

# **LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)**

**Fakultas Vokasi, Universitas Brawijaya**

**“LAPORAN PRAKTIK PEMBUATAN API MENGGUNAKAN  
LARAVEL 11 DAN NGROK”**



**Achmad Fawaz Ramdhani**

**233140700111089**

**Fakultas Vokasi, Universitas Brawijaya**

**Email : fawaz333888@gmail.com**

## Abstrak

Eksperimen ini bertujuan untuk membuat dan menguji API menggunakan Laravel 11 serta menghubungkannya dengan perangkat IoT melalui layanan Ngrok. API yang dibuat berfungsi untuk menyimpan, mengambil, memperbarui, dan menghapus data sensor yang tersimpan di database MySQL. Dengan menggunakan Postman dan simulator Wokwi, eksperimen ini membuktikan bahwa API dapat berfungsi dengan baik dan dapat diakses melalui internet.

## 1. Pendahuluan

### 1.1. Latar Belakang

Dalam pengembangan sistem IoT, API berperan sebagai jembatan komunikasi antara perangkat keras dan aplikasi. Dengan menggunakan Laravel 11, API dapat dibangun dengan cepat dan mudah. Ngrok digunakan untuk mengonlinekan API agar dapat diakses oleh perangkat IoT.

### 1.2. Tujuan

Eksperimen ini bertujuan untuk membangun API menggunakan Laravel 11, menghubungkannya dengan database MySQL, dan menguji fungsionalitasnya melalui Postman serta Wokwi IoT Simulator.

## 2. Metodologi

### 2.1. Tools & Materials (Alat dan Bahan)

- Laravel 11
- MySQL (phpMyAdmin)
- Postman
- Ngrok
- Wokwi ESP32 Simulator

### 2.2. Implementation Steps (Langkah Implementasi)

1. Membuat database di phpMyAdmin dengan nama `iot_25`.
2. Membuat model `TransaksiSensor.php` dengan perintah **`php artisan make:model TransaksiSensor`**
3. Membuat dan menyesuaikan migration untuk tabel `transaksi_sensor`.
4. Menjalankan migrasi dengan perintah **`php artisan migrate`**
5. Membuat resource API dengan perintah **`php artisan make:resource TransaksiSensorResource`**
6. Membuat API Controller `TransaksiSensorController.php`.
7. Menyesuaikan routing API dalam `routes/api.php`.
8. Menjalankan Laravel menggunakan **`php artisan serve`**
9. Menguji API menggunakan Postman dengan metode GET dan POST.
10. Menggunakan Ngrok untuk mengonlinekan API.

### 3. Results and Discussion (Hasil dan Pembahasan)

#### 3.1. Experimental Results (Hasil Eksperimen)

Dari hasil pengujian menggunakan Postman, API dapat melakukan operasi CRUD terhadap tabel transaksi\_sensor di database iot\_25. Selain itu, pengujian dengan Ngrok menunjukkan bahwa API dapat diakses secara online dan siap untuk diakses dari perangkat eksternal seperti simulator Wokwi.

### 4. Appendix (Lampiran)

```
C:\Users\ASUS\Downloads\Compressed\ngrok-v3-stable-windows-amd64\ngrok.exe - ngrok http http://localhost:8000

ngrok

[Q] Goodbye tunnels, hello Agent Endpoints: https://ngrok.com/r/aep

Session Status      online
Account             Galbert Einstein (Plan: Free)
Version             3.20.0
Region              Asia Pacific (ap)
Latency             38ms
Web Interface       http://127.0.0.1:4040
Forwarding           https://510f-180-248-17-49.ngrok-free.app -> http://localhost:8000

Connections          ttl    opn    rt1    rt5    p50    p90
                    2      0      0.00   0.00   0.88   0.97

HTTP Requests
-----
16:52:07.535 +07 POST /api/posts          201 Created
16:51:17.114 +07 GET  /api/posts          200 OK
```

HTTP <https://510f-180-248-17-49.ngrok-free.app/api/posts> Save Share

POST <https://510f-180-248-17-49.ngrok-free.app/api/posts> Send

Params Authorization Headers (10) **Body** Scripts Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON Beautify

```
1 {
2   "nama_sensor": "Sensor D",
3   "nilai1": 123,
4   "nilai2": 321
5 }
6
```

Body Cookies Headers (8) Test Results 201 Created • 886 ms • 354 B

Pretty Raw Preview Visualize JSON

```
1 {
2   "data": {
3     "id": 4,
4     "nama_sensor": "Sensor D",
5     "nilai1": 123,
6     "nilai2": 321
7   }
8 }
```

HTTP <https://510f-180-248-17-49.ngrok-free.app/api/posts> Save Share

GET <https://510f-180-248-17-49.ngrok-free.app/api/posts> Send

Params Authorization Headers (6) **Body** Scripts Settings Cookies

☒ none ☐ form-data ☐ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

This request does not have a body

Body Cookies Headers (8) Test Results 200 OK • 1266 ms • 1004 B

Pretty Raw Preview Visualize JSON

```
1 {
2   "data": [
3     {
4       "id": 2,
5       "nama_sensor": "Sensor B",
6       "nilai1": 87,
7       "nilai2": 176
8     },
9     {
10      "id": 1,
11      "nama_sensor": "Sensor A",
12      "nilai1": 100,
13      "nilai2": 200
14    },
15    {
16      "id": 3,
17      "nama_sensor": "Sensor C",
18      "nilai1": 111,
19      "nilai2": 211
20    }
21  ],
22 }
```

Server: 127.0.0.1 » Database: iot\_25 » Table: transaksi\_sensor

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Privileges](#)
[Operations](#)

✓ Showing rows 0 - 3 (4 total, Query took 0.0359 seconds.)

`SELECT * FROM `transaksi_sensor``

☐ Profiling
 [\[ Edit inline \]](#)
[\[ Edit \]](#)
[\[ Explain SQL \]](#)
[\[ Create PHP code \]](#)
[\[ Refresh \]](#)

☐ Show all
 Number of rows: 
 Filter rows: 
 Sort by key:

Extra options

	id	nama_sensor	nilai1	nilai2	created_at	updated_at
<input type="checkbox"/> <a href="#">Edit</a> <a href="#">Copy</a> <a href="#">Delete</a>	1	Sensor A	100	200	2025-03-07 16:29:54	2025-03-07 16:29:54
<input type="checkbox"/> <a href="#">Edit</a> <a href="#">Copy</a> <a href="#">Delete</a>	2	Sensor B	87	176	2025-03-07 16:30:41	2025-03-07 16:30:41
<input type="checkbox"/> <a href="#">Edit</a> <a href="#">Copy</a> <a href="#">Delete</a>	3	Sensor C	111	211	2025-03-07 09:38:18	2025-03-07 09:38:18
<input type="checkbox"/> <a href="#">Edit</a> <a href="#">Copy</a> <a href="#">Delete</a>	4	Sensor D	123	321	2025-03-07 09:52:08	2025-03-07 09:52:08

☐ Check all
 With selected: [Edit](#) [Copy](#) [Delete](#) [Export](#)

```

1  #include <Arduino.h>
2
3  const int trigPin = 5;
4  const int echoPin = 18;
5
6
7  //define sound speed in cm/uS
8  #define SOUND_SPEED 0.034
9  #define CM_TO_INCH 0.393701
10
11
12 long duration;
13 float distanceCm;
14 float distanceInch;
15
16
17 void setup() {
18   Serial.begin(115200); // Starts the serial communication
19   pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
20   pinMode(echoPin, INPUT); // Sets the echoPin as an Input
21 }
22
23
24 void loop() {
25   // Clears the trigPin
26   digitalWrite(trigPin, LOW);
27   delayMicroseconds(2);
28   // Sets the trigPin on HIGH state for 10 micro seconds
29   digitalWrite(trigPin, HIGH);
30   delayMicroseconds(10);
31   digitalWrite(trigPin, LOW);
32   // Reads the echoPin, returns the sound wave travel time in microseconds
33   duration = pulseIn(echoPin, HIGH);
34   // Calculate the distance
35   distanceCm = duration * SOUND_SPEED/2;
36   // Convert to inches
37   distanceInch = distanceCm * CM_TO_INCH;
38   // Prints the distance in the Serial Monitor
39   Serial.print("Distance (cm): ");
40   Serial.println(distanceCm);
41   // Serial.print("Distance (inch): ");
42   // Serial.println(distanceInch);
43   delay(1000);
44 }
45

```