# LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)

Fakultas Vokasi, Universitas Brawijaya

**PRAKTIK PEMBUATAN API UNTUK IoT DENGAN LARAVEL**

*BAHYDZAKY FARREL JOCELLIN*

*Fakultas Vokasi, Universitas Brawijaya*

*Email: [fjocellin9@gmail.com](mailto:fjocellin9@gmail.com)*

**Abstrak**

Laporan ini membahas implementasi pembuatan API menggunakan Laravel untuk mendukung sistem IoT. API ini digunakan untuk menyimpan dan mengelola data dari sensor yang dikirimkan ke server Laravel. Proses ini mencakup pembuatan database, model, migration, resource, controller, serta pengujian API menggunakan Postman. Selain itu, API juga dihosting secara publik menggunakan Ngrok untuk memungkinkan akses dari perangkat IoT atau simulator seperti Wokwi ESP32. Hasil pengujian menunjukkan bahwa API dapat menangani permintaan GET, POST, PUT, dan DELETE dengan baik, serta dapat diakses melalui jaringan publik.

**Kata kunci**: IoT, Laravel, API, Postman, Ngrok

**Abstraction**

This report discusses the implementation of API development using Laravel to support an IoT system. The API is used to store and manage sensor data sent to the Laravel server. This process includes database creation, model development, migration, resource management, controller implementation, and API testing using Postman. Additionally, the API is publicly hosted using Ngrok to enable access from IoT devices or simulators such as Wokwi ESP32. The test results show that the API can efficiently handle GET, POST, PUT, and DELETE requests and is accessible through a public network.

**Keywords**: IoT, Laravel, API, Postman, Ngrok

**1. Pendahuluan**

* 1. **Latar belakang**

Dalam implementasi Internet of Things (IoT), komunikasi antara perangkat sensor dan server sangat penting untuk mengelola data secara efisien. Salah satu cara untuk menghubungkan perangkat IoT dengan server adalah melalui API berbasis RESTful yang memungkinkan pengiriman dan pengambilan data dari database. Laravel sebagai framework PHP menyediakan fitur untuk membangun API dengan cepat dan efisien.

Pada praktik ini, dibuat API untuk menyimpan data dari sensor ke database menggunakan Laravel. API ini memungkinkan perangkat IoT untuk mengirimkan data sensor dan mengambil informasi yang tersimpan melalui HTTP request. Pengujian dilakukan menggunakan Postman, serta API dihosting secara publik menggunakan Ngrok agar dapat diakses dari perangkat lain.

**1.2 Tujuan eksperimen**

Praktikum ini bertujuan untuk mengembangkan API menggunakan Laravel guna mendukung komunikasi antara perangkat IoT dan server. Dalam praktik ini, dilakukan berbagai tahap pengembangan, termasuk pembuatan database dan model untuk menyimpan data sensor, pengembangan API untuk menangani operasi CRUD, serta pengujian API menggunakan Postman. Selain itu, API yang telah dibuat akan dihosting secara publik menggunakan Ngrok agar dapat diakses dari perangkat lain. Dengan adanya API ini, diharapkan integrasi antara perangkat IoT dan server dapat berjalan dengan lancar dan efisien.

**2. Metodologi**

**2.1 Alat dan Bahan**

1. Laravel
2. MySQL (phpMyAdmin)
3. Postman
4. Ngrok
5. Wokwi ESP32 (untuk simulasi perangkat IoT)
   1. **Implementation Steps (Langkah Implementasi)**

**2.2.1 Pembuatan Database**

1. Buka phpMyAdmin dan buat database baru dengan nama iot\_25.

**2.2.2 Pembuatan Model dan Migration**

1. Jalankan perintah berikut untuk membuat model TransaksiSensor:

php artisan make:model TransaksiSensor -m

1. Ubah file migration (2025\_02\_21\_074123\_create\_transaksi\_sensors\_table.php) dengan struktur berikut:

Schema::create('transaksi\_sensors', function (Blueprint $table) {

$table->id();

$table->string('nama\_sensor', 255);

$table->integer('nilai1');

$table->integer('nilai2');

$table->timestamps();

});

1. Jalankan perintah untuk membuat tabel di database:

php artisan migrate

**2.2.3 Pembuatan API Resource dan Controller**

1. Buat resource API:

php artisan make:resource TransaksiSensorResource

1. Buat controller API:

php artisan make:controller Api/TransaksiSensorController --api

1. Implementasikan fungsi CRUD dalam TransaksiSensorController.php.

**2.2.4 Konfigurasi Route API**

1. Buka file routes/api.php dan tambahkan route berikut:

Route::apiResource('/sensor', App\Http\Controllers\Api\TransaksiSensorController::class);

1. Jalankan perintah untuk memastikan route sudah terdaftar:

php artisan route:list

**2.2.5 Pengujian API Menggunakan Postman**

1. Jalankan server Laravel:

php artisan serve

1. Gunakan Postman untuk melakukan request:

**GET**: http://127.0.0.1:8000/api/sensor

**POST**: http://127.0.0.1:8000/api/sensor dengan data JSON:

{

"nama\_sensor": "Sensor A",

"nilai1": 100,

"nilai2": 200

}

**PUT**: Update data tertentu dengan ID

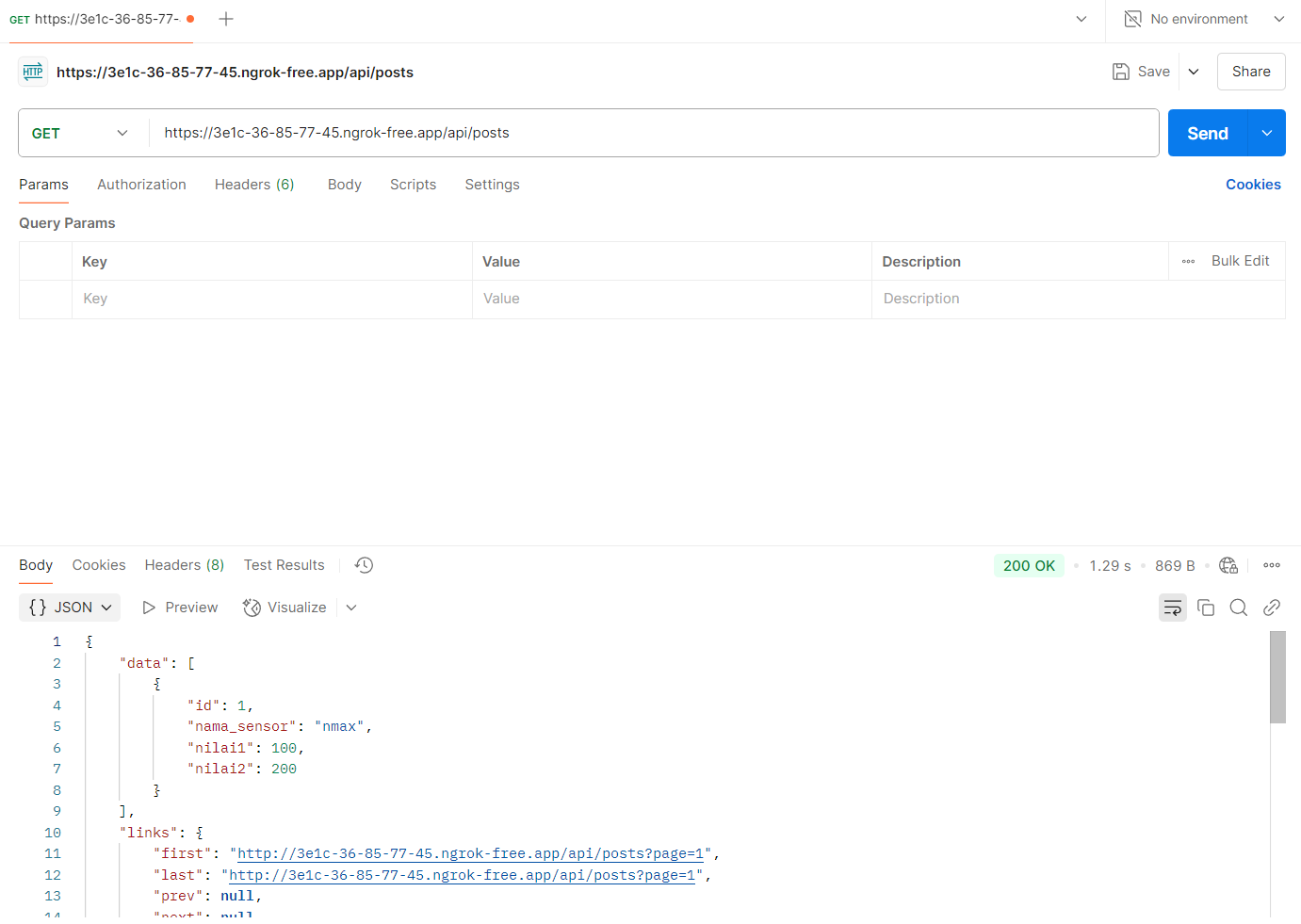
**DELETE**: Menghapus data berdasarkan ID

**2.2.6 Hosting API Secara Publik Menggunakan Ngrok**

1. Unduh dan install Ngrok dari <https://ngrok.com/>.
2. Jalankan perintah berikut untuk menghubungkan Laravel dengan Ngrok:

ngrok http 8000

1. Gunakan URL yang diberikan oleh Ngrok untuk mengakses API dari perangkat IoT.
2. **Results and Discussion (Hasil dan Pembahasan)**



Gambar 3.1

**3.1 Experimental Results (Hasil Eksperimen)**

Hasil pengujian menunjukkan bahwa API yang dikembangkan menggunakan Laravel dapat berfungsi dengan baik dalam menangani operasi CRUD. Melalui pengujian menggunakan Postman, API mampu menerima permintaan GET untuk mengambil data, POST untuk menambahkan data, PUT untuk memperbarui data, serta DELETE untuk menghapus data dari database. Data yang dikirimkan ke API dapat tersimpan dengan benar dalam database MySQL, dan setiap permintaan yang dilakukan memberikan respons yang sesuai. Selain itu, API ini berhasil dihosting secara publik menggunakan Ngrok, memungkinkan akses dari perangkat luar melalui jaringan internet.

**3.2 Pembahasan**

Implementasi API menggunakan Laravel terbukti efisien karena framework ini menyediakan berbagai fitur bawaan yang mendukung pengembangan API secara cepat dan terstruktur. Penggunaan resource dan controller dalam Laravel mempermudah pengelolaan data dan meningkatkan keterbacaan kode. Pengujian API dengan Postman juga sangat membantu dalam memastikan setiap endpoint berfungsi sebagaimana mestinya sebelum API dihubungkan dengan perangkat IoT. Selain itu, Ngrok berperan penting dalam mengizinkan akses publik ke API, sehingga perangkat IoT atau simulator seperti Wokwi ESP32 dapat mengakses API tanpa perlu konfigurasi jaringan yang kompleks. Dengan demikian, penggunaan Laravel dan Ngrok dalam pengembangan API IoT ini menjadi solusi yang efektif untuk integrasi sistem berbasis IoT dengan server berbasis web.