**LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)** Fakultas Vokasi, Universitas Brawijaya

**Praktik Simulasi Sensor Jarak (Ultrasonic)**



*William jan randall ginting*  
 Fakultas Vokasi, Universitas Brawijaya  
 Email: janrandall.ginting@gmail.com

## **Abstract (Abstrak)**

Praktikum ini bertujuan untuk mempelajari pembuatan API menggunakan Laravel 11 serta pengujian API secara publik dengan menggunakan Ngrok. Dalam praktik ini, database iot\_25 dibuat di phpMyAdmin, kemudian model TransaksiSensor dikembangkan untuk menyimpan data sensor. API dikembangkan menggunakan resource controller dan diuji dengan Postman. Hasil akhir menunjukkan bahwa API dapat diakses melalui internet menggunakan Ngrok dan berfungsi untuk CRUD data sensor.

Keywords— Laravel 11, API, Ngrok, RESTful, IoT

## **1. Introduction (Pendahuluan)**

### **1.1 Latar Belakang**

Dalam pengembangan IoT, API sangat penting untuk menghubungkan perangkat dengan sistem berbasis cloud. Laravel 11 adalah framework PHP yang mendukung pembuatan API dengan mudah dan efisien. Ngrok digunakan untuk meng-online-kan API, sehingga bisa diakses dari perangkat IoT atau simulator seperti Wokwi.

### **1.2 Tujuan**

1. Membuat API menggunakan Laravel 11 untuk menyimpan dan mengelola data sensor.
2. Menguji API dengan Postman.
3. Menggunakan Ngrok untuk menyediakan akses API secara publik.

## **2. Methodology (Metodologi)**

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

1. **Framework :** Laravel 11
2. **Database :** MySQL (phpMyAdmin)
3. **Server :** Laravel built-in server (php artisan serve)
4. **Testing Tools :** Postman
5. **Ngrok :** Untuk menghubungkan API ke internet

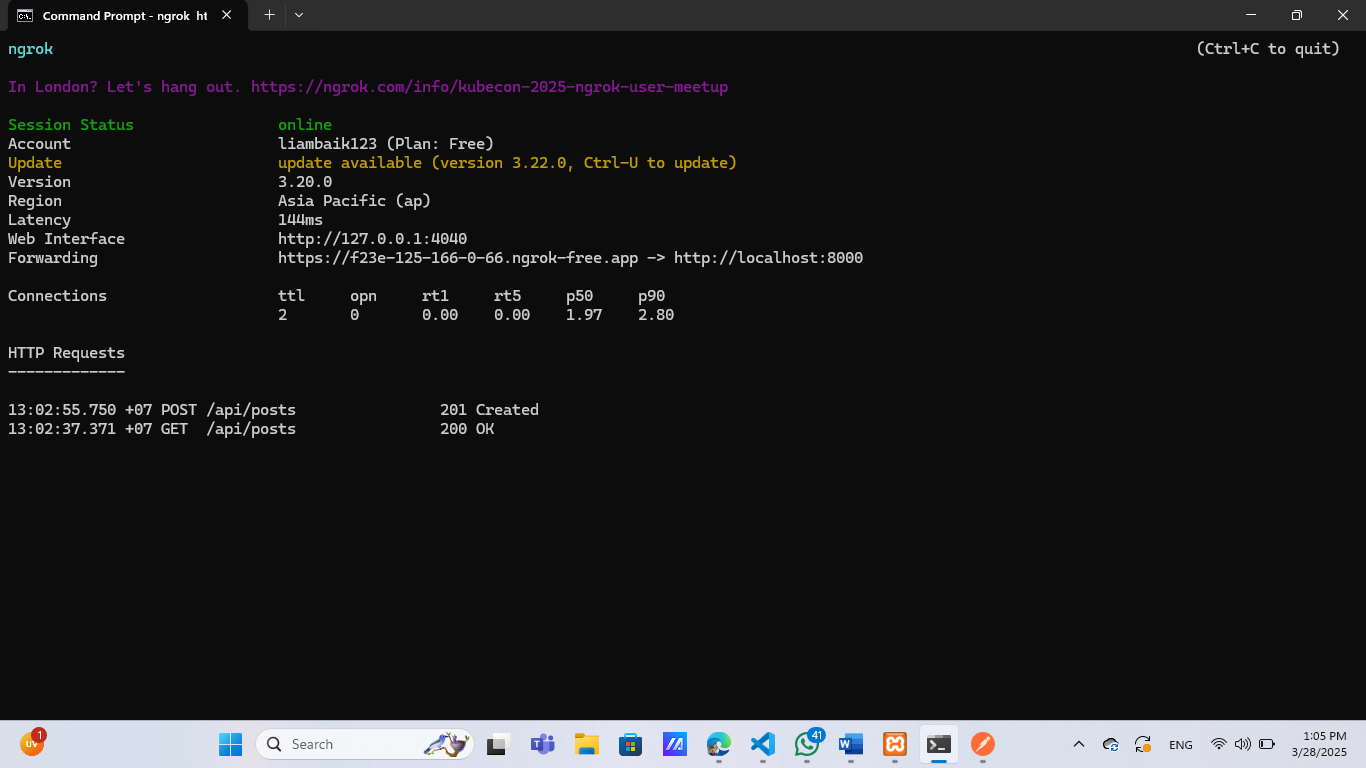
### **2.2 Implementation Steps (Langkah Implementasi)**

1. **Membuat Database**
   1. Buat database di phpMyAdmin dengan nama iot\_25.
2. **Membuat Model dan Migration**
   1. Jalankan perintah berikut:
   2. php artisan make:model TransaksiSensor -m
   3. Edit file 2025\_02\_21\_074123\_create\_transaksi\_sensors\_table.php sesuai kebutuhan.
   4. Jalankan perintah migrasi:
   5. php artisan migrate
3. **Membuat Resource Controller**
   1. Jalankan perintah:
   2. php artisan make:controller Api/TransaksiSensorController --resource
   3. Implementasikan metode CRUD di controller.
4. **Membuat Resource untuk API**
   1. Jalankan perintah:
   2. php artisan make:resource TransaksiSensorResource
   3. Sesuaikan format respons JSON di file TransaksiSensorResource.php.
5. **Menentukan Routes API**
   1. Buka file routes/api.php dan tambahkan route berikut:
   2. Route::apiResource('/posts', App\Http\Controllers\Api\TransaksiSensorController::class);
   3. Periksa daftar routes dengan:
   4. php artisan route:list
6. **Menjalankan Server dan Menguji API dengan Postman**
   1. Jalankan Laravel dengan:
   2. php artisan serve
   3. Gunakan Postman untuk mengakses http://127.0.0.1:8000/api/posts.
7. **Menggunakan Ngrok untuk Publikasi API**
   1. Install dan login ke Ngrok.
   2. Jalankan perintah:
   3. ngrok http http://localhost:8000
   4. Gunakan URL publik yang diberikan oleh Ngrok untuk mengakses API dari luar jaringan lokal.

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

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

1. API berhasil dibuat dan diuji menggunakan Postman.
2. Endpoint GET, POST, PUT, dan DELETE berfungsi dengan baik.
3. API dapat diakses secara publik menggunakan URL Ngrok.

**Screenshot hasil simulasi:**  
****

## **4. Appendix (Lampiran)**

### **4.1 JSON Respons API**

{

"data": [

{

"id": 3,

"nama\_sensor": "Sensor B",

"nilai1": 87,

"nilai2": 176

},

{

"id": 2,

"nama\_sensor": "Sensor A",

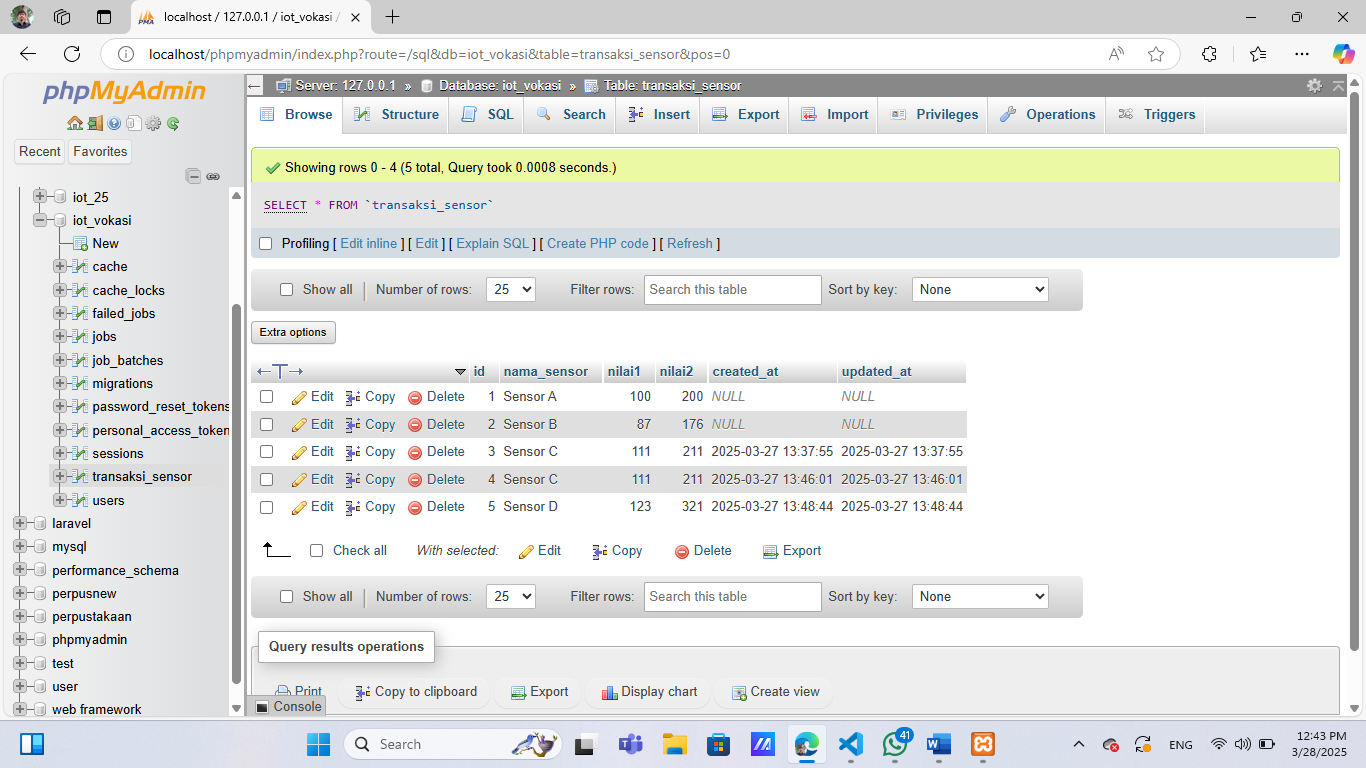
"nilai1": 100,

"nilai2": 200

}

]

}

****