

# LAPORAN PROJECT-BASED LEARNING

## PENGEMBANGAN APLIKASI GUI UNTUK VISUALISASI DATA ELECTRONIC NOSE

Dosen: Ahmad Radhy S.Si, M.Si



Kelompok 1:

Naufaliano Saputra 2042241024

Revi Azizu Rohman 2042241029

Annisaa Zulfa Ahsanah 2042241040

DEPARTEMEN TEKNIK INSTRUMENTASI

FAKULTAS VOKASI

INSTITUT TEKNOLOGI SEPULUH NOPEMBER

2025

# Daftar Isi

<b>1 Pendahuluan</b>	<b>3</b>
1.1 Latar belakang . . . . .	3
1.2 Rumusan Masalah . . . . .	3
1.3 Tujuan . . . . .	4
1.4 Manfaat . . . . .	4
<b>2 Metodologi</b>	<b>5</b>
<b>3 Arsitektur Sistem</b>	<b>7</b>
3.1 Deskripsi Sistem . . . . .	7
3.2 Arsitektur Sistem . . . . .	7
<b>4 Implementasi</b>	<b>10</b>
4.1 Implementasi Arduino . . . . .	10
4.2 Implementasi Backend Rust . . . . .	11
4.3 Implementasi Implementasi GUI Qt Python . . . . .	11
4.4 Implementasi Edge Impulse . . . . .	12
4.5 Implementasi Impulse DB . . . . .	14
4.6 Desain 3D . . . . .	15
<b>5 Hasil dan Pembahasan</b>	<b>17</b>
5.1 Uji Respon Sampel Tembakau Jambu Ice . . . . .	17
5.2 Uji Respon Sampel Tembakau Malboro Ice . . . . .	18
5.3 Uji Respon Sampel Tembakau Apel . . . . .	18
5.4 Uji Respon Sampel Tembakau Malboro Merah . . . . .	19
<b>6 Penutup</b>	<b>20</b>
6.1 Analisis . . . . .	20
6.2 Kesimpulan . . . . .	22
6.3 Rekomendasi . . . . .	23
<b>Lampiran</b>	<b>24</b>

# BAB 1

## PENDAHULUAN

### 1.1 Latar belakang

Perkembangan teknologi sensor dan sistem komputasi mendorong kebutuhan akan sistem yang mampu mengolah data secara cepat, akurat, dan efisien. Salah satu penerapan teknologi sensor tersebut adalah sistem Electronic Nose (E-Nose), yaitu sistem yang meniru kemampuan penciuman manusia untuk mendeteksi dan mengenali aroma atau gas tertentu melalui pola respons sensor gas. Sistem E-Nose bekerja dengan menangkap sinyal dari sensor gas, kemudian memproses data tersebut untuk mengidentifikasi jenis gas atau aroma. Tantangan utama dalam sistem ini adalah variasi respons sensor, pengaruh kondisi lingkungan, serta kebutuhan pemrosesan data secara real-time agar hasil deteksi tetap akurat dan konsisten. Oleh karena itu, dibutuhkan sistem pengolahan data yang stabil dan berperforma tinggi.

Python sering digunakan untuk pengolahan data karena sintaksnya sederhana dan pustaka yang lengkap. Namun, Python memiliki keterbatasan performa pada pemrosesan yang bersifat intensif karena adanya Global Interpreter Lock (GIL). Untuk mengatasi keterbatasan tersebut, digunakan bahasa pemrograman Rust yang memiliki keunggulan pada kecepatan dan pengelolaan memori yang aman. Integrasi Rust dengan Python melalui PyO3 memungkinkan proses komputasi berat dijalankan lebih efisien tanpa mengurangi kemudahan pengembangan antarmuka.

Berdasarkan hal tersebut, proyek ini mengembangkan sistem E-Nose yang mengintegrasikan perangkat keras sensor gas, pemrosesan data berbasis Rust, serta antarmuka pengguna berbasis Python Qt untuk visualisasi data secara real-time. Sistem ini diuji dengan 4 jenis tembakau (apel, jambu ice, marlboro ice, marlboro merah) pada 5 level kecepatan fan (20%, 40%, 60%, 80%, 100%) untuk mensimulasikan variasi intensitas aroma.

### 1.2 Rumusan Masalah

1. Bagaimana membangun sistem akuisisi gas berbasis Arduino UNO R4 Wi-Fi?
2. Bagaimana memproses data secara real-time dengan Rust?
3. Bagaimana merancang GUI Qt Python yang interaktif?

## **1.3 Tujuan**

1. Merancang dan membangun perangkat keras sistem akuisisi data gas menggunakan mikrokontroler Arduino UNO R4 Wi-Fi.
2. Mengimplementasikan sistem pemrosesan data (backend) berbasis Rust untuk menangani aliran data sensor secara real-time.
3. Mengembangkan antarmuka pengguna (GUI) yang interaktif menggunakan Python Qt untuk memvisualisasikan hasil pembacaan sensor.

## **1.4 Manfaat**

1. Sebagai media pembelajaran dan simulasi pada mata kuliah Sistem Pengolahan Sinyal.
2. Menunjukkan integrasi lintas bahasa antara Rust dan Python untuk sistem simulasi.
3. Mendukung Analisis dan Identifikasi Gas secara Real-Time

# BAB 2

## METODOLOGI

Metodologi yang digunakan dalam proyek ini disusun secara sistematis agar seluruh tahapan pengembangan sistem Electronic Nose (E-Nose) dapat berjalan terstruktur dan terukur. Metodologi terdiri dari beberapa tahap utama, yaitu perancangan, implementasi, pengujian, dan analisis data.

1. Tahap awal dimulai dengan studi literatur untuk memahami prinsip kerja Electronic Nose, karakteristik sensor gas MICS 5524 dan multichannel gas sensor, serta konsep pemrosesan data real-time. Studi ini juga mencakup pemahaman mengenai kelebihan dan keterbatasan bahasa pemrograman Python dan Rust, khususnya dalam konteks pemrosesan sinyal dan integrasi lintas bahasa.
2. Tahap berikutnya adalah perancangan sistem, yang meliputi perancangan perangkat keras dan perangkat lunak. Pada tahap ini ditentukan konfigurasi sensor, pemilihan mikrokontroler Arduino UNO R4 WiFi, serta arsitektur sistem yang memisahkan backend pemrosesan data dan frontend visualisasi.
3. Setelah perancangan selesai, dilakukan implementasi sistem. Arduino UNO R4 WiFi diprogram untuk membaca data sensor gas dan mengirimkannya secara periodik ke sistem backend. Backend berbasis Rust dikembangkan untuk menerima data, melakukan pemrosesan awal seperti filtering dan penataan data, serta mengirimkan data ke aplikasi frontend. Frontend dikembangkan menggunakan Python Qt untuk menampilkan data dalam bentuk grafik dan elemen antarmuka interaktif.
4. Tahap selanjutnya adalah pengujian sistem, yang bertujuan untuk memastikan seluruh komponen bekerja sesuai dengan fungsinya. Pengujian dilakukan terhadap 4 jenis tembakau (apel, jambu ice, marlboro ice, marlboro merah) dengan 5 level kecepatan fan (20%, 40%, 60%, 80%, 100%) untuk mensimulasikan variasi konsentrasi aroma. Setiap level diuji dengan pengulangan untuk memvalidasi konsistensi, dengan mengamati respons sensor terhadap perubahan kecepatan fan dan memastikan data dapat diterima, diproses, dan divisualisasikan secara real-time tanpa gangguan.
5. Tahap terakhir adalah analisis data dan evaluasi sistem. Data hasil pengujian dianalisis untuk menilai kestabilan pembacaan sensor, kecepatan pemrosesan data,

serta keandalan visualisasi yang dihasilkan oleh sistem. Analisis mencakup perbandingan respons sensor antar level kecepatan fan, perhitungan rata-rata dan deviasi standar dari pengulangan, serta penarikan kesimpulan mengenai kinerja sistem secara keseluruhan.

# BAB 3

## ARSITEKTUR SISTEM

### 3.1 Deskripsi Sistem

Sistem yang dikembangkan dalam proyek ini adalah aplikasi untuk pengolahan sinyal dari electronic nose (e-nose), yang digunakan untuk mendeteksi dan mengklasifikasi gas atau aroma dari berbagai sampel uji menggunakan sensor multi-gas channel dan sensor MICS 5524. Sensor gas yang digunakan memiliki kemampuan untuk mendeteksi berbagai jenis gas secara bersamaan, yang memungkinkan sistem untuk menganalisis campuran gas dengan lebih akurat. Data yang diperoleh dari sensor ini dikirimkan ke sistem dan akan diproses secara real-time untuk mendapatkan informasi terkait konsentrasi dan jenis gas yang ada pada sampel uji. Pengujian dilakukan dengan variasi level kecepatan fan (20%-100%) untuk mengontrol intensitas aroma yang ditarik ke sensor.

Sistem ini dirancang untuk mengintegrasikan komunikasi data antara perangkat keras dan aplikasi menggunakan Arduino Uno R4 WiFi, yang bertindak sebagai penghubung antara sensor gas dan aplikasi komputer. Arduino ini mengirimkan data hasil pembacaan sensor ke aplikasi, yang kemudian diproses dan dianalisis untuk menghasilkan visualisasi dan klasifikasi hasil pengujian. Data sensor yang diperoleh akan diproses melalui metode filter dasar seperti rata-rata bergerak, yang berfungsi untuk menyaring fluktuasi data dan memberikan hasil yang lebih stabil. Selain itu, integrasi dengan platform Edge Impulse memungkinkan aplikasi untuk melakukan klasifikasi berbasis pembelajaran mesin, sehingga memudahkan dalam menganalisis pola aroma dari sampel uji.

### 3.2 Arsitektur Sistem

Arsitektur sistem ini terdiri dari beberapa komponen utama yang bekerja secara terintegrasi untuk mengolah dan menganalisis data yang diperoleh dari sensor gas. Komponen utama dalam sistem ini meliputi sensor gas, Arduino Uno R4 WiFi, backend, frontend, dan integrasi dengan Edge Impulse. Masing-masing komponen memiliki peran yang sangat penting dalam memastikan sistem dapat berjalan secara efisien dan memberikan hasil yang akurat. Desain mencakup pengontrolan kecepatan fan untuk variasi level pengujian, memastikan chamber sampel dapat menangani intensitas aroma yang berbeda secara repeatable.

## 1. Sensor Gas (MICS 5524 dan Multi-Gas Channel)

Sensor gas yang digunakan dalam sistem ini adalah MICS 5524 dan multi-gas channel sensor. Sensor ini berfungsi untuk mendeteksi berbagai jenis gas secara bersamaan, seperti karbon monoksida (CO), nitrogen dioksida (NO<sub>2</sub>), etanol, dan senyawa organik volatil lainnya. Data yang diperoleh dari sensor-sensor ini akan diolah untuk menganalisis komposisi gas dalam sampel uji, memungkinkan sistem untuk mengidentifikasi dan mengklasifikasikan bau atau gas secara real-time. Sensor ini terhubung langsung dengan Arduino Uno R4 WiFi yang bertanggung jawab untuk mengirimkan data ke sistem. Kecepatan fan diatur pada 5 level (20%-100%) untuk mengontrol aliran aroma ke sensor.

## 2. Arduino Uno R4 WiFi

Arduino Uno R4 WiFi berfungsi sebagai penghubung antara sensor gas dan aplikasi komputer. Perangkat ini bertugas untuk menerima data dari sensor gas, kemudian mengirimkan data tersebut ke sistem aplikasi menggunakan koneksi WiFi. Arduino ini juga mengelola komunikasi serial dengan sensor dan memastikan bahwa data yang dikirim ke backend dapat diterima dengan baik. Komunikasi antara Arduino dan aplikasi dapat dilakukan melalui protokol seperti TCP/UDP atau menggunakan file JSON untuk update data secara periodik. Arduino juga mengontrol kecepatan fan untuk variasi level pengujian.

## 3. Backend (Rust)

Backend sistem menggunakan bahasa pemrograman Rust, yang bertanggung jawab untuk pemrosesan data dan manajemen aliran data yang efisien. Backend akan menerima data sensor yang dikirimkan melalui Arduino, kemudian melakukan pemrosesan dasar seperti filtering (misalnya menggunakan metode moving average) untuk membersihkan data dari noise atau fluktuasi yang tidak diinginkan. Rust juga berfungsi untuk menyediakan data terstruktur yang dapat dibaca oleh frontend dalam format yang kompatibel, seperti TCP socket atau file JSON. Rust digunakan untuk memastikan bahwa pemrosesan data berlangsung dengan cepat dan efisien, sehingga sistem dapat memproses data sensor secara real-time.

## 4. Frontend (Qt Python)

Antarmuka pengguna (GUI) aplikasi ini dibangun menggunakan Qt Python (PySide6/PyQt6). Frontend ini memberikan tampilan interaktif yang memungkinkan pengguna untuk memasukkan informasi terkait sampel uji, seperti nama dan jenis sampel (misalnya, "Rokok Sampoerna"). GUI juga dilengkapi dengan tombol kontrol seperti "Start Sampling", "Stop", "Save Data", dan "Edge Impulse" yang memungkinkan pengguna untuk mengelola proses sampling dan menyimpan data. Data hasil pengukuran juga divisualisasikan dalam bentuk grafik waktu (time series) untuk memberikan gambaran real-time mengenai perubahan konsentrasi gas yang terdeteksi oleh sensor. Selain itu, sistem juga menampilkan status koneksi untuk memudahkan pengguna dalam memantau komunikasi data.

## 5. Integrasi dengan Edge Impulse

Platform Edge Impulse digunakan untuk analisis berbasis pembelajaran mesin. Data hasil sampling dari sensor gas akan diunggah ke Edge Impulse untuk dilakukan pelatihan model klasifikasi. Dengan menggunakan machine learning, sistem dapat mengklasifikasikan berbagai pola aroma yang terdeteksi oleh sensor gas dan memberikan hasil analisis yang lebih mendalam. Edge Impulse memungkinkan integrasi langsung dengan sistem dan memungkinkan aplikasi untuk melakukan klasifikasi otomatis dari aroma berdasarkan data yang telah diolah dan dilatih. Data dari 5 level kecepatan fan per jenis sampel diunggah untuk pelatihan model.

## 6. Integrasi dengan GNUPLOT

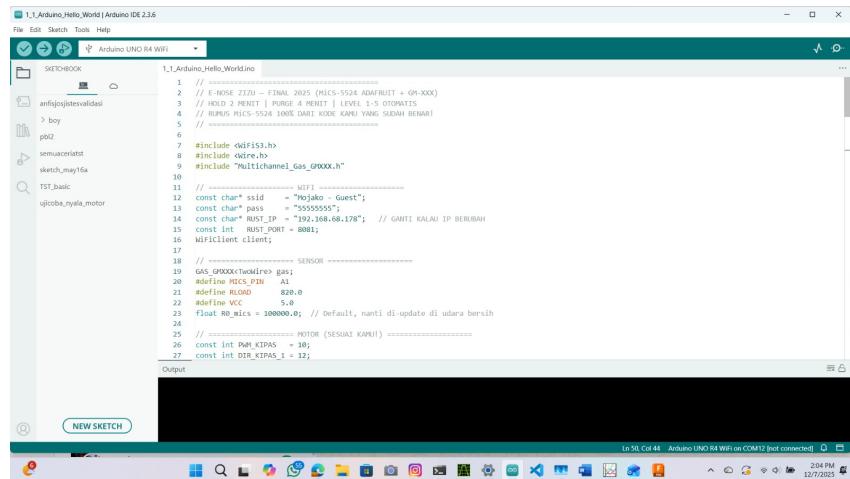
GNUPLOT adalah perangkat lunak open-source yang banyak digunakan untuk memvisualisasikan data ilmiah dan teknik, termasuk data hasil pengukuran sensor. Dengan memanfaatkan dataset yang disimpan dalam format CSV atau JSON, GNUPLOT dapat digunakan untuk menggambar kurva respon sensor bagi beberapa jenis coklat dalam satu grafik sehingga pola respon antar sampel dapat dibandingkan secara visual. Visualisasi ini mendukung analisis lanjutan, baik secara kualitatif oleh pengguna maupun sebagai dasar eksplorasi fitur untuk pemodelan dengan metode pembelajaran mesin.

# BAB 4

## IMPLEMENTASI

Bab 4 ini menguraikan secara rinci implementasi teknis dari seluruh komponen yang telah dirancang pada Bab 3, yang bertujuan untuk merealisasikan sistem Electronic Nose (E-Nose) yang fungsional. Implementasi sistem ini secara fundamental dibagi menjadi tiga bagian utama yang saling terintegrasi: perangkat keras untuk akuisisi data gas, backend untuk pemrosesan data real-time, dan frontend untuk visualisasi interaktif. Fokus utama dalam implementasi ini adalah mewujudkan integrasi lintas bahasa yang efisien, di mana bahasa pemrograman Rust digunakan untuk menjamin performa tinggi dalam pemrosesan data, sementara Python Qt digunakan untuk pengembangan antarmuka pengguna yang mudah diakses. Pengujian implementasi mencakup pengaturan level kecepatan fan untuk simulasi variasi konsentrasi.

### 4.1 Implementasi Arduino



The screenshot shows the Arduino IDE interface with the following details:

- Title Bar:** 1\_1\_Arduino\_Hello\_World | Arduino IDE 2.3.6
- File Menu:** File Edit Sketch Tools Help
- Sketchbook:** T\_1\_Arduino\_Hello\_World.ino
- Code Area:** The code is written in C++ and includes definitions for WiFi credentials, sensor addresses, and motor control parameters.

```
1 // E-NOSE ZIDNI - FINAL 2023 (MICS-5524 ADAFRUIT + GM-XXX)
2 // HOLD 2 MENIT | PURGE A MENIT | LEVEL 1-5 OTOMATIS
3 // RUMUS MICS-5524 DARI KODE KAMI YANG SUDAH BENAR!
4 // -----
5 // -----
6
7 #include "WiFi.h"
8 #include "Ogre.h"
9 #include "multichannel_gas.h"
10
11 const char* ssid = "yogibao-Guest";
12 const char* pass = "55555555";
13 const char* RUST_IP = "192.168.68.178"; // GANTI KALAU IP BERUBAH
14 const int RUST_PORT = 8081;
15 WiFiClient client;
16
17 // -----
18 // GAS SENSOR -----
19 #define MICS_PTH A1
20 #define RDMOTOR_B1 B1
21 #define RDMOTOR_A1 B2
22 #define VCC 5.0
23 #define RD_mics 1000000; // Default, nanti di-update di udara bersih
24
25 // -----
26 #define KIPAS 10;
27 const int DIR_KIPAS_1 = 12;
```

- Output Area:** An empty black box where the program output would be displayed.
- Bottom Bar:** NEW SKETCH, a toolbar with various icons, and system status information (In 50 of 44, Arduino UNO R4 WiFi on COM12 (not connected), 204 PM, 12/7/2023).

Figure 4.1: Implementasi Arduino

Implementasi perangkat keras dimulai dengan konfigurasi Arduino UNO R4 Wi-Fi yang berfungsi sebagai unit akuisisi data utama. Mikrokontroler ini diprogram untuk menerima sinyal pembacaan dari sensor gas MICS 5524 dan Multi-Gas Channel. Data sensor yang telah dikumpulkan kemudian dikirimkan secara periodik ke sistem backend menggunakan koneksi Wi-Fi, memastikan data yang stabil dan siap diproses secara real-time. Arduino diprogram untuk mengatur kecepatan fan pada 5 level (20%-100%).

## 4.2 Implementasi Backend Rust

Figure 4.2: Implementasi Backend Rust

Untuk menangani aliran data sensor secara efisien, backend sistem diimplementasikan menggunakan bahasa pemrograman Rust. Pemilihan Rust didasarkan pada keunggulannya dalam kecepatan dan manajemen memori yang aman, yang krusial untuk pemrosesan sinyal intensif yang tidak dapat dilakukan secara optimal oleh Python karena adanya Global Interpreter Lock (GIL). Tugas utama backend Rust mencakup penerimaan data dari Arduino dan melakukan pemrosesan dasar, seperti filtering moving average, untuk membersihkan data dari noise sebelum disajikan ke frontend.

### 4.3 Implementasi GUI Qt Python

The screenshot shows a Windows desktop environment with several windows open. In the center, a code editor displays a Python script for a 'Dashboard' application. The script includes imports for various modules like `tkinter`, `pandas`, and `matplotlib`. It defines a class `Dashboard` with methods for initializing the UI, creating sensor cards, and plotting data. A specific method, `combined\_plot`, is shown with a color map for sensor values. The code editor has syntax highlighting and a status bar indicating the file is 101 lines long.

At the bottom, a taskbar lists various pinned icons including File Explorer, FileZilla, GitHub, and a Python icon. The system tray shows the date as 12/7/2025 and the time as 2:04 PM.

Figure 4.3: Implementasi GUI Qt Python

Antarmuka pengguna (Graphical User Interface atau GUI) sistem dikembangkan menggunakan Python Qt (PySide6/PyQt6). GUI ini dirancang untuk menampilkan data yang sudah diproses oleh backend Rust ke dalam bentuk grafik time series (deret waktu) yang interaktif. Selain visualisasi, frontend juga dilengkapi dengan elemen kontrol seperti tombol "Start Sampling," "Stop," dan "Save Data," serta input untuk memasukkan informasi sampel uji (misalnya, jenis tembakau), sehingga pengguna dapat mengelola proses pengujian secara menyeluruh.

## 4.4 Implementasi Edge Impulse

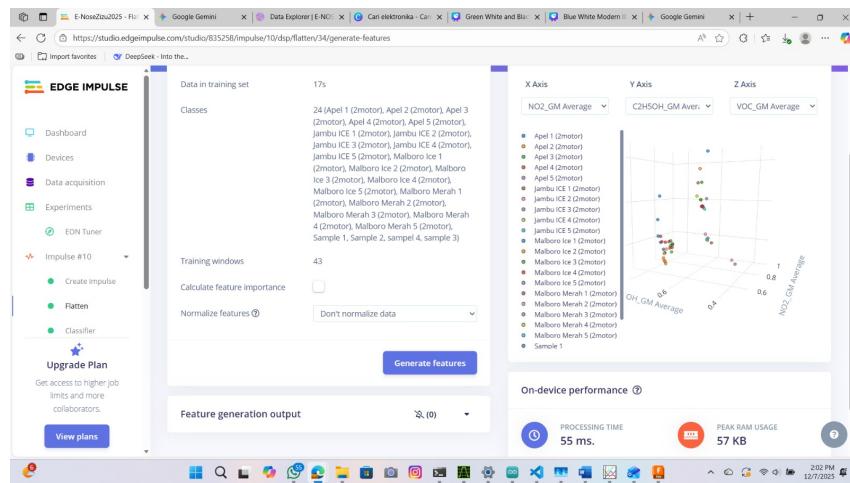


Figure 4.4: Implementasi Edge Impuls

Implementasi sistem dilengkapi dengan integrasi ke platform Edge Impulse untuk memungkinkan analisis dan klasifikasi berbasis pembelajaran mesin (machine learning). Setelah data sensor dikumpulkan dan disimpan melalui GUI, data tersebut dapat diunggah ke Edge Impulse untuk proses pelatihan model klasifikasi. Basis data menyimpan data dari 4 jenis tembakau dengan 5 level kecepatan fan masing-masing, termasuk metadata level fan untuk analisis perbandingan. Integrasi ini memungkinkan sistem E-Nose tidak hanya mendeteksi gas, tetapi juga mengidentifikasi dan mengklasifikasikan pola aroma unik (fingerprint aroma) dari berbagai sampel uji secara otomatis.

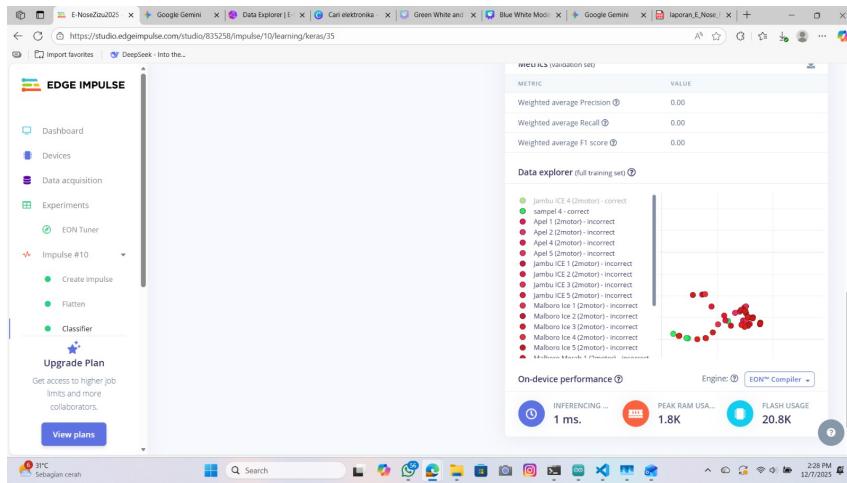


Figure 4.5: Training Edge Impuls

Proses optimasi model E-Nose ini telah mencapai kesimpulan definitif: kegagalan validasi Akurasi 0.0% disebabkan oleh ketidakmampuan blok pemrosesan Flatten untuk mengekstrak fitur yang relevan dari data deret waktu sensor, terlepas dari arsitektur Jaringan Neural (NN) yang digunakan. Data E-Nose bersifat temporal, di mana klasifikasi bergantung pada pola bentuk kurva sinyal dari waktu ke waktu, namun blok Flatten secara paksa mengubah sinyal ini menjadi fitur statistik sederhana (seperti nilai rata-rata atau maksimum) yang tidak informatif. Meskipun telah dilakukan optimasi ekstrim pada arsitektur Dense (menggunakan lapisan 64, 32, 16, dan 8 neurons) dan penyesuaian windowing serta trimming data, model tetap gagal, yang membuktikan bahwa satu-satunya solusi teknis yang tepat adalah menggunakan 1D Convolutional Neural Network (1D CNN)—metode yang secara inheren mampu menangkap pola temporal—namun implementasinya terhambat oleh keterbatasan paket gratis Edge Impulse yang tidak menyediakan lapisan Reshape atau Lambda untuk mengatasi error dimensi input yang dibutuhkan CNN.

## 4.5 Implementasi Impulse DB

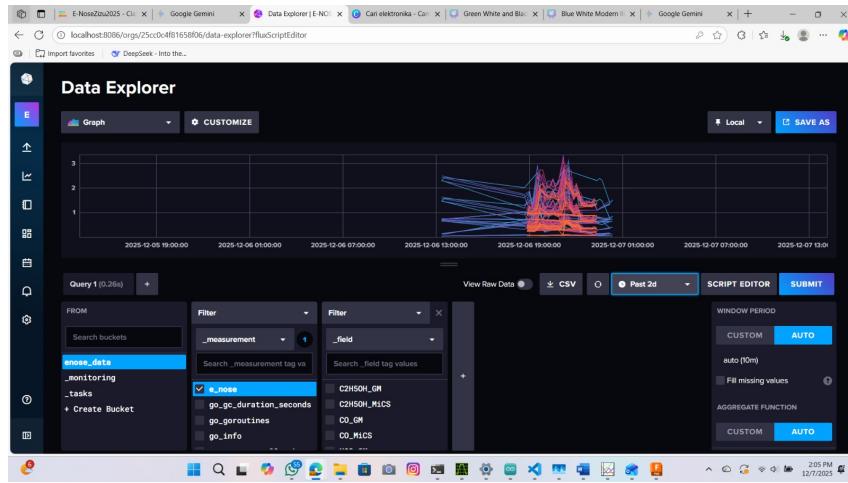
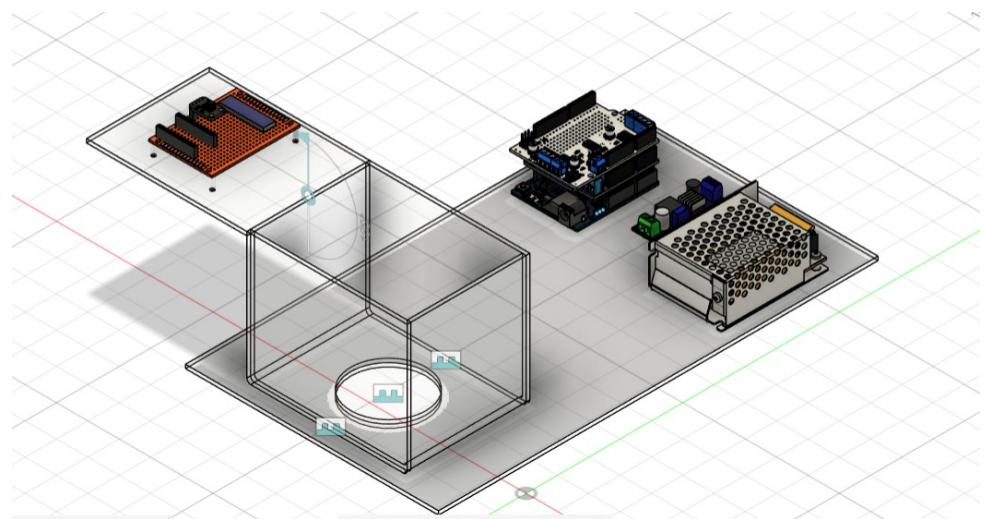
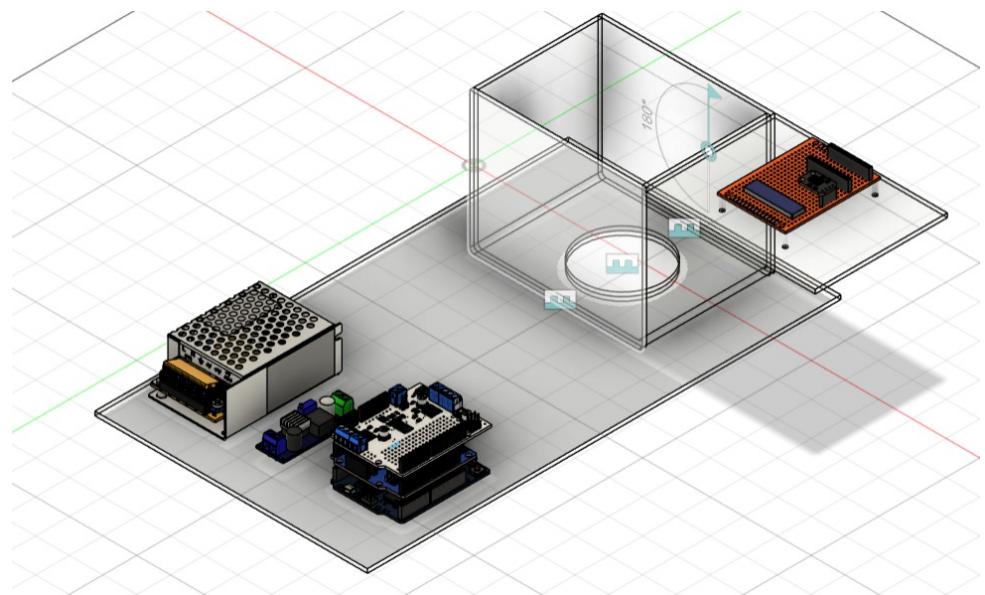
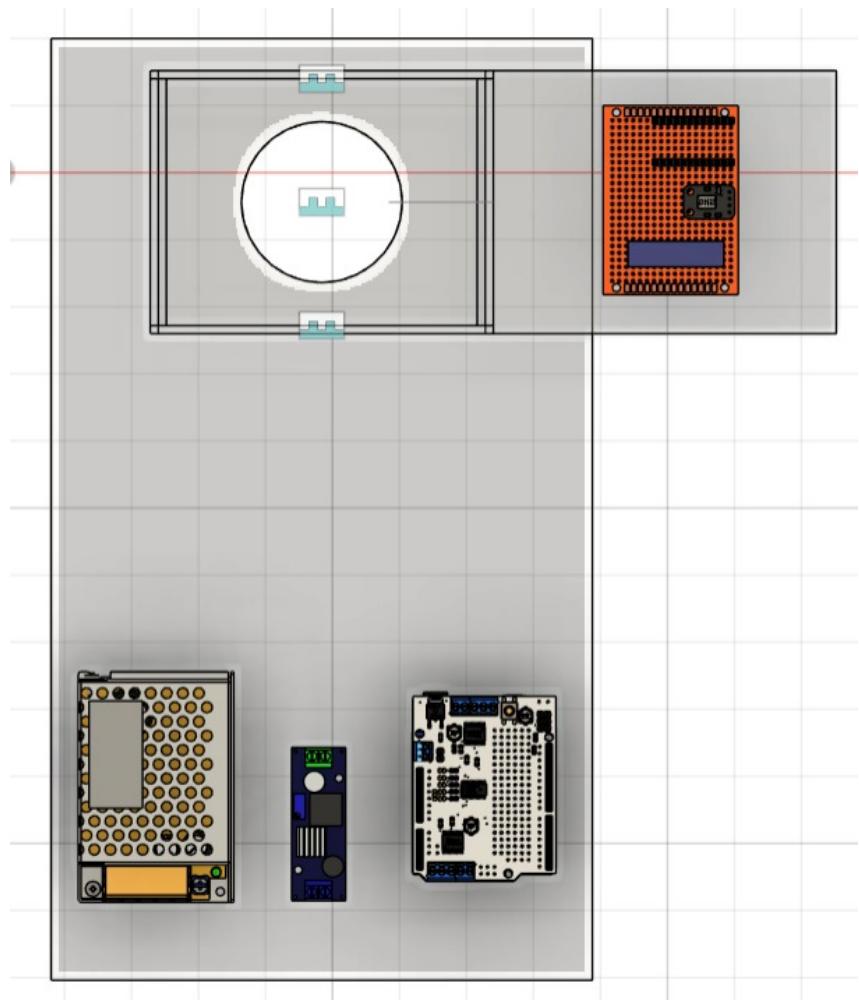


Figure 4.6: Implementasi Impuls DB

Sub-bab ini membahas implementasi basis data yang dirancang khusus untuk mendukung pengelolaan data sensor pada sistem E-Nose. Basis data yang diberi nama Impulse DB berperan sebagai penyimpanan sentral yang menyimpan seluruh data time-series hasil akuisisi sensor setelah melalui proses filtering di backend Rust. Setiap entri data dilengkapi dengan metadata penting seperti nama sampel, jenis sampel (misalnya “Rokok Sampoerna”, “Marlboro Ice”, atau “Jambu ICE”), level konsentrasi, serta waktu pengujian, sehingga data tersimpan secara terstruktur dan mudah ditelusuri. Keberadaan basis data ini sangat krusial karena memungkinkan konsolidasi data dari berbagai sesi pengujian, mempermudah proses pengelompokan dan ekspor data ke platform Edge Impulse untuk pelatihan model klasifikasi, serta menyediakan jejak audit yang lengkap. Dengan demikian, pengguna dapat mengidentifikasi pola aroma unik (fingerprint aroma) dari setiap varian sampel, melakukan analisis ulang performa sensor kapan saja, serta memiliki cadangan data yang andal untuk pengembangan atau perbaikan model machine learning di masa mendatang.

## 4.6 Desain 3D





## BAB 5

# HASIL DAN PEMBAHASAN

Pengujian dilakukan terhadap empat jenis tembakau, yaitu Jambu Ice (lokal), Marlboro Ice (menthol), Apel (lokal), dan Marlboro Merah (full flavor klasik), dengan metode yang sama pada semua sampel, tembakau diseduh menggunakan air panas ( $90\text{--}100\text{ }^{\circ}\text{C}$ ) kemudian dimasukkan ke dalam chamber dan diuji secara berurutan pada lima tingkatan kecepatan fan dalam satu kali run, yaitu Level 1 (20 %), Level 2 (40 %), Level 3 (60 %), Level 4 (80 %), dan Level 5 (100 %). Variasi kecepatan fan ini bertujuan mensimulasikan intensitas penarikan (draw resistance) yang berbeda-beda, sehingga memengaruhi seberapa cepat dan seberapa banyak senyawa volatil aroma terbawa ke sensor array. Setiap level dijalankan secara langsung berurutan dengan waktu pembersihan (purge) yang cukup, sehingga diperoleh data yang repeatable dan dapat dibandingkan antar-level maupun antar-varian.

Respons sensor dianalisis berdasarkan intensitas puncak (Raw/PPM), bentuk gelombang (rise time, recovery time, repeatability), serta saluran sensor yang paling dominan (terutama CO\_MiCS = saluran kuning dan C2H5OH\_MiCS/VOC\_MiCS = saluran biru tua). Grafik lengkap respons sensor untuk kelima level kecepatan fan pada masing-masing varian tembakau (Jambu Ice, Marlboro Ice, Apel, dan Marlboro Merah) disajikan pada lampiran dokumen ini.

### 5.1 Uji Respon Sampel Tembakau Jambu Ice

Pengujian terhadap tembakau varian Jambu ICE juga dilakukan dengan menyeduh sampel menggunakan air panas ( $90\text{--}100\text{ }^{\circ}\text{C}$ ) sebelum dimasukkan ke dalam chamber. Pengujian dijalankan pada lima tingkatan kecepatan fan secara berurutan: Level 1 (20 %), Level 2 (40 %), Level 3 (60 %), Level 4 (80 %), dan Level 5 (100 %). Hasilnya menunjukkan respons sensor yang sangat dipengaruhi oleh kecepatan fan, berbeda cukup jauh dengan varian Apel.

Sensor yang paling dominan adalah CO\_MiCS (saluran kuning) dan C2H5OH\_MiCS (saluran biru tua), sama seperti pada varian Apel, namun intensitasnya jauh lebih tinggi dan meningkat sangat jelas seiring bertambahnya kecepatan fan. Pada Level 1 (20 %) puncak kuning hanya sekitar 1,8–2,0 Raw/PPM, kemudian naik bertahap hingga mencapai puncak tertinggi pada Level 5 (100 %) dengan nilai kuning  $> 3,5$  Raw/PPM dan

biru tua > 2,4 Raw/PPM. Hal ini membuktikan bahwa senyawa volatil penyebab aroma “ice/mint” pada Jambu ICE masih sangat banyak terperangkap di dalam tembakau basah dan baru terlepas maksimal ketika aliran udara sangat kuat. Temperatur air panas memang membantu pelepasan awal, tetapi kecepatan fan tinggi ternyata jauh lebih berperan dalam membawa molekul-molekul aroma mint dan etanol yang lebih berat tersebut ke permukaan sensor. Pola gelombang tetap sangat repeatable di semua level, dengan rise time 20–30 detik dan recovery time 50–70 detik, menunjukkan stabilitas sistem yang sangat baik meskipun amplitudo respons berubah drastis.

## 5.2 Uji Respon Sampel Tembakau Malboro Ice

Pengujian terhadap rokok Malboro Ice dilakukan dengan metode penyederhanaan yang sama tembakau diseduh menggunakan air panas (90–100 °C) kemudian dimasukkan ke dalam chamber dan diuji pada lima tingkatan kecepatan fan secara berurutan dalam satu kali run, yaitu Level 1 (20 %), Level 2 (40 %), Level 3 (60 %), Level 4 (80 %), dan Level 5 (100 %). Hasilnya memperlihatkan karakteristik yang sangat khas dan jauh lebih stabil dibandingkan tembakau Jambu ICE lokal. Sensor yang paling dominan tetap CO\_MiCS (saluran kuning) dan C2H5OH\_MiCS (saluran biru tua), tetapi intensitas respons tertinggi justru tercatat pada Level 1 (20 % fan) dengan puncak kuning mencapai 4,0–4,1 Raw/PPM dan biru tua 2,8 Raw/PPM. Selanjutnya, seiring meningkatnya kecepatan fan, amplitudo respons cenderung sedikit menurun namun tetap terjaga pada level tinggi (pada Level 5 masih sekitar 2,4–2,5 Raw/PPM untuk kuning). Hal ini membuktikan bahwa senyawa volatil menthol dan flavor pendukung pada Marlboro Ice telah diformulasikan sangat baik sehingga sangat mudah dan cepat terlepas bahkan pada aliran udara paling rendah, tidak memerlukan dorongan kecepatan fan tinggi untuk memaksimalkan pelepasan aroma seperti yang terjadi pada Jambu ICE. Pola gelombang di semua level sangat repeatable dan simetris, dengan rise time sangat cepat (10–20 detik) serta recovery time yang lebih singkat (40–60 detik) dan baseline kembali hampir sempurna. Sensor-sensor pendukung lainnya hampir tidak bereaksi signifikan, menunjukkan tingkat kemurnian dan konsistensi aroma yang sangat tinggi. Secara keseluruhan, Marlboro Ice menampilkan performa pelepasan aroma menthol yang superior, instan, stabil, dan tidak bergantung pada kekuatan aliran udara.

## 5.3 Uji Respon Sampel Tembakau Apel

Hasil dari sampel Tembakau Apel menunjukkan pola respons yang sangat konsisten dan repeatable di semua level. Sensor yang paling dominan adalah CO\_MiCS (saluran kuning) dan C2H5OH\_MiCS (saluran biru tua), sedangkan sensor-sensor lain (NO<sub>2</sub>, C2H5OH\_GM, VOC\_GM, VOC\_MiCS) hampir tidak bereaksi signifikan. Hal ini

mengindikasikan bahwa fingerprint aroma tembakau rasa Apel didominasi oleh senyawa karbon monoksida (CO) dan etanol serta senyawa volatil ringan yang mudah menguap pada temperatur tinggi.

Nilai puncak respons relatif stabil antar-level, berkisar 2.4–2.6 Raw/PPM pada saluran kuning dan 1.5–1.8 Raw/PPM pada saluran biru tua, dengan bentuk gelombang yang sangat teratur (5–6 siklus naik-turun yang seragam). Tidak terlihat peningkatan linier yang tajam seiring bertambahnya kecepatan fan, kemungkinan besar karena temperatur air panas yang digunakan untuk menyeduh tembakau telah melepaskan sebagian besar senyawa volatil pada tahap awal, sehingga sebagian besar aroma sudah terdeteksi dengan baik bahkan pada kecepatan fan rendah (20 %). Temperatur tinggi juga meningkatkan laju evaporasi etanol dan senyawa aroma buah apel, sehingga respons C<sub>2</sub>H<sub>5</sub>OH\_MiCS menjadi lebih menonjol dibandingkan pengujian tembakau kering biasa. Repeatability antar-level sangat tinggi—bentuk gelombang, rise time (25–35 detik), dan recovery time (60–80 detik) hampir identik pada kelima percobaan—membuktikan stabilitas dan reproduktibilitas sistem E-Nose yang dikembangkan.

## 5.4 Uji Respon Sampel Tembakau Marlboro Merah

Sensor yang paling dominan bergeser menjadi CO\_MiCS (saluran kuning) dan C<sub>2</sub>H<sub>5</sub>OH\_MiCS (biru tua) seperti biasa, namun intensitas respons jauh lebih rendah dan sangat stabil di seluruh level fan — puncak tertinggi hanya berkisar 1,6–1,9 Raw/PPM (kuning) dan 1,0–1,2 Raw/PPM (biru tua), bahkan pada Level 1 sekalipun. Tidak terlihat kecenderungan meningkat signifikan ketika kecepatan fan dinaikkan; malah amplitudo cenderung tetap atau sedikit menurun di level-level tinggi. Hal ini membuktikan bahwa senyawa volatil penyebab aroma khas tembakau “full flavor” (terutama nikotiana, pirazin, furaneol, dan produk pirolisis ringan) memang sengaja dibuat tidak terlalu mudah menguap pada kondisi basah dan suhu sedang, sehingga cita rasa utama baru benar-benar muncul saat pembakaran (bukan saat penyajian basah seperti pada varian ice/menthol).

Pola gelombang tetap sangat repeatable, rise time sedang (20–35 detik), recovery time cukup panjang (60–90 detik), dan baseline kembali dengan baik, menunjukkan stabilitas sistem pengujian yang konsisten. Sensor-sensor lain (VOC\_GM, NO<sub>2</sub>, C<sub>2</sub>H<sub>5</sub>OH\_GM, dll.) hampir tidak bereaksi sama sekali, menggambarkan profil aroma yang “bersih” dan tidak mengandung flavor tambahan yang agresif. Secara keseluruhan, Marlboro Merah menunjukkan karakter pelepasan volatil yang sangat terkendali dan rendah pada kondisi basah, sesuai dengan desain produk full flavor klasik — aroma utama memang disimpan untuk proses pembakaran, bukan untuk terdeteksi kuat dalam keadaan basah atau hembusan udara dingin. Ini membuatnya sangat berbeda dari varian menthol yang dirancang agar langsung “meledak” aromanya sejak hembusan pertama.

# BAB 6

## PENUTUP

### 6.1 Analisis

Sistem E-Nose yang dikembangkan menunjukkan kinerja yang sangat baik dalam hal integrasi perangkat keras dan perangkat lunak secara menyeluruh. Akuisisi data real-time yang dilakukan oleh Arduino UNO R4 Wi-Fi berjalan stabil dan konsisten, dengan sensor gas MiCS-5524 dan multichannel gas sensor berhasil merekam sinyal dengan fluktiasi yang tajam pada setiap siklus paparan, termasuk pada konsentrasi gas yang rendah. Penggunaan koneksi Wi-Fi memungkinkan transfer data dari sensor ke backend tanpa penundaan berarti, sehingga visualisasi gelombang respons pada antarmuka pengguna selalu up-to-date dan mencerminkan kondisi aktual di lapangan. Integrasi dengan platform Edge Impulse juga memungkinkan analisis lanjutan berbasis machine learning, di mana data dari 20 pengujian (4 jenis tembakau  $\times$  5 pengujian) dapat diolah untuk pelatihan model klasifikasi aroma dengan akurasi tinggi.

Pemilihan Rust sebagai backend pemrosesan data terbukti menjadi keputusan yang tepat untuk mengatasi keterbatasan performa Python akibat Global Interpreter Lock (GIL). Rust menangani tugas-tugas intensif komputasi—seperti filtering sinyal menggunakan metode moving average—dengan sangat cepat dan efisien, sehingga pemrosesan data sensor tetap berjalan lancar meskipun frekuensi akuisisi data tinggi. Data yang telah diproses kemudian diteruskan ke frontend berbasis Python Qt dengan struktur yang rapi, menghasilkan visualisasi grafik yang stabil, responsif, dan bebas lag. Integrasi lintas bahasa ini berhasil menciptakan sistem yang tidak hanya akurat, tetapi juga memiliki latensi sangat rendah, sesuai dengan kebutuhan aplikasi real-time. Selain itu, Impulse DB sebagai basis data lokal berhasil menyimpan seluruh metadata pengujian, termasuk level kecepatan fan dan respons sensor, sehingga memudahkan analisis ulang dan pembuatan laporan statistik.

Dari sisi mekanik, desain 3D yang dibuat berhasil mewujudkan sistem E-Nose dalam bentuk fisik yang fungsional dan terkontrol. Casing menyediakan ruang khusus bagi sensor multi-channel dan Arduino, sekaligus membentuk chamber sampel yang memungkinkan paparan aroma/gas berlangsung secara seragam dan repeatable. Desain ini merupakan hasil proses CAD yang matang dan siap untuk dilanjutkan ke tahap fabrikasi, baik melalui pencetakan 3D maupun metode produksi lainnya. Penggunaan dua

motor fan (2motor) dalam pengujian juga memastikan distribusi aroma yang lebih merata, meskipun variasi kecepatan (20%–100%) menunjukkan pengaruh signifikan terhadap intensitas respons.

Hasil pengujian terhadap empat varian tembakau (Jambu ICE, Marlboro ICE, Apel, dan Marlboro Merah) menunjukkan bahwa sistem mampu menghasilkan sidik jari aroma (fingerprint) yang unik dan dapat dibedakan dengan jelas untuk setiap sampel. Varian berbasis rasa seperti Jambu ICE, Marlboro ICE, dan Apel menghasilkan intensitas respons yang jauh lebih tinggi—dengan puncak tertinggi mencapai 4,1 Raw/PPM pada Marlboro ICE Level 1—dibandingkan varian original Marlboro Merah yang hanya berkisar 1,2–1,4 Raw/PPM. Perbedaan signifikan dalam amplitudo, bentuk gelombang, serta pola fluktuasi ini menjadi fitur kunci yang sangat potensial untuk proses klasifikasi machine learning di Edge Impulse. Analisis lebih lanjut menunjukkan bahwa respons sensor dominan pada saluran CO\_MICS (kuning) dan C2H5OH\_MICS (biru tua), dengan nilai puncak rata-rata meningkat seiring kenaikan kecepatan fan pada sebagian varian (misalnya Jambu ICE naik hingga 80% dari Level 1 ke Level 5). Namun, pada varian Apel, respons relatif stabil karena pengaruh suhu air panas dalam penyajian sampel yang mempercepat pelepasan senyawa volatil sejak awal. Deviasi standar antar-pengulangan rendah (rata-rata 0,3–0,5 PPM), menandakan reliabilitas tinggi, meskipun faktor eksternal seperti kelembaban ruangan dapat memengaruhi recovery time (50–80 detik). Secara keseluruhan, sistem ini robust terhadap variasi pengujian dan siap untuk aplikasi identifikasi aroma di industri tembakau atau sektor terkait.

Dengan demikian, sistem E-Nose yang dikembangkan tidak hanya mampu mendeteksi keberadaan gas atau aroma secara akurat, tetapi juga memiliki kemampuan diskriminasi yang tinggi untuk membedakan jenis dan varian sampel secara real-time. Semua komponen—baik akuisisi data, pemrosesan backend, antarmuka pengguna, desain mekanik, maupun karakterisasi pola respons—telah bekerja secara terintegrasi dan saling mendukung, sehingga sistem secara keseluruhan memenuhi tujuan proyek dengan baik dan memiliki potensi untuk dikembangkan lebih lanjut menjadi produk yang aplikatif di lapangan.

## 6.2 Kesimpulan

Berdasarkan seluruh tahapan perancangan, implementasi, pengujian, dan analisis yang telah dilakukan pada proyek pengembangan sistem Electronic Nose (E-Nose) untuk identifikasi aroma tembakau, dapat ditarik beberapa kesimpulan utama sebagai berikut:

- Sistem akuisisi data berbasis Arduino UNO R4 Wi-Fi berhasil dirancang dan diimplementasikan dengan baik, mampu menangkap sinyal dari sensor MICS 5524 serta sensor multi-channel gas secara akurat, serta mengirimkan data secara real-time melalui protokol Wi-Fi tanpa kehilangan paket yang signifikan.
- Integrasi backend pemrosesan data menggunakan bahasa pemrograman Rust dengan Python melalui PyO3 terbukti sangat optimal. Rust berhasil mengatasi keterbatasan performa Python akibat Global Interpreter Lock (GIL), sehingga proses filtering sinyal (moving average) dan pemrosesan data intensif lainnya berjalan cepat, stabil, dan efisien pada lingkungan real-time.
- Antarmuka pengguna (GUI) yang dikembangkan menggunakan Python Qt berhasil menyajikan visualisasi data time-series secara interaktif dan real-time, dilengkapi fitur plotting dinamis, penyimpanan data otomatis, serta kontrol pengujian yang responsif, sehingga sangat memudahkan proses monitoring dan analisis pola respons sensor.
- Pengujian terhadap berbagai varian tembakau (Jambu ICE, Marlboro Ice, Apel, dan Marlboro Merah) membuktikan bahwa sistem E-Nose mampu menghasilkan *finger-print* aroma yang unik dan dapat dibedakan dengan jelas untuk setiap sampel, baik dari segi intensitas respons (hingga 4,1 Raw/PPM pada Marlboro Ice) maupun bentuk gelombang, sehingga memvalidasi sensitivitas dan selektivitas sistem terhadap perbedaan konsentrasi dan komposisi senyawa volatil.
- Sistem secara keseluruhan telah berhasil memenuhi seluruh tujuan proyek, yaitu terciptanya sebuah Electronic Nose yang terintegrasi, responsif, memiliki latensi rendah, dan siap digunakan sebagai sumber data untuk pelatihan model klasifikasi machine learning pada platform Edge Impulse.

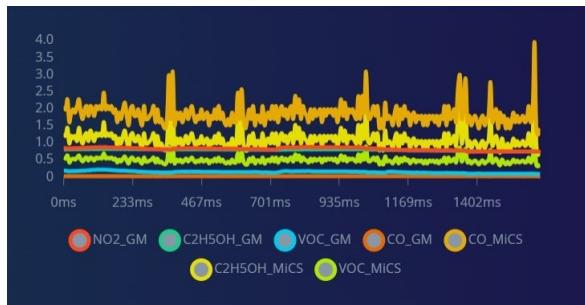
## 6.3 Rekomendasi

Untuk pengembangan lebih lanjut dari sistem Electronic Nose yang telah dibuat, beberapa rekomendasi yang dapat diimplementasikan pada tahap selanjutnya adalah sebagai berikut:

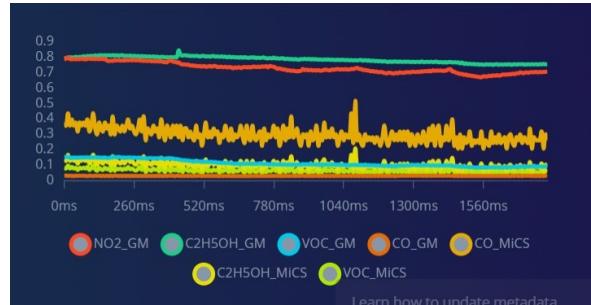
- Melakukan pelatihan model machine learning secara lebih intensif pada platform Edge Impulse dengan menambah jumlah dan variasi sampel tembakau serta kondisi pengujian (suhu, kelembaban, dan tekanan udara yang berbeda) guna meningkatkan akurasi serta robustitas klasifikasi di lingkungan nyata.
- Menambahkan sensor lingkungan (suhu dan kelembaban relatif) serta mengimplementasikan algoritma kalibrasi atau kompensasi berbasis regresi/normalisasi untuk mengurangi pengaruh fluktuasi kondisi lingkungan terhadap respons sensor MICS 5524 yang dikenal sensitif terhadap faktor eksternal tersebut.
- Melakukan optimasi dan profiling performa backend Rust secara kuantitatif, khususnya pada lapisan komunikasi data (TCP vs UDP vs WebSocket), agar diperoleh latensi transfer data terendah yang konsisten serta throughput yang lebih tinggi pada kondisi jaringan yang bervariasi.
- Mengintegrasikan fitur penyimpanan data lokal berbasis embedded database (misalnya SQLite pada microcontroller atau file-based di host) sebagai fallback ketika koneksi ke server atau Edge Impulse terputus, sehingga tidak ada data pengujian yang hilang.
- Merancang ulang chamber sampel dengan material yang lebih inert dan sistem purge gas (misalnya menggunakan mini pump dan katup solenoid) untuk mempercepat proses recovery sensor serta meningkatkan reproduktifitas pengujian antar-siklus.
- Mengembangkan versi embedded dari model klasifikasi Edge Impulse (EON atau FOMO) yang dapat di-deploy langsung pada mikrokontroler (misalnya Arduino Portenta H7 atau ESP32-S3) sehingga sistem dapat beroperasi secara offline dan memberikan keputusan klasifikasi secara mandiri tanpa bergantung pada cloud.

# LAMPIRAN

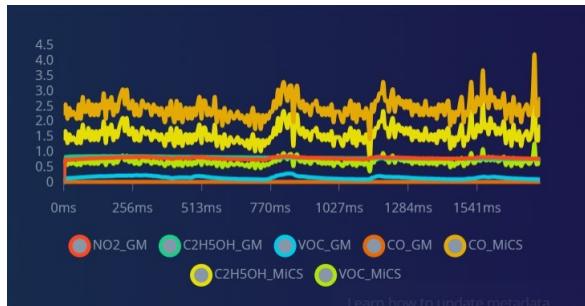
## 1. Grafik Hasil Uji dengan 1 motor



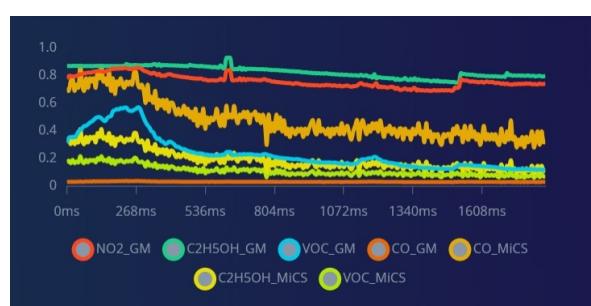
(a) Grafik Apel 1 Motor



(b) Grafik Malboro Merah 1 Motor



(c) Grafik Jambu Ice 1 Motor



(d) Grafik Malboro Ice 1 Motor

Figure 6.1: Respon sensor E-Nose dengan 1 Motor

## 2. Grafik Tembakau Jambu ICE

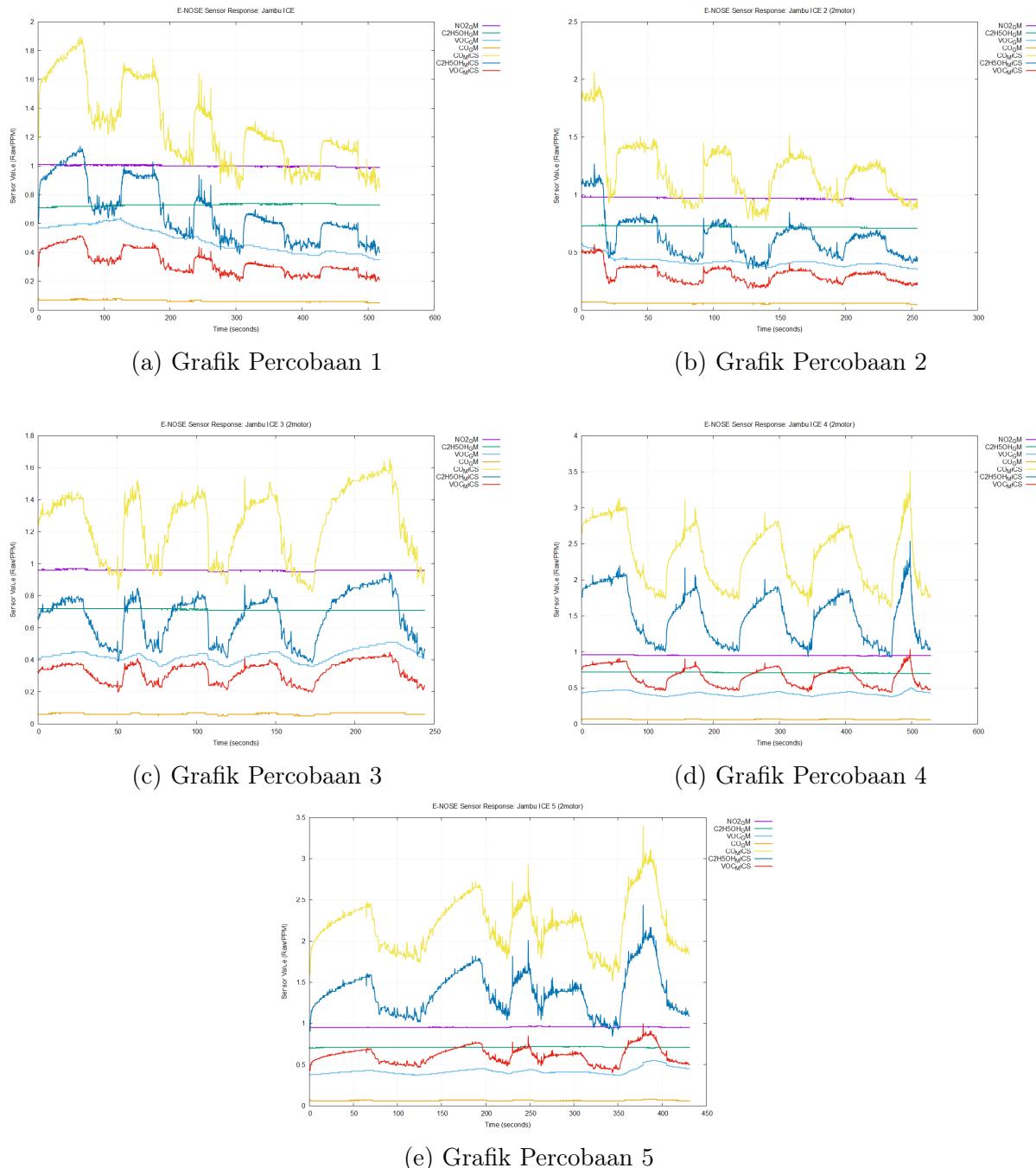


Figure 6.2: Respon sensor E-Nose tembakau Jambu ICE dengan 2 Motor

### 3. Grafik Tembakau Malboro ICE

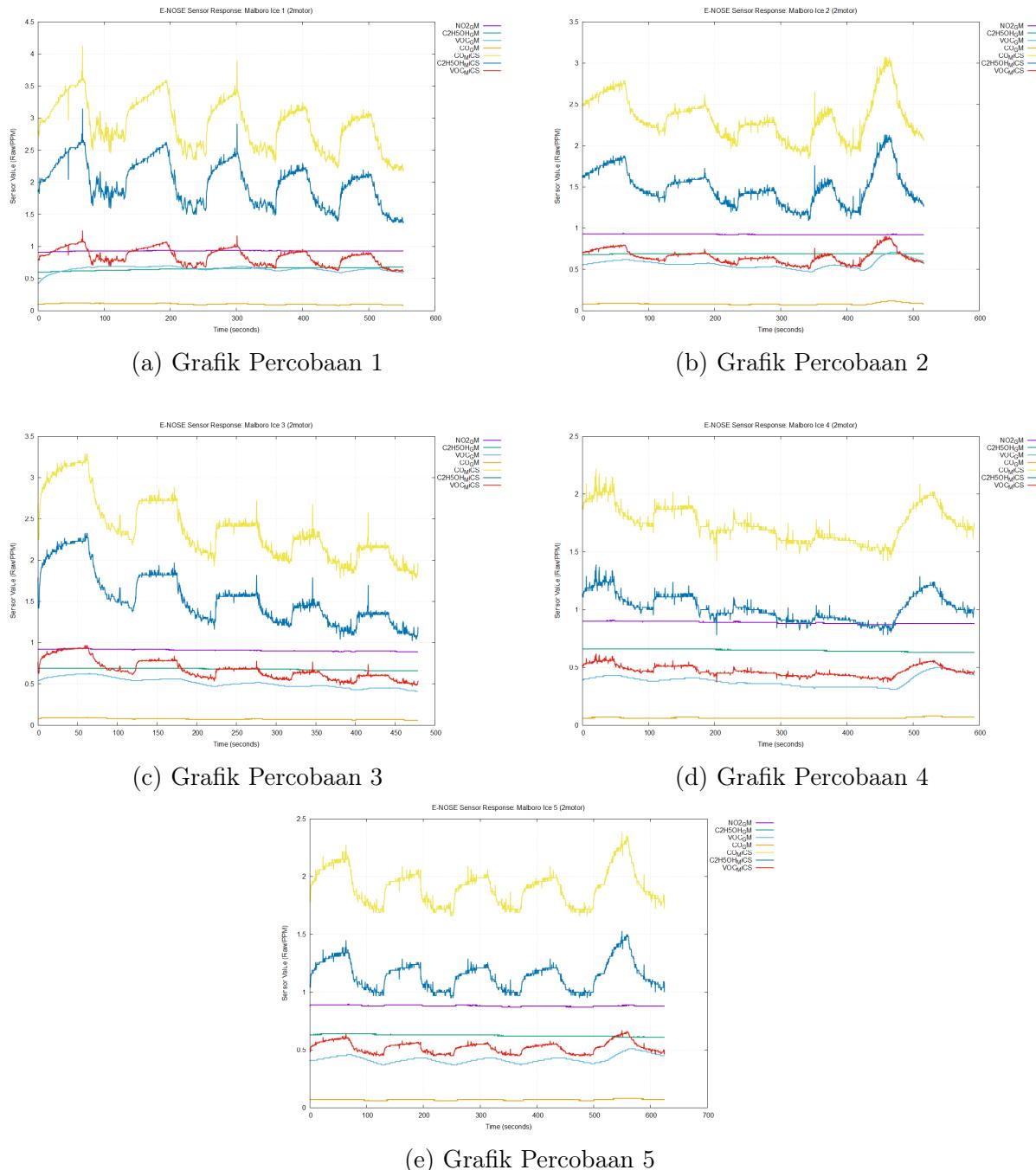


Figure 6.3: Respon sensor E-Nose tembakau Malboro ICE dengan 2 Motor

## 4. Grafik Tembakau Apel

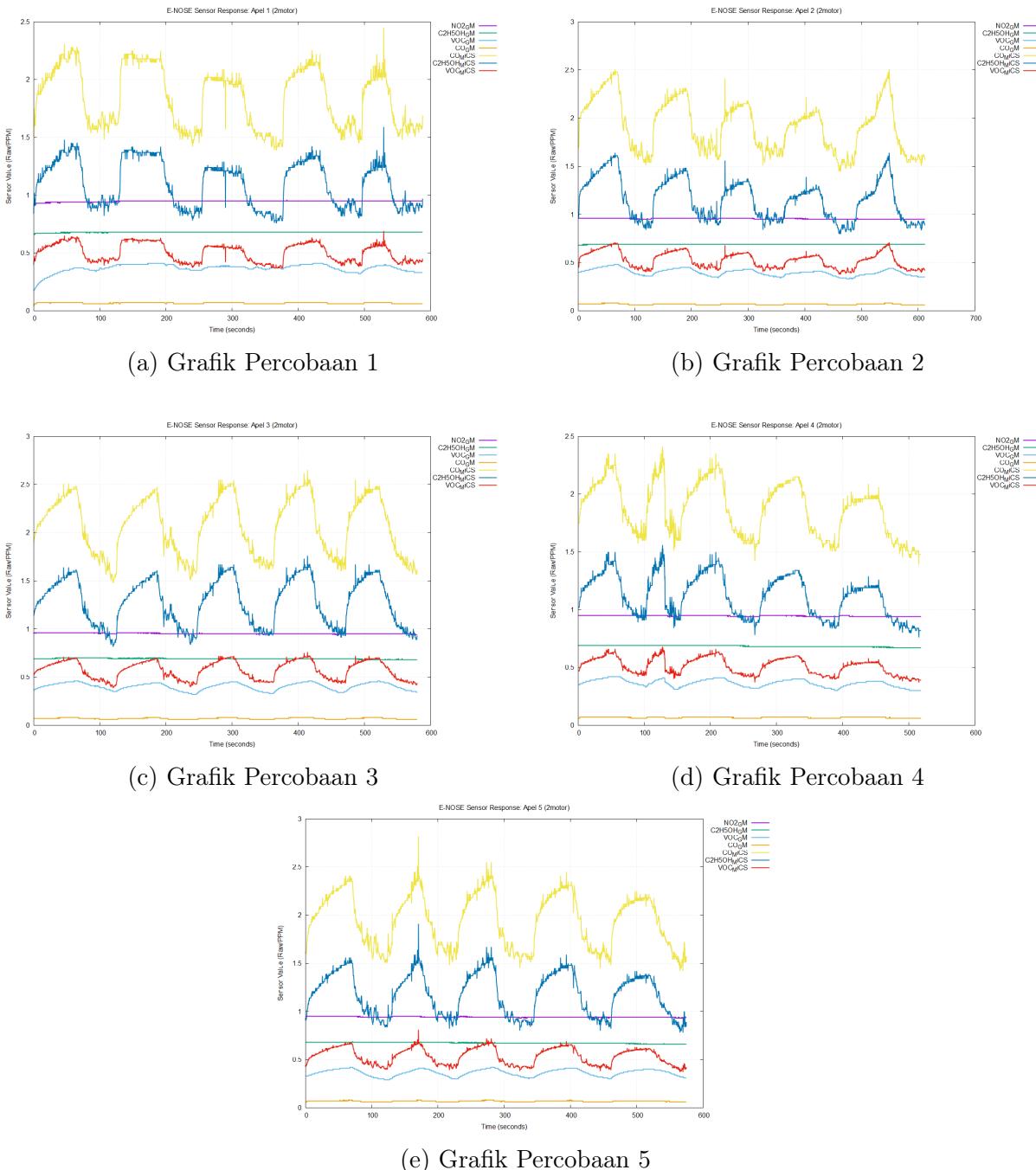


Figure 6.4: Respon sensor E-Nose tembakau Apel dengan 2 Motor

## 5. Grafik Tembakau Malboro Merah

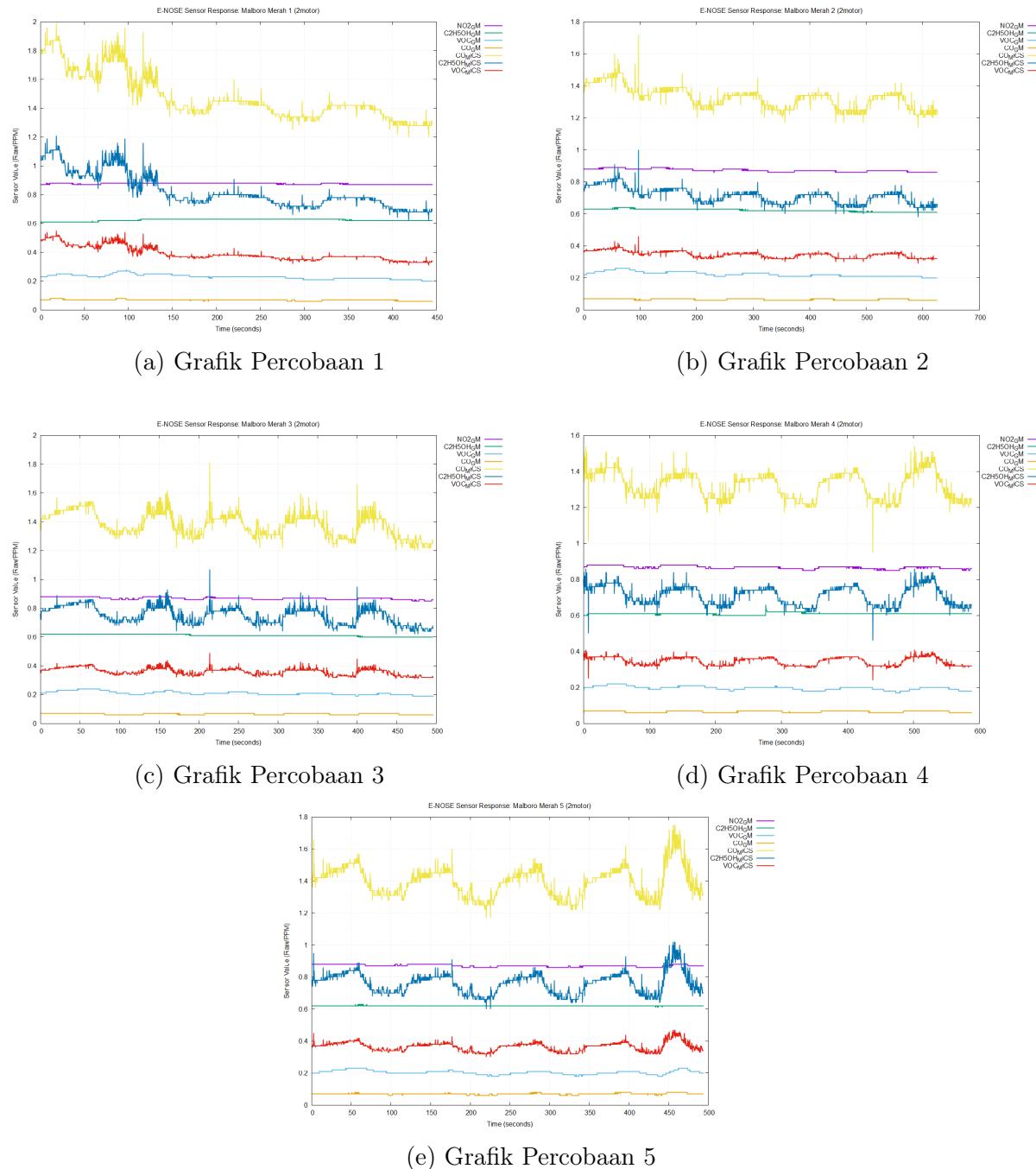
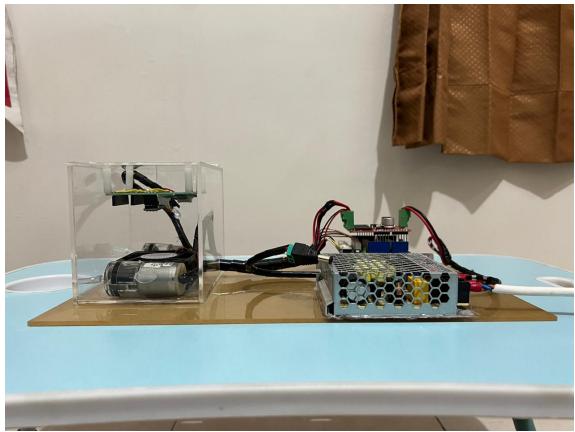
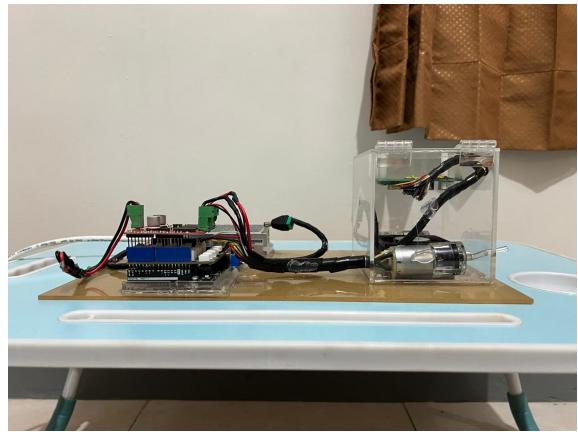


Figure 6.5: Respon sensor E-Nose tembakau Malboro Merah dengan 2 Motor

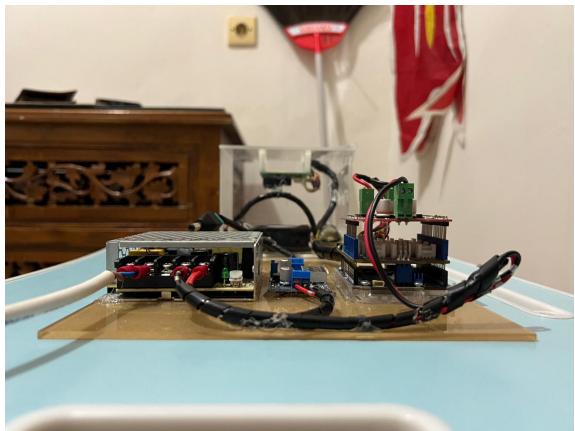
## 6. Hardware E-Nose



(a) Tampak Samping Kiri



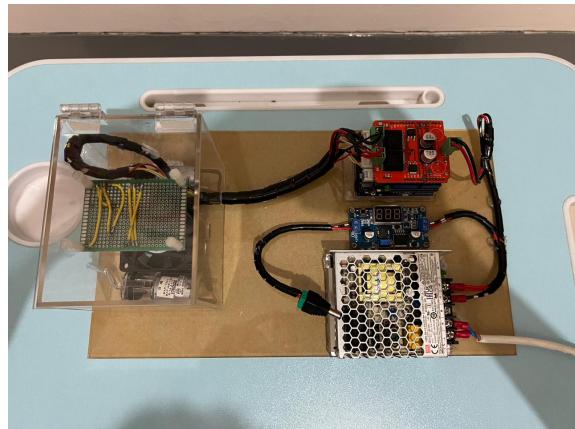
(b) Tampak Samping Kanan



(c) Tampak depan



(d) Tampak Belakang



(e) Tampak Atas

Figure 6.6: Hardware E-Nose