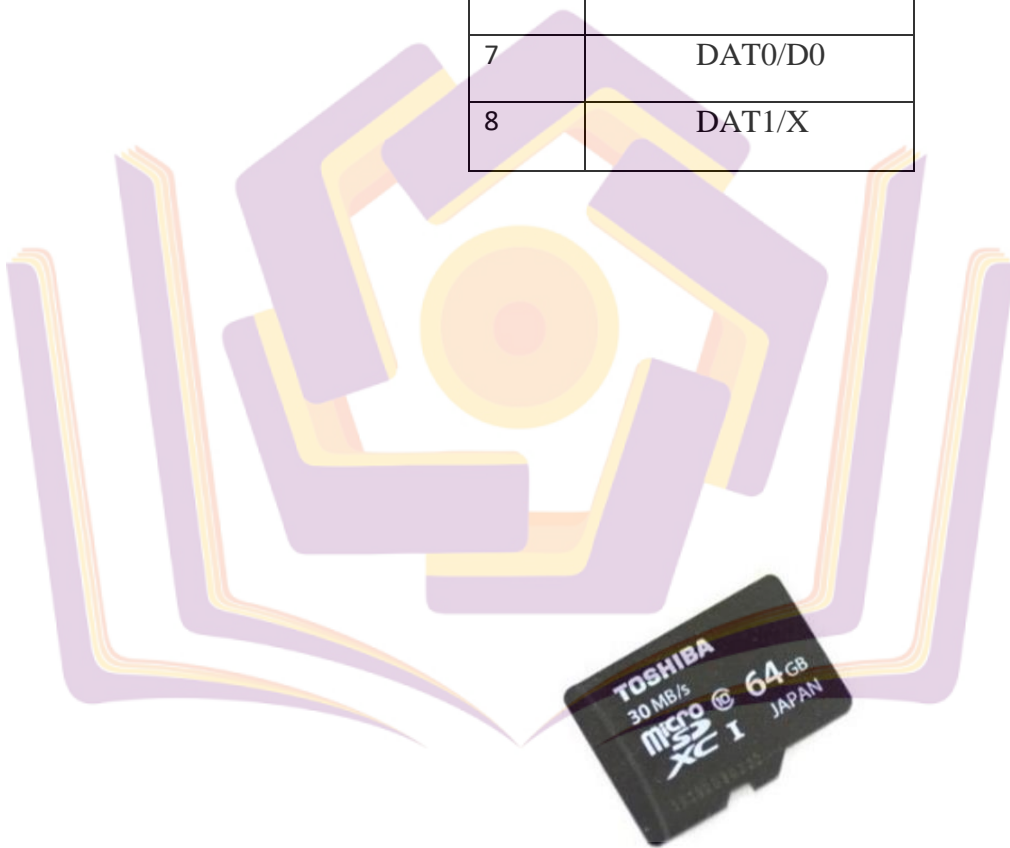
c. *Micro SD – Card*

Micro SD (*Secure Digital*) merupakan kartu memori yang dapat menulis ataupun menghapus data tanpa menggunakan catu daya (*non-volatile*) dikembangkan oleh *SD Card Association* yang digunakan di Indonesia sebagai perangkat keras (*hardware*) *portable*. Saat ini, teknologi micro SD sudah banyak digunakan lebih dari 400 merk produk. Jenis dari *micro SD* terbagi menjadi dua yaitu *micro (Secure Digital High Capacity) SDHC* dan *micro (Secure Digital Extended Capacity) SDXC*, *Micro SDHC* menyediakan kapasitas memori 4GB sampai dengan 32GB (*Giga Byte*). Sedangkan *micro SDXC* menyediakan kapasitas memori lebih dari 32GB (*Giga Byte*) dan hingga saat ini kapasitas *SDXC* sampai dengan 2TB (*Terra Byte*) (Wicaksono and Hirawan 2017)

Dalam *Dust Sensor* yang digunakan dalam penelitian dapat dideskripsikan spesifikasinya pada tabel 2.3 berikut:

Tabel 2.3 Spesifikasi *SD card*

No	Nama Pin
1	DAT2/X
2	DAT3/CS
3	CMD/DI
4	VDD/VDD
5	CLK/SCLK
6	VSS/VSS
7	DAT0/D0
8	DAT1/X



Gambar 2.3 *SD Card*

(Sumber : Wicaksono & Hirawan, 2017)

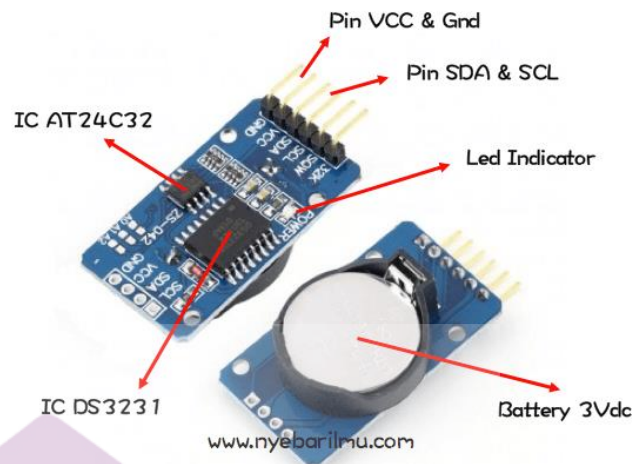
d. Real Time Clock (RTC)

Real time clock (RTC) yaitu sebuah modul yang berfungsi sebagai penghitung waktu yang dirancang menggunakan komponen elektronik berupa *chip* yang mampu melakukan proses kerja seperti jam pada umumnya, seperti melakukan perhitungan detik, menit, dan jam. Perhitungan tersebut dihitung secara akurat dan tersimpan secara *real time*. *Chip* RTC ini nantinya akan diintegrasikan dengan sebuah kontroler dengan melakukan fungsi kerja tertentu (Abdullah, 2018).

Dalam *Dust Sensor* yang digunakan dalam penelitian dapat dideskripsikan spesifikasinya pada tabel 2.4 berikut:

Tabel 2.4 Spesifikasi *Real Time Clock*

No	Nama Pin
1	VCC
2	GND
3	SDA
4	SCL
5	SQW
6	32k



Gambar 2.4 *Real Time Clock*

(Sumber : Faudin, 2017)

e. *Kabel Jumper*

Jumper cable merupakan piranti yang berfungsi sebagai konektor atau penghubung sirkuit elektrik yang digunakan untuk menghubungkan atau memutus hubungan pada suatu sirkuit. *Jumper cable* juga digunakan untuk melakukan setting atau pengaturan pada papan elektrik (Loveri, 2017).



Gambar 2.5 Kable *jumper*

(Sumber : Loveri, 2017)

f. Adaptor

Secara umum *Adaptor* adalah rangkaian elektronika yang berfungsi untuk mengubahtegang AC (arus bolak-balik) yang tinggi menjadi tegangan DC (arus searah) yang lebih rendah (Loveri 2017)



Gambar 2.6 *Adaptor*

(Sumber : Loveri, 2017)

g. SD – Card Module

Proses penyimpanan data hasil dari pengamatan dan pemantauan akan disimpan oleh perangkat *SD* (*Secure Data*) *Card* ini. Kelebihan *SD Card* dibandingkan dengan media penyimpanan lainnya yaitu terletak pada fasilitas untuk membaca data dalam bentuk *file*. *File* yang memiliki format *CSV* (*Command- Separate Value*) pun bisa dibaca dengan mudah (Kadir, 2018)

Dalam *SD Card Module* yang digunakan dalam penelitian dapat dideskripsikan spesifikasinya Pada Tabel 2.5 sebagai berikut:

Tabel 2.5 Tabel spesifikasi *Sd Card Module*

No	SDmodule
1	MISO
2	LED-GND
3	LED
4	S-GND
5	V0
6	Vcc

(Sumber: SenSorApp, 2013)



Gambar 2.7 *SD Card Module*
(Sumber: Utomo & Wirawan, 2018)

B. Penelitian Sebelumnya

Pada Penelitian pertama dilakukan oleh Ratna Susana, Muhammad Ichwan, dan Savero Al Pard yang berjudul Penerapan Metoda *Serial Peripheral Interface (SPI)*. Tujuan dari penelitian tersebut adalah untuk mengetahui karakteristik protokol komunikasi SPI pada *data logger* yang menggunakan *SD Card*, sehingga proses *read / write* data pada *SD card* dapat dilakukan oleh mcikrokontroler. Perbedaanya pada penelitian yang dilakukan *SD card* digunakan untuk menyimpan data yang berformat tipe FAT 16, dan data berupa sekumpulan sensor diambil secara periodik berformat *csv*.

Penelitian yang dilakukan oleh Hermawansa dan Toibah Umi Kalsum dengan judul Analisis Kinerja Sensor Pada Robot Pendeteksi Kotoran Debu Dan Air. Tujuanya adalah untuk menjaga kebersihan lantai rumah dari kotoran seperti tumpahan air dan debu,

sehingga kita harus membersihkan lantai tersebut menggunakan sapu dan kain lap, hal seperti ini memerlukan waktu untuk melakukannya dan terkadang menimbulkan kejenuhan. Perbedaanya adalah pada penelitian ini mendeteksi debu dan air lalu mengambil tindakan membersihkannya.

Penelitian yang dilakukan oleh Agus Sulistiyo dan Suryono dengan judul *Wireless Sensor System Untuk Monitoring Konsentrasi Debu Menggunakan Algoritma Rule Based*. Tujuanya untuk melakukan pemantauan menggunakan metode telemetri, Telemetri memiliki kelebihan melakukan pengukuran dari jarak jauh sehingga tidak perlu memantau pada titik pengukuran. Hampir sama dengan *Monitoring* kualitas udara hanya perbedaanya berada pada alat dan metode yang digunakan yaitu telemetri.

Penelitian yang dilakukan oleh Irma Dita Kurniawati, Ulfa Nurullita, dan Mufbakhuddin dengan judul *Indikator Pencemaran Udara Berdasarkan Jumlah kendaraan Dan Kondisi Iklim*. Tujuan dari penelitian ini adalah untuk mengetahui hubungan jumlah kendaraan dan kondisi iklim dengan konsentrasi CO, mengukur konsentrasi SO₂ dan NO₂ di terminal mangkang dan terminal penggaron Semarang. pada penelitian ini menggunakan *Analitik Observational* dengan pendekatan *Cross Sectional*.

Penelitian yang dilakukan oleh Yanuar Mahir Hermawan, Subagiya, dan Ato Sulistyio dengan judul *Kajian Penggunaan Debu*

Limbah Tembakau Dan Pemberian Vermikompos Terhadap Populasi *Ditylenchus* Pada Bawang Merah. Tujuan dari penelitian ini adalah untuk menurunkan populasi nematoda *ditylenchus* dan nematoda parasite di jaringan tanaman dengan menggunakan debu limbah tembakau dan vermikompos.



BAB III

METODE PENELITIAN

A. Tempat dan Waktu Penelitian

Adapun tempat penelitian Dilakukan di desa Tambak dan waktu penelitian selama 4 bulan, dari bulan April sampai bulan Juli 2020 untuk pencarian komponen yang diperlukan lalu melakukan perakitan supaya menjadi sebuah penelitian.

B. Metode Pengumpulan Data

Metode Pengumpulan data merupakan satu cara yang dipakai oleh para peneliti untuk mengumpulkan, menggali dan mencari semua informasi yang dibutuhkan. Metode pengumpulan data yang dilakukan pada penelitian supaya memperoleh informasi yang dibutuhkan sebagai berikut:

1. Observasi

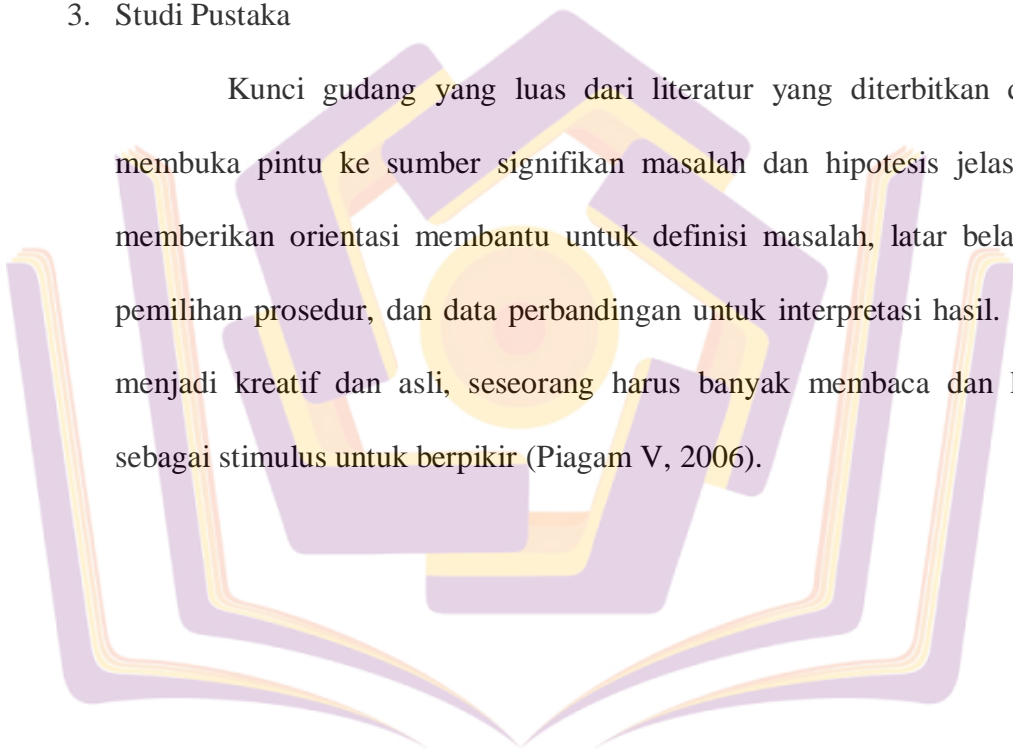
Observasi adalah suatu proses pengamatan dan pencatatan secara sistematis, logis, objektif dan rasional mengenai berbagai fenomena, baik dalam situasi yang sebenarnya maupun dalam situasi buatan untuk mencapai tujuan tertentu (Arifin, 2011).

2. Wawancara

Mendefinisikan bahwa wawancara dengan tujuan percakapan tertentu. Dalam metode ini peneliti dan responden secara langsung (tatap muka) untuk memperoleh informasi secara lisan dengan mendapatkan data tujuan yang bisa menjelaskan masalah penelitian (Moleong, 1991).

3. Studi Pustaka

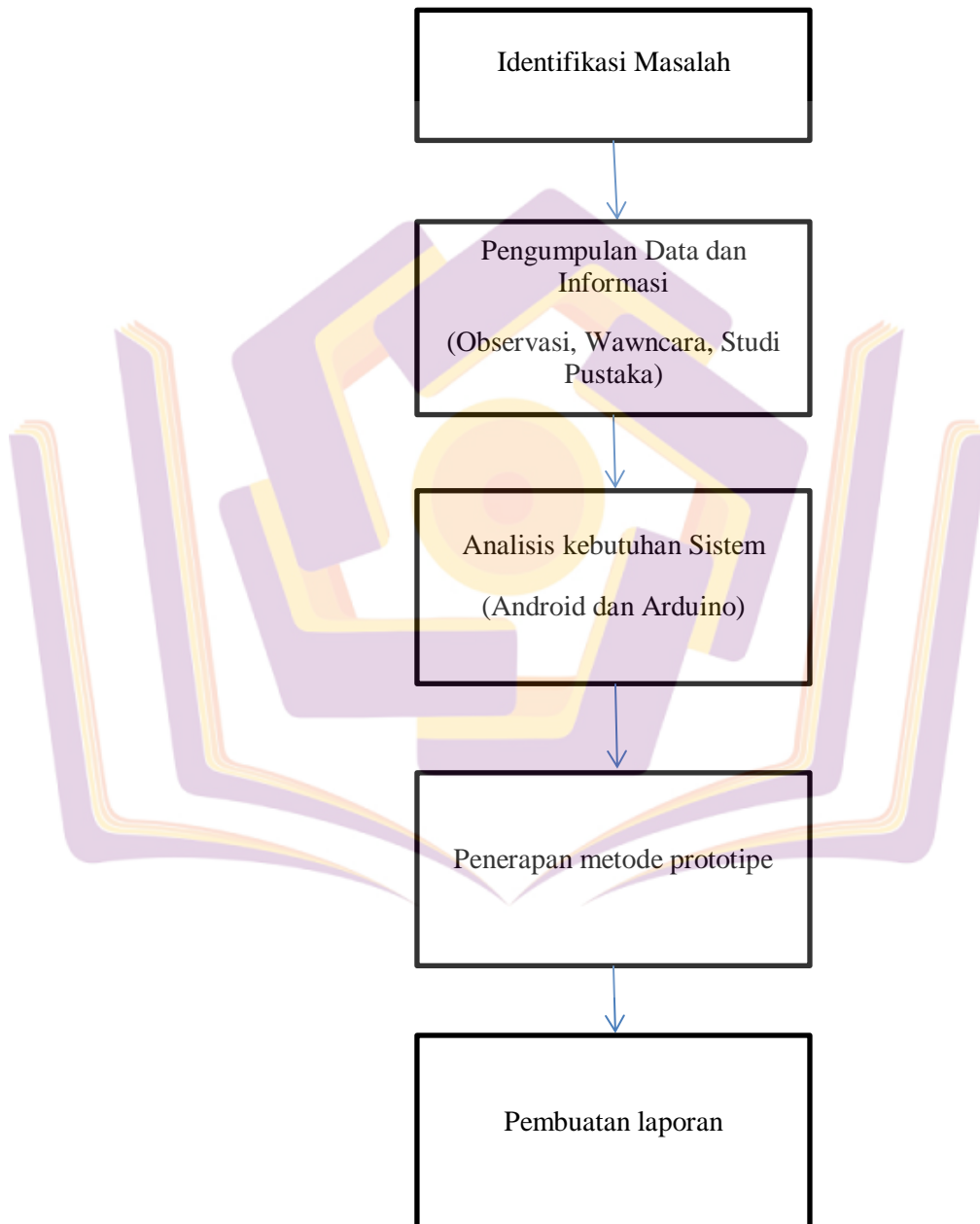
Kunci gudang yang luas dari literatur yang diterbitkan dapat membuka pintu ke sumber signifikan masalah dan hipotesis jelas dan memberikan orientasi membantu untuk definisi masalah, latar belakang pemilihan prosedur, dan data perbandingan untuk interpretasi hasil. Agar menjadi kreatif dan asli, seseorang harus banyak membaca dan kritis sebagai stimulus untuk berpikir (Piagam V, 2006).





C. Konsep Penelitian

1. Kerangka Berpikir



Gambar 3.1 Kerangka Berpikir

a. Identifikasi Masalah

Tahapan ini merupakan awal dari penelitian untuk mengidentifikasi masalah pada objek yang diteliti dan merumuskan untuk dijadikan latar belakang dalam penelitian.

b. Pengumpulan Data & Informasi

Pada tahap ini peneliti akan melakukan pengumpulan data berupa informasi ataupun data yang diperlukan. Adapun beberapa teknik untuk melakukan penelitian sebagai berikut:

1). Observasi

Pada tahap ini melakukan pengamatan terhadap objek yang akan diteliti yaitu jalan raya dekat puskesmas tambak 1 dengan bapak kepala puskesmas.

2). Wawancara

Tahap ini peneliti akan melakukan kegiatan wawancara dengan selaku kepala Puskesmas Tambak 1 untuk mendapatkan informasi maupun data yang diperlukan dalam penelitian.

3). Studi Pustaka

Mengumpulkan data dan informasi yang berhubungan dengan penelitian yang dilakukan.

c. Analisis Kebutuhan Sistem

Setelah memperoleh data yang diperlukan, selanjutnya adalah menganalisis kebutuhan system untuk membuat alat prototype dan system.

1) Analisis Kebutuhan *Arduino*

Menganalisis semua kebutuhan dari *Arduino* mulai dari *software* dan *Hardware*. Pada tahap ini berfungsi sebagai meminimalisir dana yang akan dikeluarkan untuk alat nantinya. Dan memilih *Software* yang nantinya akan digunakan untuk kebutuhan *design* dan penulisan untuk skrip program.

2) Analisis Kebutuhan *Android*

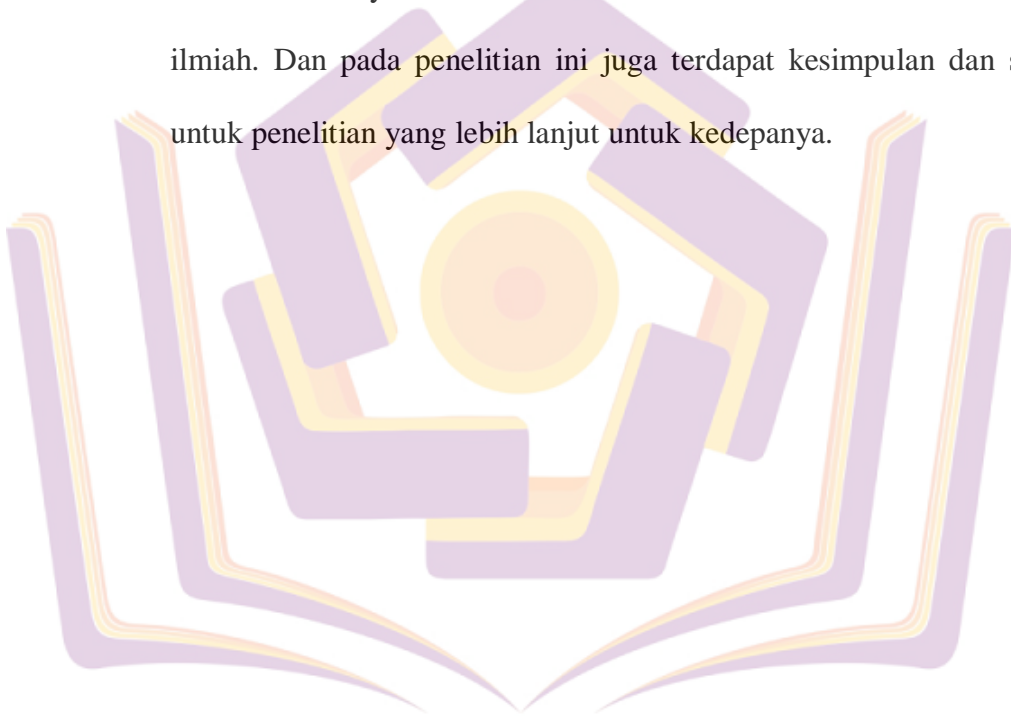
Menganalisis kebutuhan mulai dari tampilan antarmuka hingga fungsi dari aplikasi agar terkoneksi dengan alat.

d. Penerapan Metode Prototipe

Metode Prototipe dapat dimanfaatkan untuk menyelaraskan ketidakpahaman antara pelanggan dengan masalah teknis dan memperjelas spesifikasi kebutuhan yang memang diinginkan oleh pelanggan terhadap pengembang perangkat lunak.

e. Pembuatan Laporan

Pada akhirnya akan di-dokumentasikan ke dalam bentuk tulisan ilmiah. Dan pada penelitian ini juga terdapat kesimpulan dan saran untuk penelitian yang lebih lanjut untuk kedepannya.



BAB IV

HASIL DAN PEMBAHASAN

A. Analisis Hasil

Pada proses pembuatan alat prototipe dan sistem, memerlukan beberapa langkah dalam perancangan, antara lain mendengarkan pelanggan, membangun / memperbaiki, dan pelanggan melihat / menguji. Pada pembahasan ini peneliti akan menganalisis sistem yang telah dibuat untuk melengkapi data yang mendasari sistem yang dirancang, sebagai berikut:

1. Mendengarkan Pelanggan

Mendengarkan masukan pelanggan bertujuan untuk mengetahui masalah yang ada pada *monitoring* kualitas udara. Pada kali ini terdapat 2 tahapan yang dilakukan yaitu mengidentifikasi masalah dan identifikasi penyelesaian masalah.

a. Identifikasi Masalah

- 1) Potensi polusi dan debu pada tiap jamnya.
- 2) Mengetahui tingkat kepadatan pada tiap jam.

b. Identifikasi Penyelesaian Masalah

- 1) *Monitoring* dengan menggunakan prototipe alat ini mulai dari debu hingga polusi akibat gas pembuangan asap kendaraan menggunakan mikrokontroler arduino uno dan dengan beberapa sensor seperti *dust sensor* untuk mengukur kadar debu dan mq-7 untuk mengukur kadar polusi udara dari gas buang asap kendaraan. Serta menggunakan alat

pemantau kualitas udara yang tersambung dengan internet sehingga dapat dipantau lewat *Android* secara *realtime* dan bisa dilihat dari mana saja.

- 2) Pengguna dapat memantau kualitas udara saat ini secara *realtime* menggunakan *Android* dari mana saja yang telah tersambung dengan internet.
- 3) Memberikan data dan informasi tentang kepekatan debu dan polusi pada udara sekitar jalan raya Puskesmas 1 Tambak, sehingga dapat mengetahui tingkat kepekatan di tiap jam nya.

2. Membangun / Memperbaiki *Mock Up*

Bertujuan untuk membangun dan mendesain seluruh kebutuhan dari pengguna serta merevisi atau mengganti kekurangan yang ada pada sistem menurut pengguna.

a. Analisis Kebutuhan Perangkat Lunak (*Software*)

Pada perangkat lunak (*software*) yang akan digunakan untuk merancang sistem dan pembuatan skrip program sebagai berikut:

- 1) Arduino IDE (*Integrated Development Environment*) 1.8.12

Berguna untuk menulis skrip program yang diperlukan, dimana selanjutnya akan di- *upload* ke arduino uno r3.

- 2) Android Studio

Software ini berguna untuk melakukan penulisan skrip program pada *android* yang selanjutnya digunakan untuk pembuatan aplikasi pada *Android*.

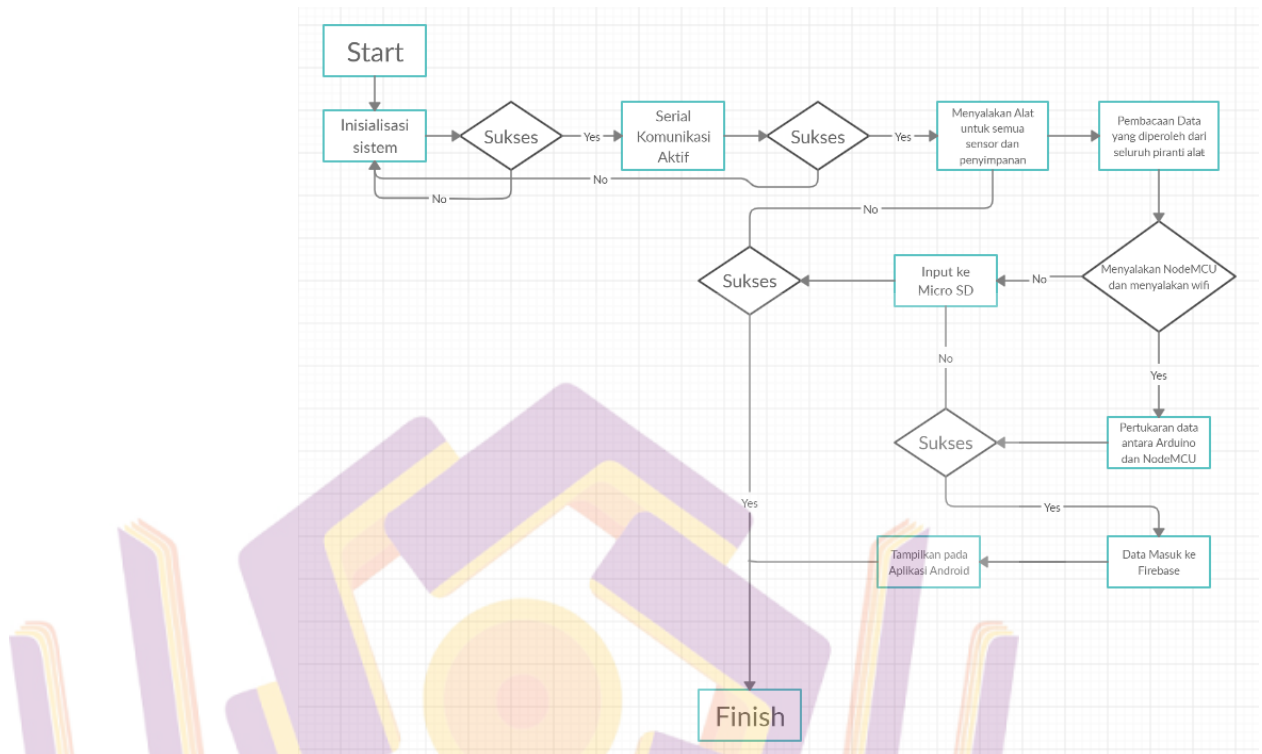
b. Analisis Kebutuhan Perangkat Keras (*Hardware*)

Pada bagian perangkat keras (*Hardware*) yang akan digunakan pada penelitian merupakan alat yang sudah diap digunakan. Pada penelitian ini, peneliti merangkai dan merancang bagian-bagian dari alat yang diperlukan untuk menjadikan sistem monitoring kualitas udara dengan menggunakan *arduino uno r3* yang sudah didukung oleh *dust sensor* dan *mq-7* dan datanya akan tersimpan pada *Micro SD Card*.

Adapun perangkat yang digunakan sebagai berikut:

- 1) *Arduino Uno R3*
- 2) *Dust Sensor*
- 3) *MQ-7*
- 4) *Micro SD Card*
- 5) Modul *Wifi NodeMCU ESP8266*
- 6) *Micro SD Card Module*
- 7) Kabel Jumper (*male to male, male to female, female to female*)
- 8) *Breadboard*
- 9) Adaptor DC 12V 1A
- 10) *Smartphone* berbasis *Android*
- 11) *Laptop*
- 12) *RTC*

a. *Flowchart* Program



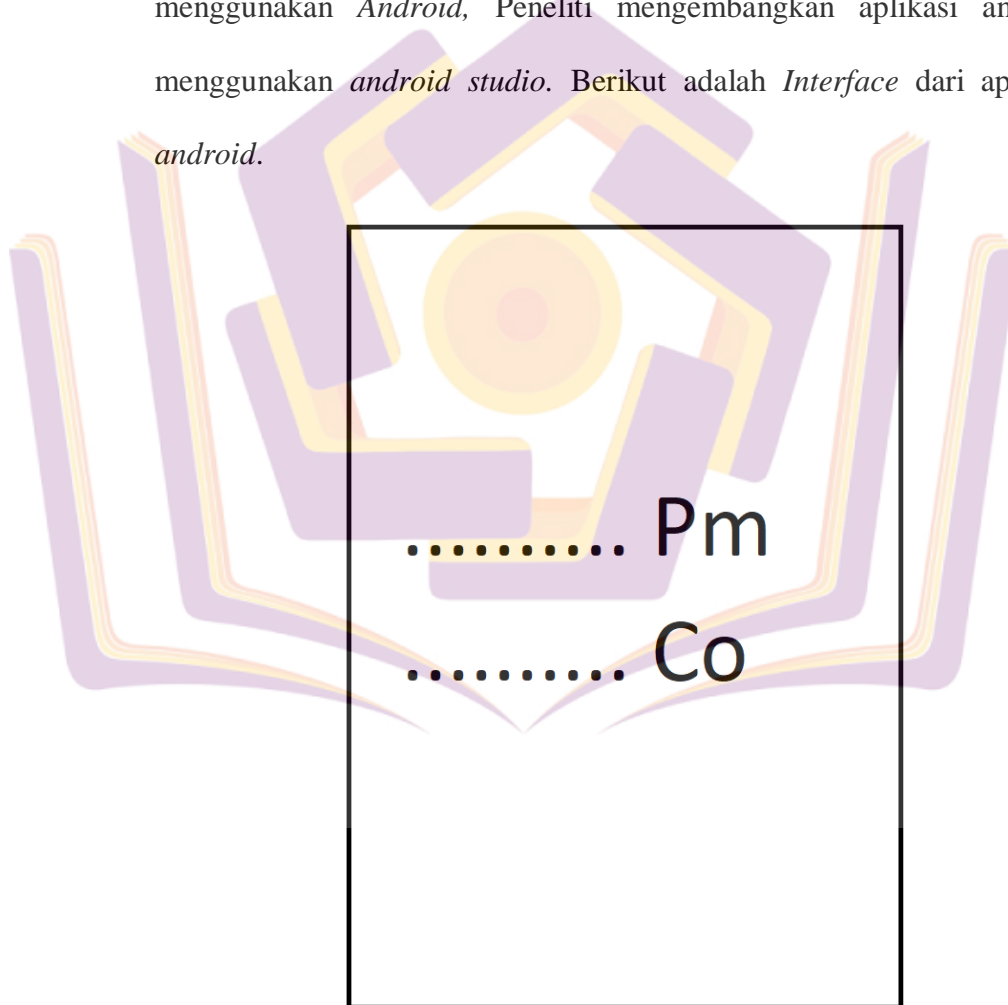
Gambar 4.2 *Flowchart* Program

Pada *flowchart* bisa dilihat pada sistem akan mulai melakukan inisialisasi terlebih dahulu sebelum melanjutkan dengan mengaktifkan alat termasuk semua sensor dan piranti lainnya. Setelah semua menyala akan ada pembacaan data sensor (*dust sensor dan Mq-7*) dan yang terakhir RTC. Lalu dilanjutkan dengan menghubungkan *nodeMCU* dan menyalakan Wifi. Jika tersambung dari Arduino dengan *NodeMCU* maka akan ada komunikasi yang bertukar berupa data yang didapat dari sensor arduino lalu dilanjutkan ke *nodeMCU*. Dan terakhir akan diupload ke *Firestore* saat data telah diterima oleh *nodeMCU* dengan bantuan dari wifi

internet. Setelah sukses *upload* ke *Firebase* maka otomatis data otomatis akan tampil pada Aplikasi *Android*. Dan data juga akan tersimpan pada *micro sd* tiap 1 detik.

b. Perancangan Pada desain *Interface Android*

Untuk desain sangat sederhana untuk monitoring kualitas udara menggunakan *Android*, Peneliti mengembangkan aplikasi android menggunakan *android studio*. Berikut adalah *Interface* dari aplikasi *android*.

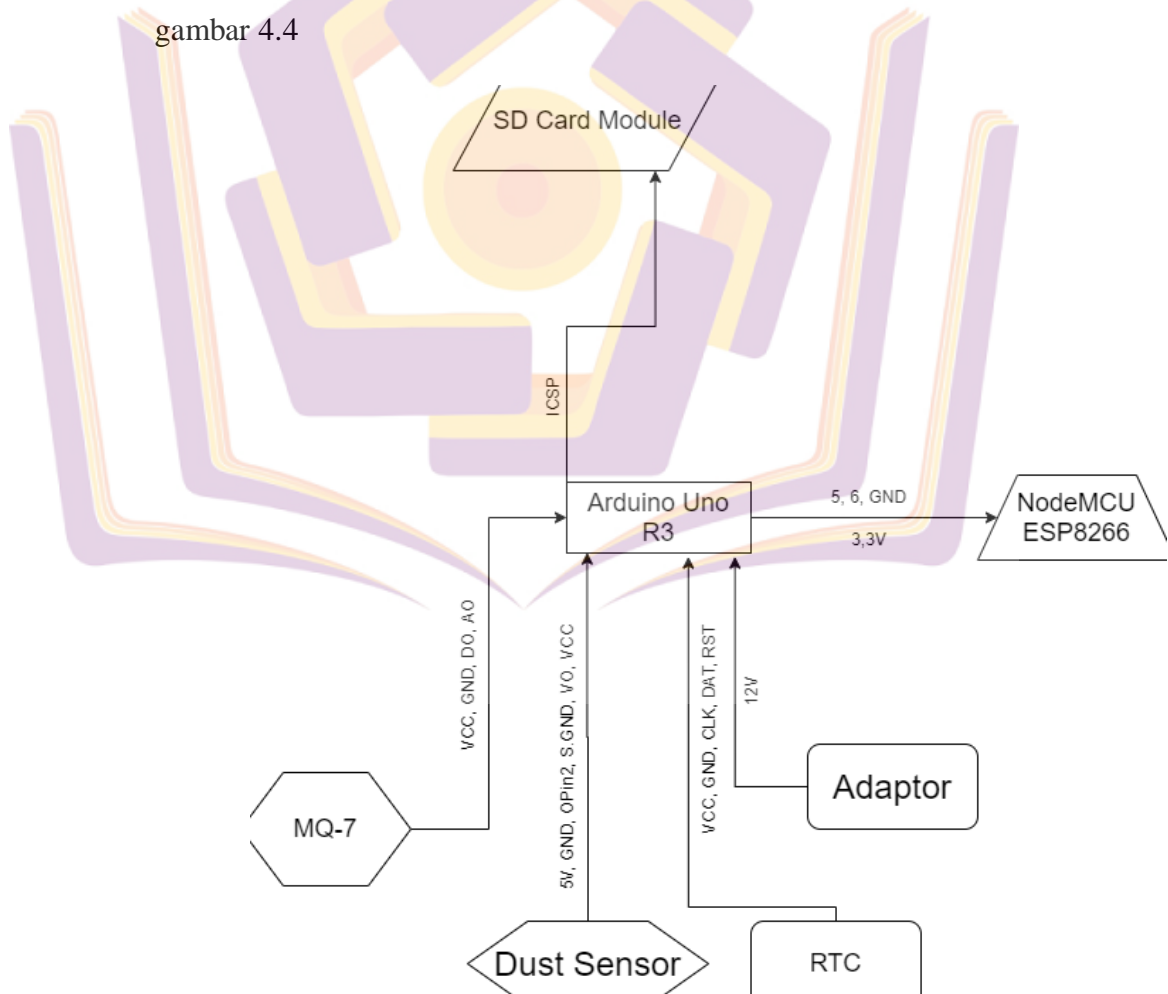


Gambar 4.3 Tampilan *Interface* Android

Untuk desain aplikasi monitoring kualitas udara memiliki tampilan indikator Pm (debu udara) dan Co (karbondioksida). Dan nilai data akan muncul pada titik-titik yang terletak sebelah depan indikator.

c. Perancangan Sistem Monitoring

Perancangan sistem monitoring dapat digambarkan melalui diagram yang bertujuan untuk memudahkan pembuatan rangkaian dari seluruh piranti. Berikut gambar diagram blok yang bisa dilihat pada gambar 4.4



Gambar 4.4 Perancangan Sistem Monitoring

Dari gambar diatas dijelaskan terdapat berbagai komponen piranti yang berperan masing masing sebagai *input*, proses, dan *output*.

1) Perangkat *Input*

a) Adaptor 12V

Merupakan pemberi tegangan utama untuk arduino Uno R3.

b) Adaptor 6V

Merupakan pemberi tegangan utama untuk NodeMCU ESP8266

c) RTC (*Real Time Clok*)

Merupakan komponen untuk membaca tanggal mulai dari hari, bulan, tahun, detik, menit, jam.

d) *Dust sensor*

Komponen ini berfungsi untuk mengukur tingkat kepekatan debu yang melayan pada udara serta memberikan output berupa nilai kepekatan debu.

e) *MQ-7 sensor*

Komponen sensor ini berguna untuk mengukur kepekatan karbon monoksida yang berapa pada udara dengan memberikan output berupa nilai kepekatan karbon monoksida.

f) *SD Card Module*

Komponen ini sebagai penerima data yang selanjutnya akan dipindahkan ke *micro sd card* untuk menyimpan data dari hasil output yang didapat.

g) *NodeMCU ESP8266*

Komponen ini digunakan untuk mengirimkan atau upload data yang dikeluarkan dari output sensor yang kemudian bertukar data dengan arduino uno r3 dan kemudian bertukar secara *realtime*.

2) Proses

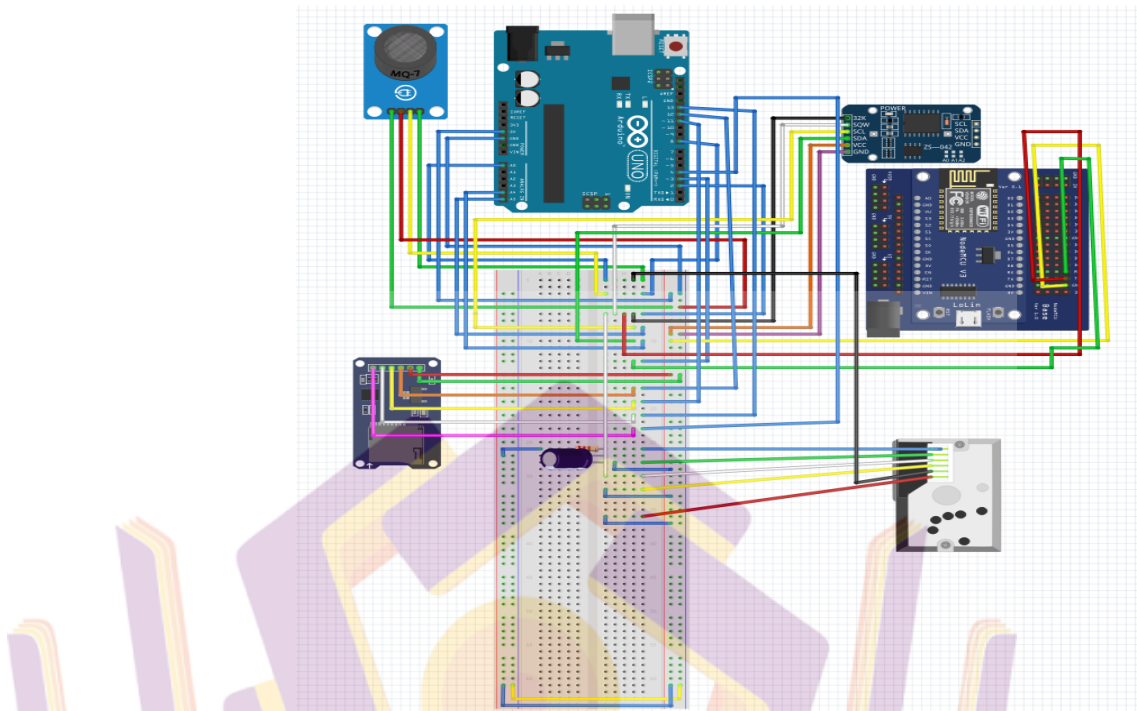
Data yang diperoleh dari (*dust sensor dan MQ-7*) nantinya akan dikirim dan diproses untuk selanjutnya diterima oleh *arduino uno r3*. Setelah *arduino uno r3* menerima data, maka akan diteruskan mengirim data ke *NodeMCU* untuk diupload ke dalam *Firestore* untuk menandakan konsep *Internet of Things* yang dapat dipantau dan ditampilkan pada aplikasi *android*.

3) Perangkat Output

Data nilai berupa kepekatan debu dan polusi gas buang kendaraan yang nantinya akan ditampilkan pada aplikasi *android*.

d. Perancangan Prototipe Alat

1) Skematik Rangkaian



Gambar 4.5 Sketsa Prototipe Alat

Pada gambar diatas merupakan simulasi rangkaian prototipe dari alat *monitoring* mikrokontroler *arduino uno r3* sebagai pusat dari sistem *monitoring* kualitas udara yang juga didukung dengan beberapa sensor yaitu *dust sensor* dan *MQ-7* sebagai sensor untuk membaca kualitas kepekatan debu dan polusi pada udara, serta piranti lainya seperti *RTC* sebagai penentu waktu secara *realtime* dan *SD Card Module* sebagai piranti untuk *save* data yang didapat oleh sensor yang kemudian disimpan pada *SD Card* pada tiap 1 detiknya, Selanjutnya *NodeMCU* yang mendukung akses *Internet of Things* (IoT) dengan menyambungkan *arduino uno r3* dengan *NodeMCU* *ESP 8266* dan dukungan akses *wifi* untuk melakukan pertukaran data.

Untuk output akan muncul pada aplikasi android berupa angka dan satuan Pm dan Co.

2) Perancangan Kotak Pada Alat

Untuk perancangan wadah atau sebagai kotak untuk meletakkan komponen alat menjadi satu dalam sebuah kotak peneliti menggunakan toples yang dimodifikasi untuk meletakkan tiap komponen alat.

3) Rangkaian Alat Prototipe dan Desain Aplikasi Android Secara Keseluruhan

Rangkaian pada alat monitoring kualitas udara telah dibuat dengan menggunakan komponen inti arduino uno r3 dan beberapa komponen lain seperti *dust sensor*, *MQ-7*, *SD Card Module*. Dan RTC menggunakan toples. Pada gambar 4.6 merupakan gambar tampilan dari alat dari belakang yang dibuat, tidak terdapat komponen apapun.



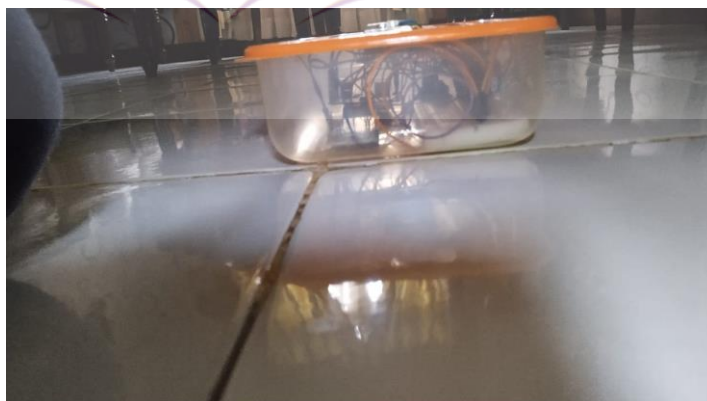
Gambar 4.6 Alat tampak dari belakang

Pada gambar 4.7 merupakan tampilan yang tampak dari arah atas yang telah dibuat. Terdapat *Dust sensor* yang mencatat kepekatan debu dan tutup untuk menutupi komponen yang ada didalamnya.



Gambar 4.7 Alat tampak dari atas

Gambar 4.8 Berikut merupakan tampilan alat dari arah kiri. Pada keseluruhannya tidak terdapat sensor maupun komponen apapun pada bagian kiri alat.



Gambar 4.8 Alat Tampak Samping Kiri

Gambar 4.9 Berikut merupakan tampilan alat dari arah kanan.

Bagian ini terdapat 2 port untuk arduino uno r3.



Gambar 4.9 Alat Tampak Samping Kanan



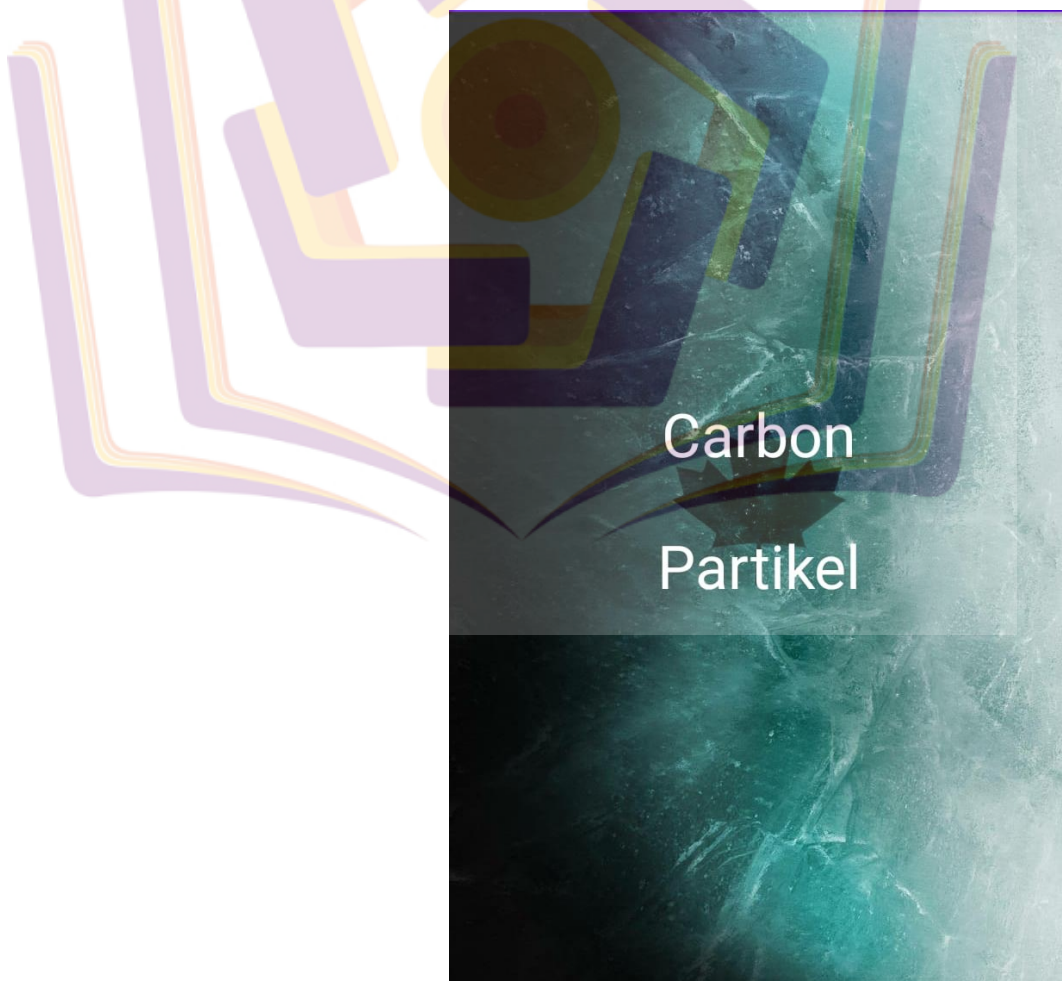
Gambar 4.10 tampilan *interface* Android

e. Tampilan desain aplikasi *Android* setelah perbaikan dari pelanggan

Setelah aplikasi android dibuat dan menunjukkan kepada Bapak dr.Hari Widiyatmoko selaku kepala Puskesmas 1 Tambak, beliau menyarankan untuk menambahkan indikator partikel (debu) dan *Carbon* (karbondioksida) sebelum memulai monitoring, dan selanjutnya untuk penggantian background yang lebih menarik dengan warna yang lebih sederhana namun tetap menarik.

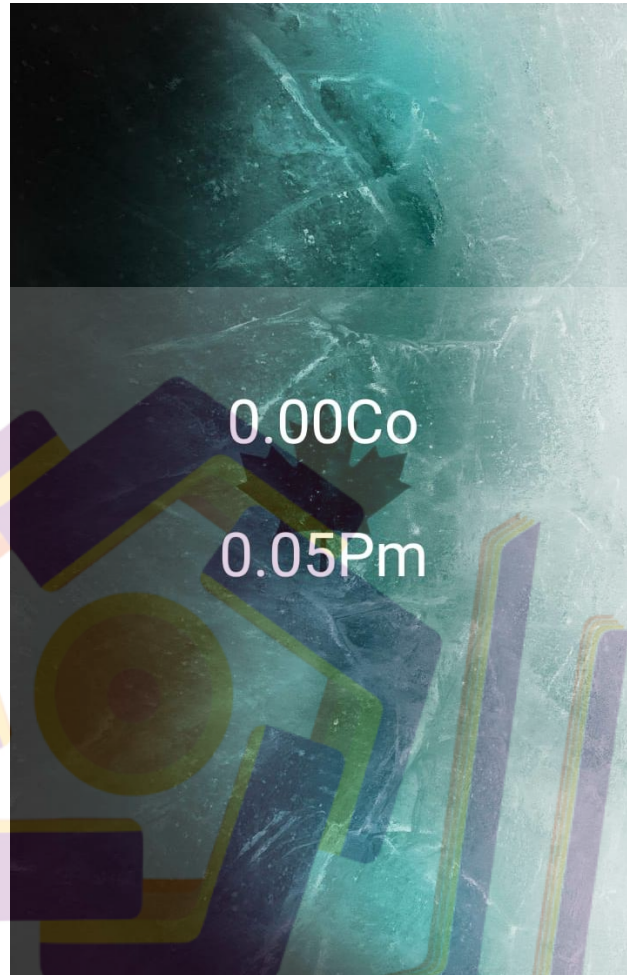
Berikut adalah tampilan pada android setelah melakukan revisi.

- Sebelum memulai monitoring kualitas udara



Gambar 4.11 tampilan *interface* android

- Tampilan sesudah mulai memonitoring

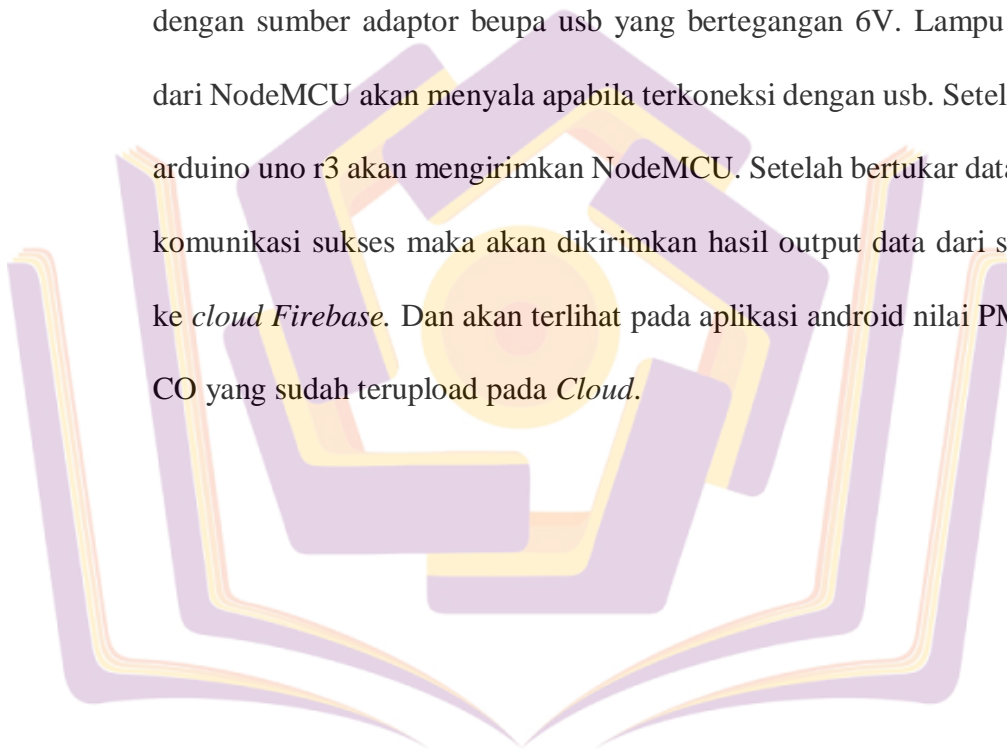


Gambar 4.12 tampilan *interface* Android

f. Cara Kerja Sistem

Arduino uno r3 yang memperoleh supply daya dari adaptor sebesar 12v atau daya dari usb, lampu pada arduino uno r3 akan menyala yang bertanda bahwa alat sudah siap digunakan. Kode program akan mewakili seluruh perintah yang diberikan mengikuti *dust sensor*, RTC, *micro sd card module*, MQ-7. Setelah proses upload melalui *software* Arduino IDE selesai pada arduino uo r3, kemudian akan membaca nilai

data output berupa PM dan CO dari *dust sensor* dan MQ-7 lalu akan menampilkan hasilnya pada aplikasi android. Hasil output data kepekatan debu dan karbon monoksida akan tersimpan pada *micro SD card* menggunakan *micro SD card Module*. Penyimpanan data akan dilakukan setian 1 detik sekali. Untuk selanjutnya monitoring secara jarak jauh, Disambungkan dengan NodeMCU 8266 dan disambungkan dengan sumber adaptor berupa usb yang bertegangan 6V. Lampu LED dari NodeMCU akan menyala apabila terkoneksi dengan usb. Setelah itu arduino uno r3 akan mengirimkan NodeMCU. Setelah bertukar data atau komunikasi sukses maka akan dikirimkan hasil output data dari sensor ke *cloud Firebase*. Dan akan terlihat pada aplikasi android nilai PM dan CO yang sudah terupload pada *Cloud*.



g. Penulisan Kode Program

1) Kode Program Void *Dust Sensor*

```
void dust_sensor() {
  static unsigned long timepoint = millis();
  if (millis() - timepoint > 1000U) {
    timepoint = millis();

    digitalWrite(ledPower, LOW);
    delayMicroseconds(samplingTime);

    voMeasured = analogRead(measurePin);

    delayMicroseconds(deltaTime);
    digitalWrite(ledPower, HIGH);
    delayMicroseconds(sleepTime);

    calcVoltage = voMeasured * (5.0 / 1024);
    dustDensity = 0.17 * calcVoltage - 0.1; // proses dust sensor
  }
  if ( dustDensity < 0)
  {
    dustDensity = 0.00;
  }

  Serial.println("Raw Signal Value (0-1023):");
  Serial.println(voMeasured);

  Serial.println("Voltage:");
  Serial.println(calcVoltage);

  Serial.println("Dust Density:");
  Serial.println(dustDensity);
  dustDensity = pmmilai;
}
```

Pada baris program diatas merupakan proses untuk *dust sensor* yang diperlukan untuk mengaktifkannya dan *timepoint* yang diambil. Lalu untuk parameter yang diambil antara lain voltase, nilai pm, dan *raw signal value* yang kemudian akan dikirimkan sensor ke arduino.

2) Kode Program Void MQ-7

```

void mq7() {
  static unsigned long timepoint = millis();
  if (millis() - timepoint > 1000U) {
    timepoint = millis();

    value = analogRead(AOUTpin); //reads the analaog value from the CO sensor's AOUT pin
    limit = digitalRead(DOUTpin); //reads the digital value from the CO sensor's DOUT pin
  }
  Serial.print("CO value: ");
  Serial.println(value); //prints the CO value
  Serial.print("Limit: ");
  Serial.print(limit);
  value = conilai;
}

```

Pada kode program void diatas merupakan void untuk MQ-7 yang berisikan parameter yang diambil yaitu CO (karbon monoksida) dan selanjutnya timepoint untuk pengambilan data tiap detiknya. Pada void ini merupakan penghubung antara sensor dengan arduino uno r3.

3) Kode Program void Real Time Clock

```

void jam() {
  static unsigned long timepoint = millis();
  if (millis() - timepoint > 1000U) {
    timepoint = millis();
    myRTC.updateTime();
  }
}

```

Pada baris kode diatas merupakan *timepoint* yang diambil untuk menghitung tiap detiknya data yang keluar.

Berikut merupakan baris kode lanjutan untuk *void Real Time Clock*

```

//penulisan data pada serial monitor komputer
Serial.print("Current Date / Time: ");
//fungsi penulisan data untuk tanggal
Serial.print(myRTC.dayofmonth);

//penulisan data "/" sebagai separator
Serial.print("/");

//fungsi penulisan data untuk bulan
Serial.print(myRTC.month);

//penulisan data "/" sebagai separator
Serial.print("/");

//fungsi penulisan data untuk tahun
Serial.print(myRTC.year);

//penulisan data untuk jarak
Serial.print(" ");

//fungsi penulisan data untuk jam
Serial.print(myRTC.hours);

Serial.print(":");

//fungsi penulisan data untuk menit
Serial.print(myRTC.minutes);

Serial.print(":");

//fungsi penulisan data untuk detik
Serial.println(myRTC.seconds);
}

```

Pada baris kode diatas merupakan void dari *Real Time Clock* yang berguna untuk menghitung waktu pengambilan data, pada kode program diatas berisi parameter pencatatan waktu mulai dari detik, menit, jam, tanggal, bulan, tahun.

4) Kode Program *Void SD Card Module*

```

void sd_loop(){
  static unsigned long timepoint = millis();
  if (millis() - timepoint > 1000U) { //time interval: 5s, 300000 = 5 menit, 60000 = 1 menit, 3600000
    timepoint = millis();
    myFile = SD.open("test.txt", FILE_WRITE);
  }
}

```

Pada baris kode diatas merupakan *timepoint* atau pengambilan data perdetik dan nama file data yang tersave sebagai *notepad* yang berformat “*Test.txt*” pada *micro SD card*.

Berikut merupakan baris kode lanjutan untuk void *SD card* module.

```

if (myFile) {
  Serial.print("Writing to test.txt...");
  //myFile.println("testing 1, 2, 3.");
  myFile.print( "Tanggal/Waktu : " );
  myFile.print(myRTC.dayofmonth);
  myFile.print( " : " );
  myFile.print(myRTC.month);
  myFile.print( " : " );
  myFile.print(myRTC.year);
  myFile.print( "/" );
  myFile.print(myRTC.hours);
  myFile.print( " : " );
  myFile.print(myRTC.minutes);
  myFile.print( " : " );
  myFile.println(myRTC.seconds);
  myFile.print( " PPM " );
  myFile.print( " = " );
  myFile.println(dustDensity);
  myFile.print( " Co " );
  myFile.print( " = " );
  myFile.println(value);

  myFile.close();
  Serial.println("done.");
} else {
  Serial.println("error opening test.txt");
}

```

Pada baris kode program diatas merupakan void sd card module yang berisi tentang parameter berupa tahun, bulan, tanggal, jam, menit, detik pencatatan yang diambil RTC untuk selanjutnya disimpan pada *micro SD card*.

5) Kode Program *Firebase Host* dan *Wifi SSID*

```

#include <SoftwareSerial.h>
#include <FirebaseArduino.h>
#include <ESP8266WiFi.h>
#include <ArduinoJson.h>
#define FIREBASE_HOST "sensor-b7e28.firebaseio.com"
#define FIREBASE_AUTH "4fo91b83JCnaxWqxSOMC5faQlPKT5oWfT9BXZIUe"
#define WIFI_SSID "kungkingkang"
#define WIFI_PASSWORD "mogadadi"

```

Baris kode program diatas merupakan penghubung antara *cloud* dengan NodeMCU dengan menggunakan wifi yang digunakan untuk menghubungkan dengan *cloud*.

Berikut merupakan baris kode lanjutan untuk pemrosesan *Micro SD*

```

SoftwareSerial s(D5,D6); //rx tx
String nilaipm, nilaico;
void setup() {
s.begin(115200);
Serial.begin(115200);
// connect to wifi.
WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
Serial.print("connecting");
while (WiFi.status() != WL_CONNECTED) {
Serial.print(".");
delay(500);
}
Serial.println();
Serial.print("connected: ");
Serial.println(WiFi.localIP());
Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
}

```


Baris Kode program diatas berisi tentang *port* yang diambil oleh nodeMCU dan parameter yang nantinya akan diupload (*PM* dan *CO*) lalu *delay* yang ditentukan untuk terkoneksi dengan wifi dan *firebase* atau *cloud* yang digunakan.

6) Baris Kode Aplikasi Android

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:background="@drawable/hp3"
    android:layout_gravity="center_vertical"
    android:layout_centerVertical="true"
    android:gravity="center_vertical"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:orientation="vertical"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <TextView
        android:id="@+id/txtcarbon"
        android:layout_gravity="center"
        android:textSize="35sp"
        android:textColor="#FFFFFF"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Carbon" />
    <Space
        android:layout_width="match_parent"
        android:layout_height="30dp"/>
    <TextView
        android:id="@+id/txtpartikel"
        android:layout_gravity="center"
        android:textSize="35sp"
        android:textColor="#FFFFFF"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Partikel" />
</LinearLayout>

```


Pada baris kode diatas merupakan *activity main.xml* berisikan tentang pengaturan warna, ukuran, dan posisi text saat ditampilkan pada android, selanjutnya pada bagian bawah terdapat tulisan yang nantinya akan dimunculkan sebagai parameter sebelum mulai menghitung kepekatan debu (PM) dan karbon Monoksida (CO).

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.skripsinew2">
    <uses-permission android:name="android.permission.INTERNET" />
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>

```

Pada baris kode diatas merupakan bagian *manifest.xml* dan berisi tentang perijinan untuk mengatur akses aplikasi dengan internet karena aplikasi harus tersambung dengan internet.

```

package com.example.skripsinew2;

import ...

public class MainActivity extends AppCompatActivity {
    TextView txtco;
    TextView txtpm;
    DatabaseReference drfirebase;
    String fireco, firepm, stringco, stringpm;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        txtco = (TextView) findViewById(R.id.txtcarbon);
        txtpm = (TextView) findViewById(R.id.txtpartikel);

        drfirebase = FirebaseDatabase.getInstance().getReference();
        drfirebase.addValueEventListener(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                fireco = dataSnapshot.child("carbon").getValue().toString();
                firepm = dataSnapshot.child("partikel").getValue().toString();

                txtco.setText(fireco + "Co");
                txtpm.setText(firepm + "Pm");
            }

            @Override
            public void onCancelled(@NonNull DatabaseError databaseError) {
            }
        });
    }
}

```

Pada kode program diatas merupakan *main activity.java* yang digunakan untuk mengatur data output yang akan ditampilkan dengan parameter karbon monoksida (CO) dan debu (PM). $\mu\text{g}/\text{m}^3$

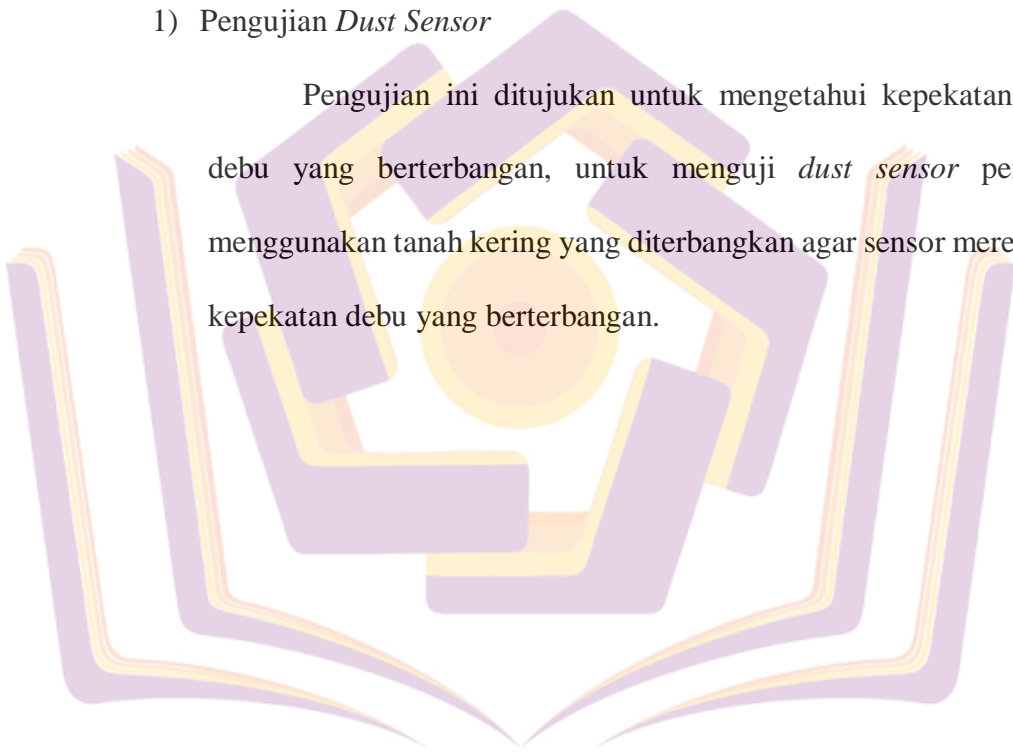
2. Pengujian *Mock Up*

Pengujian *Mock Up* dilakukan untuk mengetahui bagaimana sistem dan alat prototipe yang telah dibuat dalam penelitian ini dapat bekerja sesuai dengan yang diharapkan atau tidak, serta dapat digunakan sebagai parameter keberhasilan suatu sistem. Berikut merupakan pengujian yang akan dilakukan.

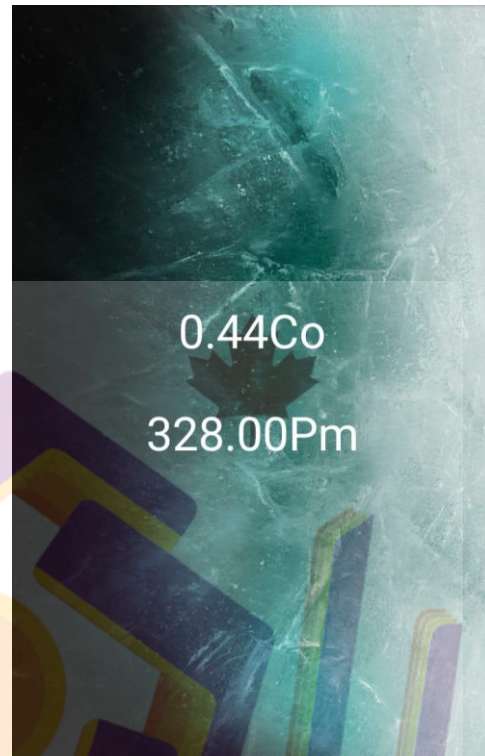
a. Pengujian di Laboratorium

1) Pengujian *Dust Sensor*

Pengujian ini ditujukan untuk mengetahui kepekatan dari debu yang berterbangan, untuk menguji *dust sensor* peneliti menggunakan tanah kering yang diterbangkan agar sensor merespon kepekatan debu yang berterbangan.



i. Pengujian Menggunakan Debu tanah



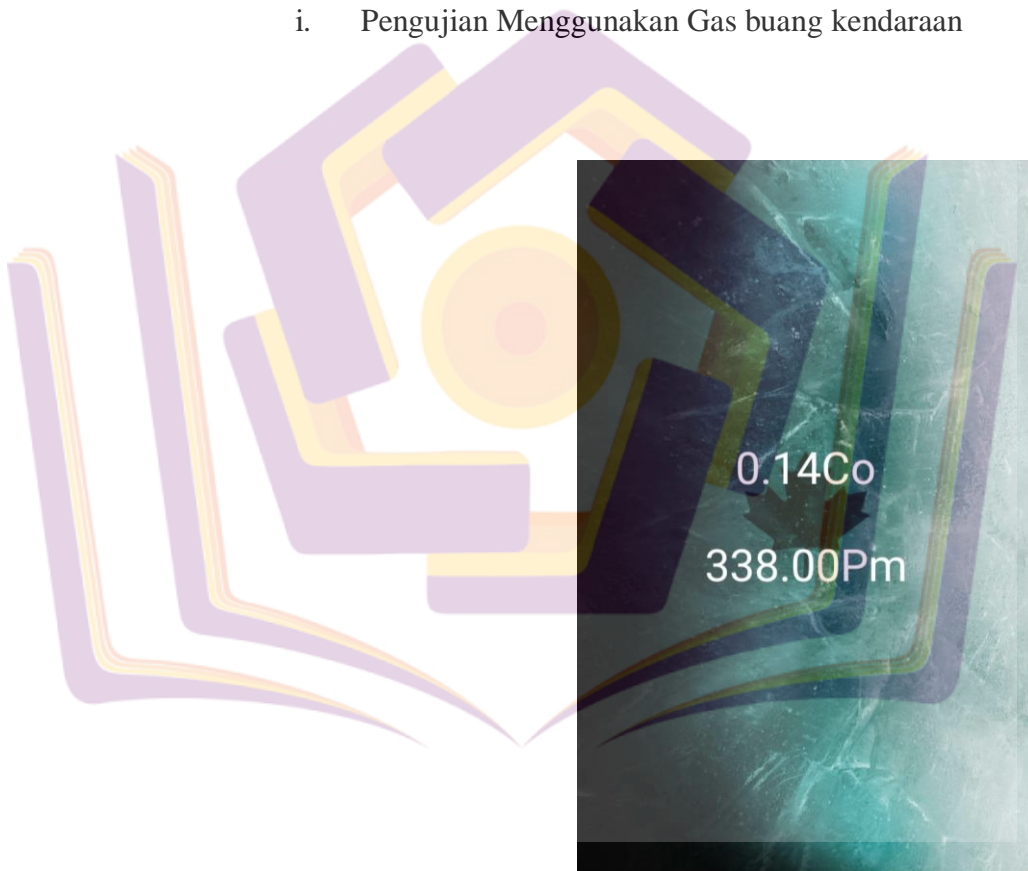
Gambar 4.13 Tampilan Aplikasi Android Pada Pengujian debu tanah

Dari Hasil Pengujian diatas dapat disimpulkan bahwa pengujian debu tanah dan bedak menampilkan angka diatas 70 yang berarti *dust sensor* berfungsi dengan baik.

2) Pengujian sensor MQ-7

Pengujian ini ditujukan untuk mengetahui kepekatan karbon monoksida yang berada pada udara, dan pada pengujian sensor MQ-7 ini peneliti menggunakan gas buang kendaraan. Pada penelitian ini mengamati apakah sensor MQ-7 berfungsi dengan baik atau tidak. Dan berikut adalah pengujian yang dilakukan untuk sensor MQ-7 menggunakan gas buang kendaraan.

i. Pengujian Menggunakan Gas buang kendaraan

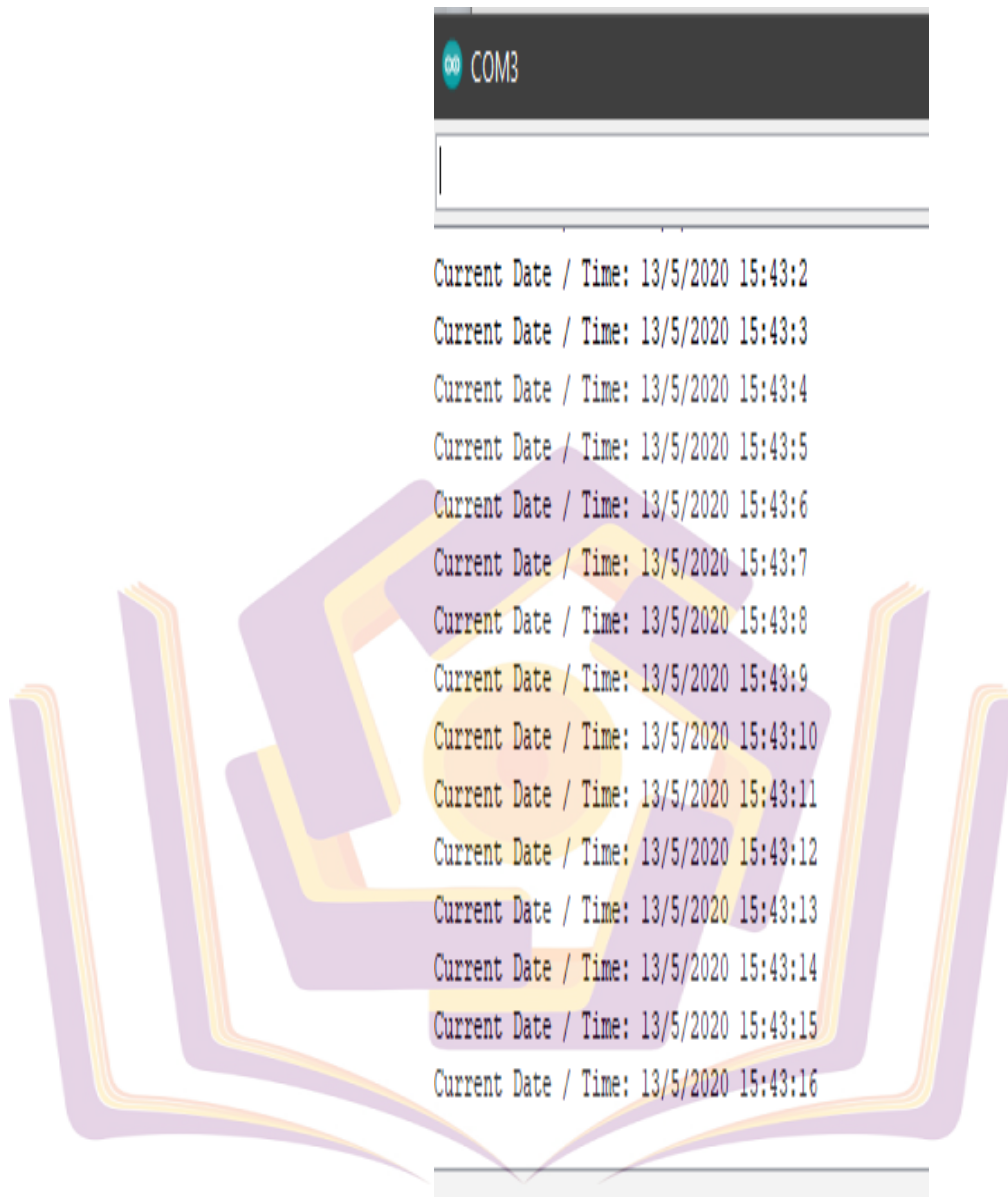


Gambar 4.14 Tampilan Aplikasi Android Pada Pengujian Gas Buang Kendaraan

Dari Hasil Pengujian diatas dapat disimpulkan bahwa pengujian gas buang kendaraan menampilkan angka dibawah 6 yang menunjukkan bahwa udara mengandung gas karbon monoksida. Hal ini bisa disimpulkan bahwa Sensor MQ-7 berfungsi dengan baik.

b. Pengujian RTC (*Real Time Clock*)

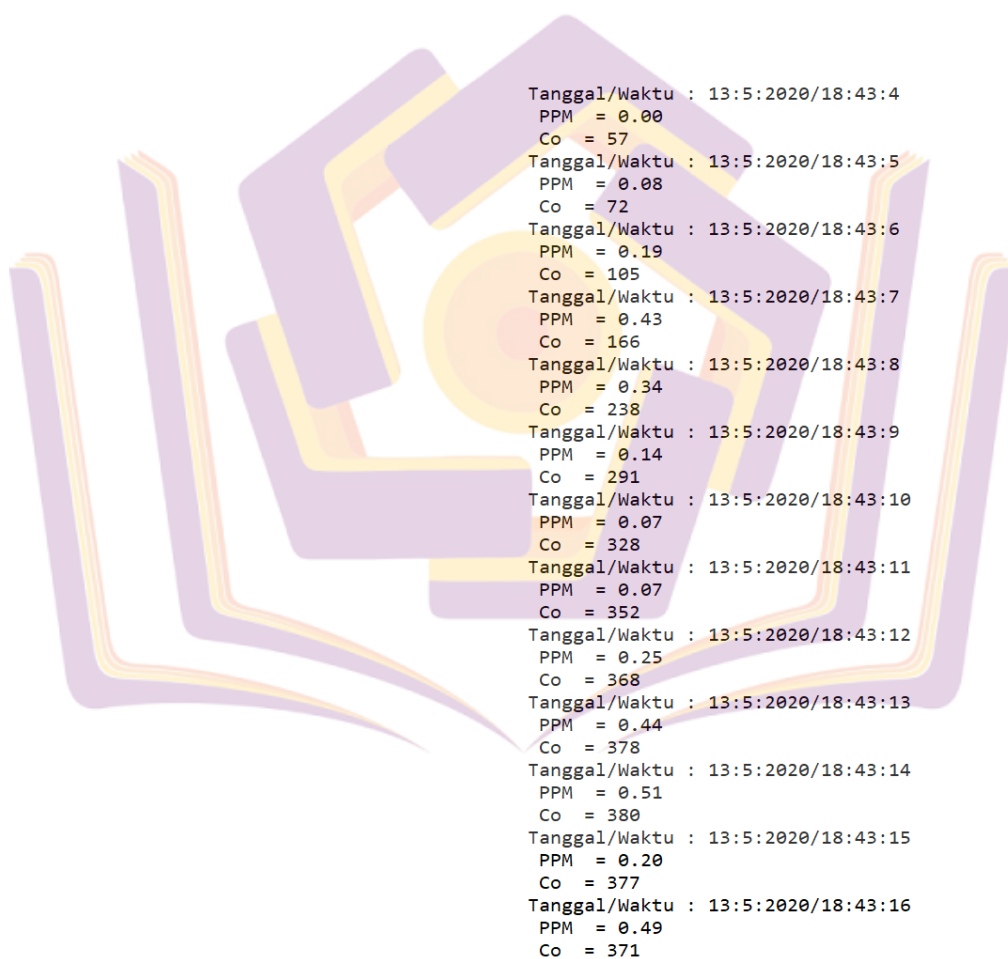
Pengujian ini ditujukan untuk mengetahui apakah RTC (*Real Time Clock*) dapat mencatat dan menampilkan data waktu berupa jam, menit, detik, tanggal, bulan, tahun, Peneliti melakukan pengujian selama beberapa detik untuk mengetahui RTC (*Real Time Clock*) dapat berfungsi dengan baik. Berikut pada gambar 4.? Pengujian yang dilakukan menampilkan waktu.



Gambar 4.15 Hasil Pengujian *Real Time Clock*

c. Pengujian Micro SD Card Module dan Micro SD Card

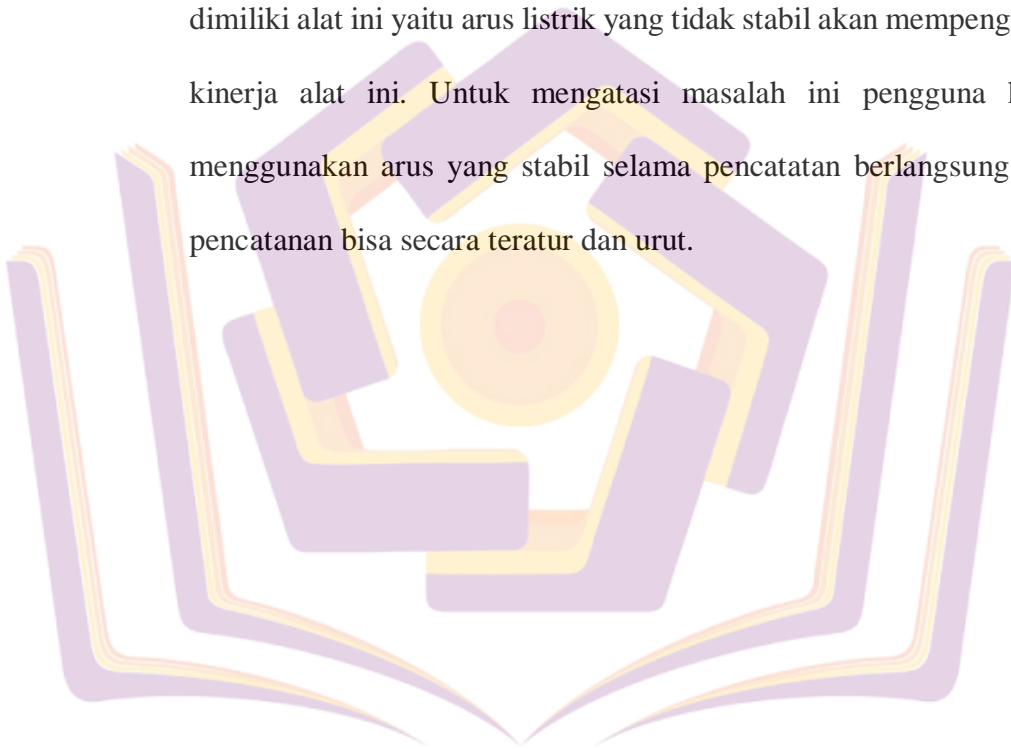
Pengujian ini ditujukan untuk mengetahui apakah sistem dapat menyimpan data yang diperoleh untuk selanjutnya disimpan pada *micro SD card*. Peneliti melakukan pengaturan penyimpanan data ke dalam *micro SD card*. Berikut adalah hasil penyimpanan *File* berformat “*TEST.txt*” yang terlihat pada gambar 4.?



```
Tanggal/Waktu : 13:5:2020/18:43:4  
PPM = 0.00  
Co = 57  
Tanggal/Waktu : 13:5:2020/18:43:5  
PPM = 0.08  
Co = 72  
Tanggal/Waktu : 13:5:2020/18:43:6  
PPM = 0.19  
Co = 105  
Tanggal/Waktu : 13:5:2020/18:43:7  
PPM = 0.43  
Co = 166  
Tanggal/Waktu : 13:5:2020/18:43:8  
PPM = 0.34  
Co = 238  
Tanggal/Waktu : 13:5:2020/18:43:9  
PPM = 0.14  
Co = 291  
Tanggal/Waktu : 13:5:2020/18:43:10  
PPM = 0.07  
Co = 328  
Tanggal/Waktu : 13:5:2020/18:43:11  
PPM = 0.07  
Co = 352  
Tanggal/Waktu : 13:5:2020/18:43:12  
PPM = 0.25  
Co = 368  
Tanggal/Waktu : 13:5:2020/18:43:13  
PPM = 0.44  
Co = 378  
Tanggal/Waktu : 13:5:2020/18:43:14  
PPM = 0.51  
Co = 380  
Tanggal/Waktu : 13:5:2020/18:43:15  
PPM = 0.20  
Co = 377  
Tanggal/Waktu : 13:5:2020/18:43:16  
PPM = 0.49  
Co = 371
```

Gambar 4.16 Hasil penyimpanan *File* format *.txt*

Micro SD Card Module dapat mencatat data ke dalam media penyimpanan *Micro SD Card* setiap 1 detik sekali. Pada penggunaan piranti penyimpanan ini dapat membantu untuk mengetahui kepekatan debu dan karbon monoksida pada udara pada tiap detiknya secara jarak jauh. Pengguna dapat mengambil *micro sd card* untuk mengetahui data pembacaan sensor. Adapun kekurangan yang dimiliki alat ini yaitu arus listrik yang tidak stabil akan mempengaruhi kinerja alat ini. Untuk mengatasi masalah ini pengguna harus menggunakan arus yang stabil selama pencatatan berlangsung agar pencatatan bisa secara teratur dan urut.



BAB V

PENUTUP

A. Kesimpulan

Setelah melakukan pengujian alat monitoring udara berbasis *Arduino UNO* alat ini berjalan dengan lancar. Dan dapat diperoleh kesimpulan antara lain sebagai berikut:

1. Berhasil dibuatnya sistem *monitoring* kualitas udara pada menggunakan *Arduino Uno R3* dengan menggunakan beberapa sensor antara lain *dust sensor* dan *MQ-7* yang dapat dilakukan secara *realtime* dan sudah didukung dengan *Internet of Things* yang dapat dipantau jarak jauh melalui *Android*.
2. Penggunaan alat *monitoring* kualitas udara bisa maksimal untuk *memonitoring* kualitas udara jika penggunaan tidak terlalu lama, karena suhu panas akan berpengaruh terhadap arduino dan akan berakibat pada kualitas pembaca data yang akan mengalami kerusakan. Secara optimal penggunaan hanya bisa sampai kurang dari 50 menit.
3. Pada NodeMCU sebagai modul *wifi* yang berperan sebagai perantara pertukaran data antara alat dengan *firebase* pada internet dapat berkerja dengan baik apabila penggunaan adaptor yang tepat dan stabil pada teganganya.

4. Alat bisa digunakan untuk mengecek kualitas kepadatan debu dan polusi pada udara. 67

B. Saran

Untuk pengembangan alat selanjutnya dari sistem *monitoring* kualitas udara menggunakan mikrokontroler arduino uno r3 untuk lebih baik, maka peneliti ingin memberikan saran yang dapat dilakukan sebagai berikut:

1. Fitur yang dapat ditambahkan dengan mengembangkan suhu dan kelembapan pada udara, sementara untuk alat pada bagian *NodeMCU* masih terdapat beberapa kelemahan seperti kurang *efisien*.
2. Tidak hanya *memonitoring* udara saja, namun dapat dikembangkan untuk *memonitoring* dengan *range* yang lebih jauh dan bisa mendeteksi dengan waktu yang lebih lama dengan ketahanan terhadap hujan dan suhu ekstrim.

DAFTAR PUSTAKA

- Abdullah, M. (2018). Sistem pemberian nutrisi dan penyiraman tanaman otomatis berdasarkan real time clock dan tingkat kelembaban tanah berbasis Mikrokontroler Atmega32. *FISITEK- Jurnal Ilmu Fisika Dan Teknologi*, 2(2), 33–41.
- Arni, D. U. (2018). *Pengertian dan dasar-dasar dari Mikrokontroler*. Diambil dari <https://garudacyber.co.id/artikel/694-pengertian-dan-dasar-dasar-dari-mikrokontroler>, diakses tanggal 10 Maret 2020.
- Fatkhan. (2006). *Pengertian dan definisi kajian pustaka*. Diambil dari <http://fatkhan.web.id/pengertian-dan-definisi-kajian-pustaka/>, diakses tanggal 10 Maret 2020.
- Hermawan, M. Y., Subagiya, S., dan Sulistiyo, A. (1970). Kajian penggunaan debu limbah tembakau dan pemberian Vermikompos terhadap populasi Ditylenchus pada Bawang Merah. *Journal of Chemical Information and Modeling*, 53(9), 1689–1699.
- Hermawansa, H., dan Kalsum, T. U. (2019). Analisis kinerja sensor pada robot pendeteksi kotoran debu dan air. *ILKOM Jurnal Ilmiah*, 11(1), 53–58.
- Hidayat, I. (2017). Prototype Robot penyedot debu berbasis mikrokontroler Atmega328 dan Fuzzy logic dengan kendali Smarthone Android di Universitas Serang Raya. *ProTekInfo (Pengembangan Riset Dan*, 3(1), 36–43.
- Huda, F. A. (2016). *Pengertian dan definisi kajian pustaka*. Diambil dari <http://fatkhan.web.id/pengertian-dan-definisi-kajian-pustaka/>, diakses tanggal 10 Mei 2020.
- Ilham, M. (2020). *pengertian wawancara menurut para ahli*. Diambil dari <https://materibelajar.co.id/pengertian-wawancara-menurut-para-ahli/>, diakses tanggal 10 Maret 2020.
- Kamus Besar Bahasa Indonesia. (2019). *Pengertian debu*. Diambil dari <https://kbbi.kemdikbud.go.id/entri/debu>, diakses tanggal 17 November 2019.
- Kurniawati., Nurullita., dan Mifbakhudin. (2017). Kendaraan dan kondisi iklim (Studi di Wilayah Terminal Mangkang dan Terminal Penggaron Semarang). *Kesehatan Masyarakat*, 12(2), 19–24.
- Lexy, J. L., dan Moleong. (1991). *Metodelogi penelitian kualitatif*. Bandung: Remaja Risdakarya

- Loveri, T. (2017). Rancang bangun pendeteksi asap rokok menggunakan sensor Mq 2 berbasis Arduino. *Jurnal J-Click*, 4(2), 179–185.
- Lugina, R, S. (2019). Rancang bangun multimedia pembelajaran dengan berbantu metode Explicit Intruction untuk mata pembelajaran algoritma dan pemrograman dasar pada materi sorting. *Journal of Chemical Information and Modeling*, 53(9), 1689–1699.
- Mulyana., I. E., dan Kharisman, R. (2015). Perancangan alat peringatan dini bahaya banjir dengan Mikrokontroler Arduino Uno R3. *Creative Information Technology Journal*, 1(3), 171-178.
- Nugraha, W., dan Syarif, M. (2018). Penerapan metode Prototype dalam perancangan Sistem Informasi penghitungan Volume dan Cost penjualan minuman berbasis Website. *JUSIM (Jurnal Sistem Informasi Musirawas)*, 3(2), 94–101.
- Rohmah, K. N (2019). *Apa saja dampak debu terhadap pernafasan*. Diambil dari <https://www.dictio.id/t/apa-saja-dampak-debu-terhadap-pernapasan/121336>, diakses tanggal 4 maret 2020.
- Sharp. (2006). *Dust sensor*. Diambil dari: https://global.sharp/products/device/lineup/data/pdf/datasheet/gp2y1010au_appl_e.pdf, diakses tanggal 8 mei 2020.
- Sulistiyo, A., dan Suryono. (2016). Wireless sensor system untuk monitoring konsentrasi debu menggunakan Algoritma Rule Based. *Youngster Physics Journal*, 5(2), 43–50.
- Susana, R., Ichwan, M., dan Phard, S. A. (2018). Penerapan Metoda Serial Peripheral Interface (SPI) pada Rancang Bangun Data Logger berbasis SD card. *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 4(2), 208-227.
- Utomo, A. P., dan Wirawan, N. A. (2018). Perancangan alat monitoring Air Conditioner menggunakan Mikrokontroler Wemos. *Jurnal Telematika edisi Industrial Engineering Seminar and Call for Paper (IESC)*, dipresentasikan di Institut Teknologi Harapan Bangsa Bandung, pada 29 Oktober 2018.
- Wicaksono, H., dan Hirawan, D. (2017). A Prototype of a Shooting Equipment Using Raspberry Pi Microcontroller and Android. *Jurnal Ilmiah Komputer Dan Informatika (KOMPUTA)*, 1(1), 1-8.
- Yudhanto, Y., dan Wijayanto, A. (2018). *Mudah membuat dan berbisnis aplikasi android dengan android studio*: Jakarta: Elex Media Komputindo.
- Zakky. (2011). *Pengertian observasi*. Diambil dari <https://www.zonareferensi.com/pengertian-observasi/>, diakses tanggal 10 Maret 2020.



LAMPIRAN

Lampiran 1. Wawancara dengan Bapak dr. Harry Widiyatmoko di Puskesmas 1 Tambak dan pengujian alat.



Pengujian alat pada depan Puskesmas 1 Tambak dekat jalan raya



Lampiran 2. Wawancara dengan bapak dr. Harry Widiyatmoko

Wawancara dengan Kepala Puskesmas 1 Tambak pada hari jumat 3 Juli 2020 :

Keterangan : (Narasumber : N, Peneliti : P)

Hasil Wawancara :

Peneliti : Untuk khususnya pasien, penyakit rata-rata yang diderita pasien itu apa saja yang berobat ke puskesmas 1 tambak ini?

Narasumber : Terdapat perubahan sebelum pandemic covid-19 dan sebelum covid-19, sementara penyakit paling tinggi itu adalah penyakit ISPA seperti batuk, pilek, radang tenggorokan. Sementara setelah pandemi rata-rata pasien hanya menderita pegal-pegal.

Peneliti : Untuk total karyawan yang bekerja di Puskesmas 1 Tambak ini ada berapa karyawan?

Narasumber : Total semuanya kita ada 54 orang, termasuk saya sebagai kepala Puskesmas 1 Tambak

Peneliti : Berapa rata-rata jumlah pasien yang datang untuk setiap harinya?

Narasumber : Sejak pandemi pasien yang awalnya 100 orang sekarang turun hingga 50%

Peneliti : Untuk pasien rawat inap disediakan berapa kamar?

Narasumber : Disini disediakan 8 kamar, Setiap 1 kamar berisi 2 kasur mas

Peneliti : Untuk penelitian sebelumnya di Puskesmas 1 Tambak ini sudah berapa banyak pak?

Narasumber : Untuk penelitian sudah ada ratusan mas, ada yang hanya mengambil sampel da nada yang mengambil datanya saja

Lampiran 3. Hasil Kerjasama dengan pihak Puskesmas 1 Tambak



PEMERINTAH KABUPATEN BANYUMAS
DINAS KESEHATAN
PUSKESMAS TAMBAK I
Jl. Raya Barat Tambak No.9 Kode Pos 53196
Telp. (0287) 472495 Email:puskesmas_tambak@gmail.com

SURAT KETERANGAN
440/392 /VII/2020

Yang bertanda tangan dibawah ini:

Nama : dr. HARRY WIDYATOMO
NIP : 198212202010011016
Unit kerja : KEPALA PUSKESMAS TAMBAK I

Menerangkan bahwa:

Nama : FADILA AKTAR ZIYADI
Alamat : Desa Purwodadi RT 03 RW 01, Kecamatan Tambak, Kabupaten
Banyumas
NIM : 16110079
Universitas : AMIKOM PURWOKERTO

Telah melaksanakan penelitian yang berjudul Sistem Pemantauan Kualitas Udara di Jalan Raya Berbasis Mikrocontroler di laksanakan di Puskesmas Tambak I pada tanggal 5 Juni s/d 3 Juli tahun 2020

Demikian surat keterangan ini dibuat untuk dipergunakan sebagaimana mestinya.

Tambak, 3 Juli 2020
Puskesmas Tambak I



dr. Harry Widyatomo
NIP. 198212202010011016

Lampiran 4 Kartu Bimbingan Skripsi

DOKUMEN KARTU BIMBINGAN SKRIPSI	Nomor Dokumen	PP.05/02/55201/AMIKOMPWT
	Revisi	1
	Halaman	1 dari 1
	Tanggal	18 Mei 2013
		Token : 26529

KARTU BIMBINGAN SKRIPSI

Nama Mahasiswa : FADILA AKTAR ZIYADI
 NPM : 16.11.0079
 Program Studi : INFORMATIKA S1
 Judul Skripsi : SISTEM PEMANTAUAN KUALITAS UDARA DI JALAN RAYA BERBASIS MIKROKONTROLER
 Dosen Pembimbing : Hendra Marcos

Histori Bimbingan Skripsi**Bimbingan Ke - : 1**

Hari / Tanggal : Jumat / 20 Maret 2020

Keterangan : 1. BAB I. Perbaiki penulisan Latar belakang masalah

Bimbingan Ke - : 2

Hari / Tanggal : Selasa / 24 April 2020

Keterangan : 1. BAB I. Perbaiki penulisan
 2. BAB II. Gunakan medeley untuk daftar pustaka dan sitasi

Bimbingan Ke - : 3

Hari / Tanggal : Rabu / 15 April 2020

Keterangan : 1. BAB I. Acc
 2. BAB II. Perbaiki penulisan
 3. persiapkan projek

Bimbingan Ke - : 4

Hari / Tanggal : Kamis / 30 April 2020

Keterangan : 1. BAB II. Acc
 2. Lanjutkan BAB III

Bimbingan Ke - : 5

Hari / Tanggal : Senin / 15 Juni 2020

Keterangan : 1. BAB III. perbaiki penulisan di metode pengembangan sistem, diganti sketsa rangkaian alat

Bimbingan Ke - : 6

Hari / Tanggal : Senin / 29 Juni 2020

Keterangan : 1. BAB III. Acc
 2. lanjutkan BAB IV

Bimbingan Ke - : 7

Hari / Tanggal : Kamis / 02 Juli 2020


Keterangan : 1. BAB IV. Acc
 2. BAB V. Acc
 3. Draft Jurnal OK
 4. Projek OK

DOKUMEN KARTU BIMBINGAN SKRIPSI	Unsur Dokumen	PP.65/62/55261/AMIKOMPHIT
	Kejuri	
	Halaman	1 dari 1
	Tanggal	18 Mei 2015
		Token : 41673

KARTU BIMBINGAN SKRIPSI

Nama Mahasiswa : PADHA AETAR ZIVADI
 NPM : 16.11.0079
 Program Studi : PROSMA TEKSI
 Judul Skripsi : SISTEM PEMANTAUAN KUALITAS UDARA DI JALAN RAYA BERBASIS MICROCONTROLLER
 Dosen Pembimbing : Hagus Adh Kusuma

Histori Bimbingan Skripsi

Bimbingan Ke - : 1 


Hari / Tanggal : Sabtu / 28 Maret 2020

Keterangan : Novelty masih kurang memadai, silakan melakukan tinjauan pustaka (LR).

Bimbingan Ke - : 2 


Hari / Tanggal : Senin / 30 Maret 2020

Keterangan : Cari Gap penelitian menggunakan tabel LR

Bimbingan Ke - : 3 


Hari / Tanggal : Selasa / 31 Maret 2020

Keterangan : Perbaiki rumusan masalah

Bimbingan Ke - : 4 


Hari / Tanggal : Selasa / 07 April 2020

Keterangan : Buatlah rincian BoM dan kesiapan dalam merancang alat.

Bimbingan Ke - : 5 

Hari / Tanggal : Rabu / 15 April 2020

Keterangan : Rinci setiap metode pada BAB III dari awal sampai dengan akhir proses penelitian, agar dapat menjadi pedoman penelitian

Bimbingan Ke - : 6 


Hari / Tanggal : Rabu / 06 Mei 2020

Keterangan : Lakukan testbed dengan melengkapi alat.

Bimbingan Ke - : 7 


Hari / Tanggal : Selasa / 12 Mei 2020

Keterangan : Pelajari Multitasking dan serial communication pada arduino

Bimbingan Ke - : 8 


Hari / Tanggal : Senin / 15 Juni 2020

Keterangan : Tambahkan tabel persamaan dan perbedaan penelitian sebelumnya.

Bimbingan Ke - : 9 

Hari / Tanggal : Jumat / 19 Juni 2020

Keterangan : Lakukan simulasi project menggunakan proteus

Bimbingan Ke - : 10 

Hari / Tanggal : Kamis / 25 Juni 2020

Keterangan : Pengujian alat dengan running test.



