

Carbon Monoxide Monitoring System Prototype based on Laravel

Prototipe Sistem Monitoring Kadar Karbon Monoksida berbasis Laravel

Delly Yusar Akbar¹, Yoga Alif Kurnia²

^{1,2}Teknik Elektro, Fakultas Teknik, Universitas Widya Kartika
E-mail: *¹ yusardelly@gmail.com, ²yoga.alif@widyakartika.ac.id

Abstract – Indonesia, with its incredible natural wealth, is facing a serious threat in the form of air pollution. The increase in carbon monoxide (CO) levels, a toxic gas resulting from various human activities, is a major concern. Sources include transportation and industry, particularly from the combustion of fossil fuels. Forest fires also contribute to this issue. Carbon monoxide can harm human health, especially vulnerable groups like children and the elderly. Its impact extends to the environment, causing acid rain and damaging the ozone layer. Utilizing the MQ-7 sensor, Wemos D1 R2, Laravel, and MySQL, this study implements a website to display processed data from the MQ-7 sensor using Wemos D1 R2. The test results for MQ-7 sensor PPM values show an error percentage ranging from a minimum of 2.5% to a maximum of 47.5%. The design and results of the monitoring website function well with an internet connection.

Keywords — carbon monoxide, laravel, monitoring, MySQL, MQ-7 sensor

Abstrak – Indonesia dengan kekayaan alam yang luar biasa, menghadapi ancaman serius berupa pencemaran udara. Peningkatan kadar karbon monoksida (CO), gas beracun dari berbagai aktivitas manusia, menjadi perhatian utama. Sumbernya berasal dari transportasi dan industri, terutama dari pembakaran bahan bakar fosil. Kebakaran hutan juga turut berkontribusi. Karbon monoksida dapat merugikan kesehatan manusia dan lingkungan, menciptakan hujan asam dan merusak lapisan ozon. Menggunakan sensor MQ-7, Wemos D1 R2, Laravel dan MySQL. Pada penelitian ini menerapkan website untuk menampilkan hasil pengolahan data dari sensor MQ-7 menggunakan Wemos D1 R2. Hasil dari pengujian sensor MQ-7 nilai PPM memiliki prosentase kesalahan dari terkecil sebesar 2,5% dan terbesar dengan prosentase kesalahan 47,5% dan hasil pengukuran dari sensor MQ-7 dapat dilihat di website monitoring dengan koneksi internet.

Kata Kunci — karbon monoksida, laravel, monitoring, MySQL, sensor MQ-7

1. PENDAHULUAN

Indonesia sedang menghadapi tantangan serius dalam bentuk pencemaran udara. Salah satu ancaman tersembunyi yang semakin memprihatinkan adalah peningkatan kadar karbon monoksida (CO) di udara. Gas beracun ini, yang dihasilkan dari berbagai sumber aktivitas manusia, dapat memiliki dampak yang merugikan terhadap kesehatan manusia dan lingkungan [1].

Pencemaran udara karbon monoksida di Indonesia berasal dari berbagai aktivitas manusia, terutama dari sektor transportasi dan industri. Gas ini dilepaskan saat pembakaran bahan bakar fosil, seperti bensin dan minyak diesel, yang menjadi bahan bakar penggerak utama untuk kendaraan bermotor dan mesin industri. juga kebakaran hutan yang sering terjadi di beberapa wilayah Indonesia [2].

Karbon monoksida dapat merusak kesehatan manusia secara serius. Gas ini dapat menggantikan kadar oksigen di dalam darah, yang menyebabkan kurangnya kadar oksigen di seluruh tubuh. Paparan jangka panjang terhadap karbon monoksida dapat menyebabkan masalah pernapasan,

penyakit jantung, dan bahkan dapat berakibat fatal [3]. Kelompok rentan seperti anak-anak, lansia, dan individu dengan kondisi kesehatan yang sudah melemah lebih berisiko terkena dampak negatif ini.

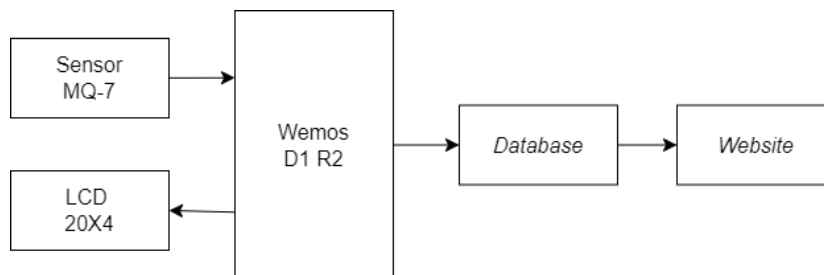
Pencemaran udara karbon monoksida juga memiliki dampak yang signifikan pada lingkungan. Gas ini dapat berkontribusi pada terbentuknya hujan asam, yang dapat merusak tanah dan sumber daya air. Selain itu, karbon monoksida juga dapat merusak lapisan ozon atmosfer, yang menyebabkan perubahan iklim dan pemanasan global.

Dalam rangka mengatasi permasalahan tersebut, penelitian mempunyai tujuan untuk merancang dan mengimplementasikan sistem pemantauan kadar karbon monoksida yang inovatif dan efisien, dengan memanfaatkan kerangka pengembangan aplikasi *web Laravel*. *Laravel* dipilih karena merupakan salah satu kerangka pengembangan PHP yang terkenal karena kecepatan, keamanan, dan kemudahan penggunaannya. Melalui pendekatan berbasis *Laravel*, diharapkan sistem monitoring ini tidak hanya dapat memberikan data yang akurat dan *real-time* terkait kualitas udara, tetapi juga menjadi solusi yang dapat diintegrasikan dengan mudah dalam lingkup pengembangan aplikasi *web* modern. Integrasi antara *Laravel* dan sensor CO akan memungkinkan pengumpulan data yang efisien dan manajemen informasi yang terstruktur.

2. METODE PENELITIAN

2.1. Perancangan Sistem Keseluruhan

Sistem yang akan dirancang yaitu sistem monitoring kadar karbon monoksida dengan sensor MQ-7, mikrokontroller *Wemos D1 R2*, LCD 20x4 dan *Laravel*. Sensor MQ-7 sebagai masukan untuk menghitung kadar karbon monoksida yang ada pada udara, mikrokontroller *Wemos D1 R2* digunakan untuk tempat mengolah data dari sensor MQ-7. LCD 20x4 untuk menampilkan data dari pengolah data sensor dan *Laravel* digunakan untuk menampilkan data dengan model *series*, *export data* dan menampilkan data dengan menggunakan *website*.

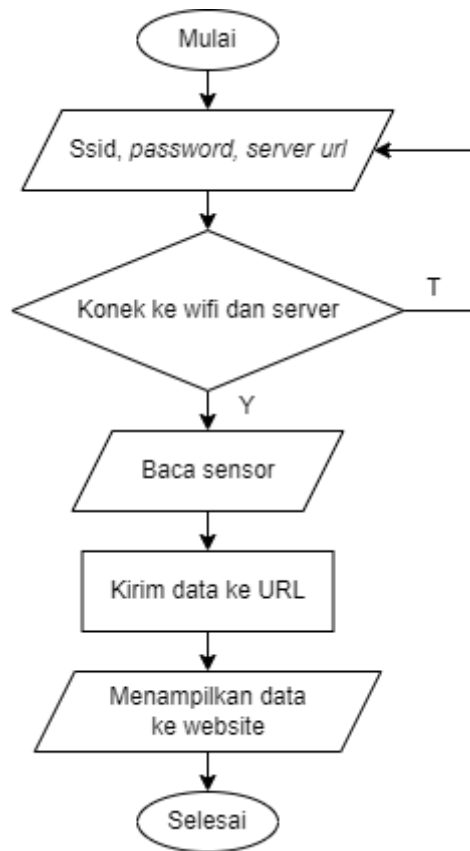


Gambar 1 Blok Diagram Sistem

Dari gambar 1 dapat diketahui bahwa penelitian menggunakan mikrokontroller *Wemos D1 R2* yang mendapatkan data kadar karbon monoksida dari sensor MQ-7 yang terhubung dengan pin A0 dengan tegangan kerja sebesar 5VDC [4], I²C LCD 20x4 digunakan untuk menampilkan data dari pengolah data *Wemos D1 R2* dan *Wemos D1 R2* mengirim data ke *database* menggunakan *Laravel* yang dapat diakses menggunakan *website*.

2.2. Perancangan Perangkat Lunak

Perancangan perangkat lunak membuat *flowchart* agar penerapan prosedur telah sesuai dan menganalisa kesalahan yang ada pada suatu system, berikut adalah tampilan *flowchart*-nya.

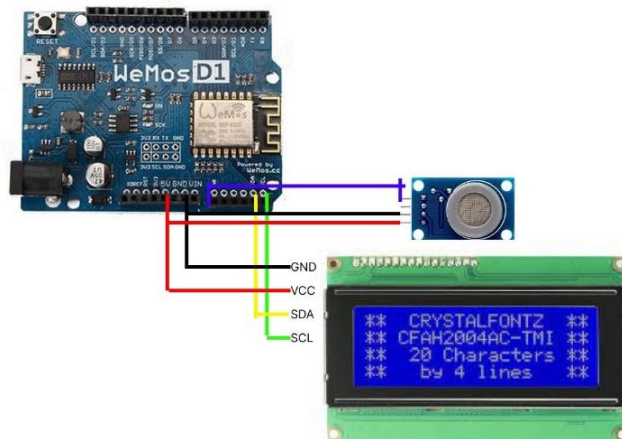


Gambar 2 Flowchart Penelitian

Gambar 2 menunjukkan perancangan perangkat lunak dimulai dengan menghubungkan *Wemos D1 R2* dengan *wifi* dan *server*. Pembacaan sensor MQ-7 akan dimulai Ketika *Wemos D1 R2* terhubung dengan koneksi internet ke *server* dan data akan dikirim dan disimpan di *database* menggunakan *framework PHP Laravel*. *Website* dapat diakses melalui koneksi internet.

2.3. Perancangan Perangkat Keras

Perancangan perangkat keras dari prototipe sistem monitoring kadar karbon monoksida berbasis *Laravel* ini menggunakan sensor MQ-7 berikut ini:



Gambar 3 Perancangan Perangkat Keras

Dari gambar 3. dapat diketahui penelitian ini menggunakan Wemos D1 R2, Sensor MQ-7 koneksi dengan pin A0 dan LCD 20x4 menggunakan I²C yang koneksi dengan pin SDA, SCL dengan tegangan kerja sebesar 5VDC.

2.4. Perancangan Perangkat Keras

Sensor MQ-7 adalah sensor yang bisa mengukur kadar gas karbon monoksida yang ada pada udara yang tersusun dari AL₂O₃, SnO₂, elektroda pengukur dan lapisan dari plastik sebagai pemanas dan permukaan dengan jaring. Sensor ini bekerja pada tegangan 5VDC dengan keluaran sinyal analog [5].

Rumus untuk mengonversi nilai ADC ke PPM adalah menggunakan persamaan 1 – 3 sebagai berikut:

$$VRL = \text{Nilai Sensor} \cdot \frac{VCC}{1024} \dots\dots\dots (1)$$

$$Rs = \left(VCC \cdot \frac{RL}{VRL} \right) - RL \dots\dots\dots (2)$$

$$100PPM = \left(\frac{RS}{R0} \right) = 1 \dots\dots\dots (3)$$

Keterangan :

- VRL = keluaran tegangan rangkaian
- VCC = nilai tegangan sensor MQ-7
- Rs = nilai tahanan sensor MQ-7
- RL = nilai tahanan beban rangkaian
- Ro = nilai tahanan kalibrasi sensor
- PPM = nilai kadar karbon monoksida



Gambar 4 Sensor MQ-7

2.5. Laravel

Laravel adalah *framework PHP* populer dan *open-source*. Dengan fokus pada sintaks ekspresif dan produktivitas tinggi, *Laravel* menyediakan berbagai fitur kuat untuk mempermudah pembuatan aplikasi *web* [6]. *Framework* ini mengusung konsep *Model-View-Controller* (MVC) dengan pemisahan logika bisnis, presentasi, dan manajemen data.

Salah satu fitur kunci *Laravel* adalah *Eloquent ORM*, yang memungkinkan pengembang berinteraksi dengan basis data menggunakan sintaks yang ekspresif dan intuitif. *Laravel* juga menyediakan *Blade*, sistem *templating* yang memungkinkan pengembang membuat tampilan *web* dengan mudah dan efisien [7].

Dalam hal keamanan, *Laravel* menonjolkan penggunaan *Eloquent ORM* untuk melindungi aplikasi dari *SQL injection attack* dan *Cross-Site Scripting* (XSS). *Laravel* juga menyertakan sistem

manajemen otentikasi yang dapat diimplementasikan dengan cepat untuk melindungi area terbatas aplikasi.

2.6. MySQL

MySQL adalah (RDBMS) *open-source* yang banyak digunakan saat ini. Dikembangkan oleh Oracle Corporation, MySQL menyediakan lingkungan yang handal dan efisien untuk menyimpan, mengelola, dan mengakses data [8]. MySQL menggunakan bahasa *query SQL (Structured Query Language)* untuk berinteraksi dengan basis data.

Salah satu keunggulan MySQL adalah performa tinggi dan skalabilitasnya. Sistem ini dapat menangani sejumlah besar data dan kueri dengan cepat, membuatnya cocok untuk aplikasi-aplikasi dengan volume tinggi. MySQL juga mendukung replikasi, memungkinkan replikasi data antara *server*, yang berguna untuk meningkatkan ketersediaan dan ketahanan [9].

2.7. Indeks Standar Polusi Udara

Indeks Standar Polusi Udara (ISPU) adalah parameter yang digunakan untuk mengukur tingkat pencemaran udara di suatu wilayah atau lokasi tertentu. ISPU memberikan gambaran tentang seberapa baik kualitas udara di suatu area dengan merinci tingkat polutan yang ada. ISPU biasanya mencakup beberapa parameter utama, seperti PM10 dan PM2.5, O₃, NO₂, SO₂, dan CO. Data yang dihasilkan dari pengukuran ini sering kali diwakili dalam bentuk indeks numerik atau kategoris yang memberikan informasi tentang tingkat risiko kesehatan Masyarakat [10].

3. HASIL DAN PEMBAHASAN

3.1. Pengujian Sensor MQ-7

Pengujian sensor MQ-7 dilakukan untuk mengetahui keakuratan sensor MQ-7 untuk menghitung kadar karbon monoksida dalam ppm di udara yang dibandingkan dengan *carbon monoxide meter* dengan cara menghembuskan gas di depan sensor MQ07 secara bertahap. Pengujian ini dilakukan sebanyak sepuluh kali. Berikut tabel 1 adalah hasil pengujian sensor MQ-7

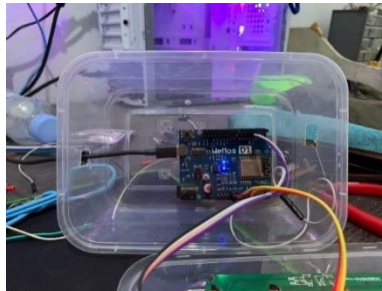
Tabel 1. Hasil Pengujian Sensor MQ-7

Pengujian Ke	MQ-7 (ppm)	CO Meter (ppm)	Kesalahan (%)
1	12,3	12	2,5
2	1,9	1	47,4
3	32,1	31	3,4
4	15,8	15	5,1
5	12,0	11	8,3
6	1,2	1	16,7
7	6,5	6	7,7
8	5,2	5	3,8
9	6,4	6	6,3
10	7,2	7	2,8
Kesalahan Rata-Rata			10,4

Berdasarkan tabel 1 pengujian sensor MQ-7 memiliki persentase kesalahan dari terkecil sebesar 2,5% dan terbesar dengan persentase kesalahan 47,5% dengan rata-rata kesalahan sebesar 10,4%. Persentase kesalahan didapatkan dari perbedaan pembacaan data gas dari sensor MQ-7 dengan *carbon monoxide meter*.



Gambar 5 Tampilan Luar Pengukuran Gas



Gambar 6 Tampilan Dalam Pengukuran Gas

```

#include <LiquidCrystal_I2C.h>
#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>
LiquidCrystal_I2C lcd(0x27,D1,D2);
const char* ssid = "ssid";
const char* password = "password";
const char* serverUrl = "http://url/api/store-gas-data";
const int gasSensorPin = A0;
void setup() {
  Serial.begin(115200);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.println("Connecting to WiFi...");
  }
  Serial.println("Connected to WiFi");
  lcd.init();
  lcd.init();
  lcd.backlight();
}
void loop() {
  float gasValue = analogRead(gasSensorPin);
  float VRL = gasValue*5.00/1024;
  float rs = ( 5.00 * 1000 / VRL ) - 1000;
  float ppm = 100 * pow(rs / 873,-1.53);
  WiFiClient client;
  HTTPClient http;
  http.begin(client, serverUrl);
  http.addHeader("Content-Type", "application/x-www-form-urlencoded");
  String postData = "gas_level=" + String(ppm);
  int httpCode = http.POST(postData);
  if (httpCode > 0) {
    String response = http.getString();
    Serial.println("HTTP response code: " + String(httpCode));
    Serial.println("Response: " + response);
  } else {
    Serial.println("HTTP POST request failed");
  }
  http.end();
  lcd.clear();
  lcd.setCursor(3,0);
  lcd.print(ppm);
  lcd.setCursor(9,0);
  lcd.print("ppm");
  delay(5000);
}

```

Gambar 7 Kode Sumber untuk MQ-7

3.2. Pengujian Database

Pengujian *database* digunakan untuk mengetahui apakah data dari ESP-32 sudah tersimpan dengan benar sesuai dengan data yang dikirim. Sebelum menguji *database* buat terlebih dahulu model di *Laravel* dengan *command: php artisan make:model GasModel* lalu buat program seperti pada gambar 8.

```
<?php
namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class GasData extends Model
{
    use HasFactory;
    protected $fillable = ['gas_level', 'timestamp'];
}
```

Gambar 8 Kode Sumber Model Database MQ-7

Hasil dari Model pada gambar 8 dapat dilihat pada MySQL yang ditunjukkan oleh Gambar 9 dengan nama tabel **gas_data** dengan mengakses C-Panel pada *web*.



Gambar 9 Hasil Model

Setelah membuat model lalu buat *controller* yang digunakan untuk memproses semua permintaan http dari pengguna dan mengembalikan respon sesuai dengan permintaan pengguna dengan *command: php artisan make:controller GasData*.

```
<?php
namespace App\Http\Controllers;

use App\Models\GasData;
use Illuminate\Http\Request;

class GasDataController extends Controller
{
    public function store(Request $request)
    {
        $data = $request->validate([
            'gas_level' => 'required|numeric',
        ]);
        GasData::create($data);
        return response()->json(['message' => 'Data stored successfully'], 200);
    }

    public function showGasLevel()
    {
        $gasLevels = GasData::latest('created_at')->get();
        $labels = $gasLevels->pluck('created_at')->map(function ($item) {
            return $item->format('Y-m-d');
        });
        $values = $gasLevels->pluck('gas_level');
        $gasLevelData = GasData::latest()->first();
    }
}
```

Gambar 10 Kode Controller Gas Data

Setelah membuat *controller* lalu tambahkan *route* di *api.php* agar pengguna dapat mengirim data dari ESP-32 dan mengakses data untuk ditampilkan di *browser*.

```
<?php
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Route;

Route::middleware('auth:sanctum')->get('/user', function (Request $request) {
    return $request->user();
});
Route::post('/store-gas-data', '\App\Http\Controllers\GasDataController@store');
Route::post('/mon-data', '\App\Http\Controllers\MondataController@store');
```

Gambar 11 Kode Sumber *Route*

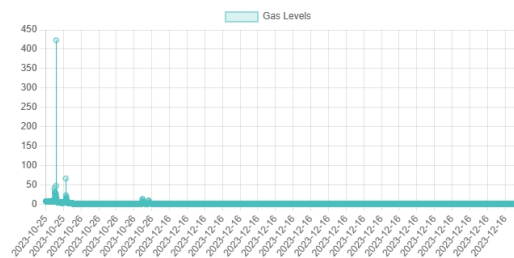
3.3. Pengujian Website

Pengujian *website* monitoring berjalan di *live server* dengan *url: url/mon-data* sesuai dengan yang telah direncanakan, hasil pengujian tampilan *website* dapat dilihat pada gambar 12. Untuk dapat mengakses *website* harus menggunakan koneksi internet.

Sistem Monitoring Gas Karbon Monoksida

Level Gas :

1.22 ppm



Gambar 12 Tampilan *Website Monitoring*

4. KESIMPULAN

Dari hasil pengujian alat dapat diambil kesimpulan bahwa berdasarkan data hasil pengujian sensor MQ-7 nilai PPM memiliki prosentase kesalahan dari terkecil sebesar 2,5% dan terbesar dengan prosentase kesalahan 47,5%. Perancangan dan hasil dari *website monitoring* dapat berjalan dengan baik dengan koneksi internet.

DAFTAR PUSTAKA

- [1] I. M. Zaqi and N. Y. Sudiar, "CARBON MONOXIDE MEASURING SYSTEM USING MQ-7 SENSOR WITH CLOUD STORAGE," vol. 16, no. 2, pp. 89–98, 2023.
- [2] M. Mujirudin, N. F. Noviandy, and H. Ramza, "Pengujian Respirator KN95 Menggunakan Sensor MQ-7 Berbasis Mikrokontroler Arduino UNO," *Pros. Semin. Nas. Teknoka*, vol. 6, no. 2502, pp. 223–227, 2022, doi: 10.22236/teknoka.v6i1.469.
- [3] M. Zidni, M. Hannats, H. Ichsan, and S. R. Akbar, "Sistem Monitoring Kesehatan Udara menggunakan Sensor MQ7 dan MQ135 terhadap Berbagai Gas Berbahaya pada Mobil," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 6, no. 9, pp. 4322–4328, 2022.
- [4] A. A. Rosa, B. A. Simon, and K. S. Lieanto, "Sistem Pendeteksi Pencemaran Udara Portabel

-
- Menggunakan Sensor MQ-7 dan MQ-135,” *Ultim. Comput. J. Sist. Komput.*, vol. 12, no. 1, pp. 23–28, 2020, doi: 10.31937/sk.v12i1.1611.
- [5] M. A. Sebayang, “Journal of Informatics and Telecommunication Engineering Stasiun Pemantau Kualitas Udara Berbasis Web Web Based Quality Air Monitor Station melakukan perancangan alat , perancangan melakukan MQ-7 Sensor MQ-7 tersusun oleh tabung,” *J. Informatics Telecommun. Eng.*, vol. 1, no. 1, pp. 24–33, 2017, [Online]. Available: <http://ojs.uma.ac.id/index.php/jite>
- [6] Desma Aipina and Harry Witriyono, “Pemanfaatan Framework Laravel Dan Framework Bootstrap Pada Pembangunan Aplikasi Penjualan Hijab Berbasis Web,” *J. Media Infotama*, vol. 18, no. 1, pp. 36–42, 2022.
- [7] N. H. Oktavanusa and R. Kurniawan, “Pemanfaatan Laravel Eloquent ORM Pada Fitur Kelola Transaksi Dalam Aplikasi Balelabs Billing,” *Jurnal Portal - Universitas Islam Indonesia*, vol. 3, no. 1, pp. 3–13, 2022.
- [8] D. D. Jantce TJ Sitinjak, . Maman, and J. Suwita, “Analisa Dan Perancangan Sistem Informasi Administrasi Kursus Bahasa Inggris Pada Intensive English Course Di Ciledug Tangerang,” *Insa. Pembang. Sist. Inf. dan Komput.*, vol. 8, no. 1, 2020, doi: 10.58217/ipsikom.v8i1.164.
- [9] I. Sholihah and C. Darujati, “Sistem Replikasi Basis Data berdasarkan MySQL menggunakan Container Docker,” *Maj. Ilm. Teknol. Elektro*, vol. 21, no. 2, p. 209, 2022, doi: 10.24843/mite.2022.v21i02.p08.
- [10] R. Arissa and A. A. Kiswandono, “Kajian Indeks Standar Polusi Udara (ISPU) Pm10, So2, O3, Dan No2 Di Kota Bandar Lampung,” *Anal. Environ. Chem.*, vol. 2, no. 2, pp. 38–46, 2017.

