LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)

Fakultas Vokasi, Universitas Brawijaya

**Praktik Simulasi Relay, Button & LED**

*Nadila Yanuarika Rimawati*

*Fakultas Vokasi, Universitas Brawijaya*

*Email:* [*nadiladila751@student.ub.ac.id*](mailto:nadiladila751@student.ub.ac.id)

**Abstract**

*Simulasi* dan pemrograman mikrokontroler menjadi aspek penting dalam pengembangan aplikasi *otomatisasi*. Praktikum ini bertujuan untuk mensimulasikan penggunaan *relay*, *push button*, dan *LED* menggunakan *ESP32* dalam lingkungan *simulasi* Wokwi. Dalam eksperimen ini, *push button* digunakan sebagai input untuk mengaktifkan *relay* dan menyalakan *LED* sebagai indikator status. Implementasi ini bertujuan untuk memahami bagaimana *ESP32* dapat mengontrol perangkat eksternal melalui *relay*, yang sering digunakan dalam sistem otomatisasi seperti kontrol beban listrik. Hasil *simulasi* menunjukkan bahwa sistem dapat bekerja sesuai dengan desain, di mana menekan tombol akan mengaktifkan *relay* dan *LED*, serta melepaskannya akan memutuskan arus ke beban.

**Kata kunci**: ESP32, Relay, Push Button, LED, Simulasi, Otomatisasi.

**1. Introduction**

* 1. **Latar belakang**

Dalam dunia otomasi, relay digunakan untuk mengontrol perangkat listrik dengan sinyal dari mikrokontroler, sementara push button berfungsi sebagai input manual dan LED sebagai indikator. Memahami kerja sama antara komponen-komponen ini sangat penting dalam pengembangan sistem kendali berbasis mikrokontroler. Perkembangan teknologi juga mendorong penggunaan simulator seperti Wokwi, yang mempermudah pengujian rangkaian sebelum implementasi fisik. Maka, praktikum ini dilakukan untuk mensimulasikan kontrol relay, button, dan LED menggunakan ESP32, sehingga memberikan pengalaman dalam pemrograman dan pemahaman tentang sistem kendali sederhana.

* 1. **Tujuan eksperimen**

1. Mensimulasikan rangkaian relay, push button, dan LED dengan ESP32.
2. Memahami cara kerja relay dalam mengontrol perangkat Listrik.
3. Memprogram ESP32 untuk membaca input tombol dan mengontrol relay serta LED.
4. Mengamati cara kerja sistem dalam simulasi.

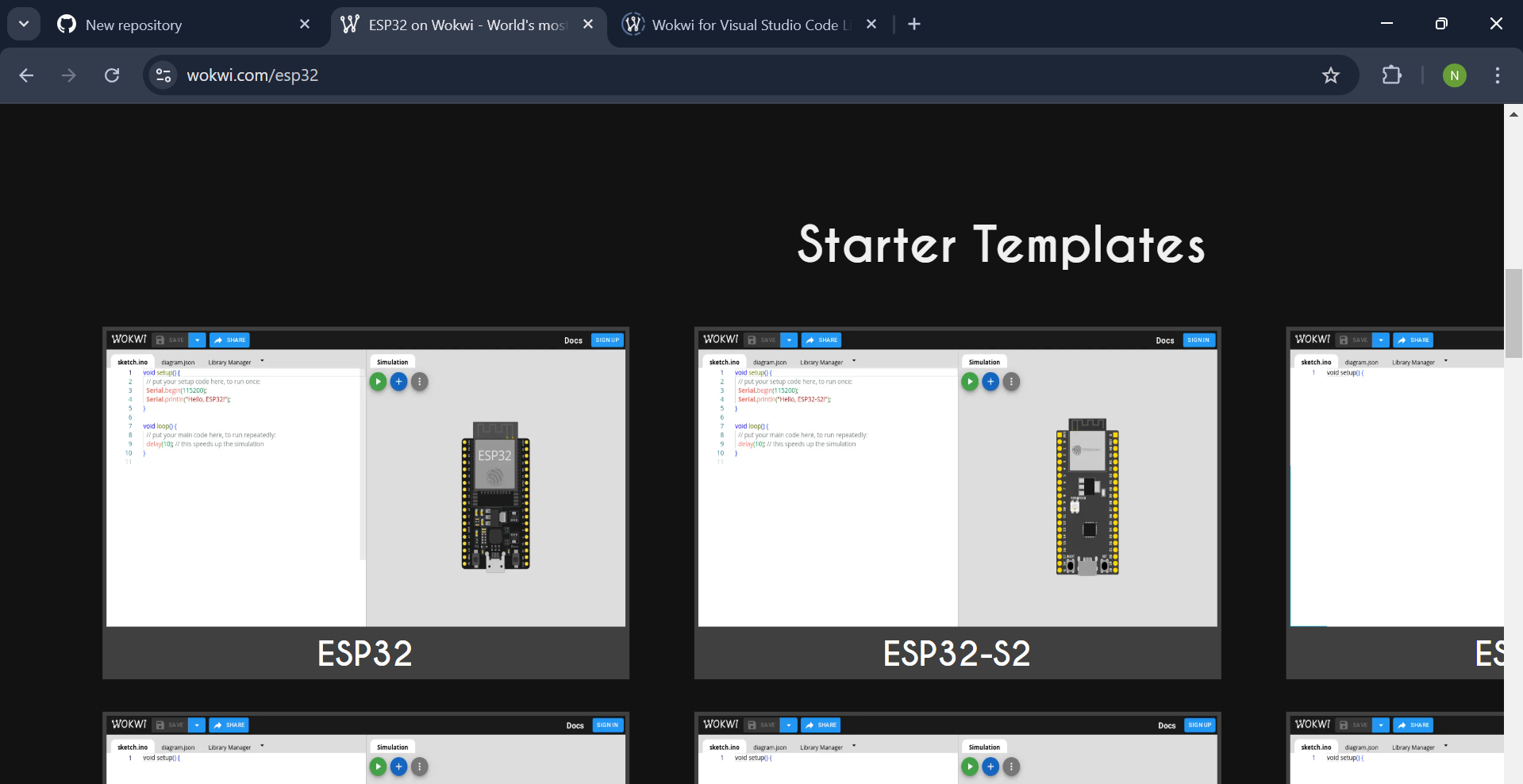
**2. Methodology**

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

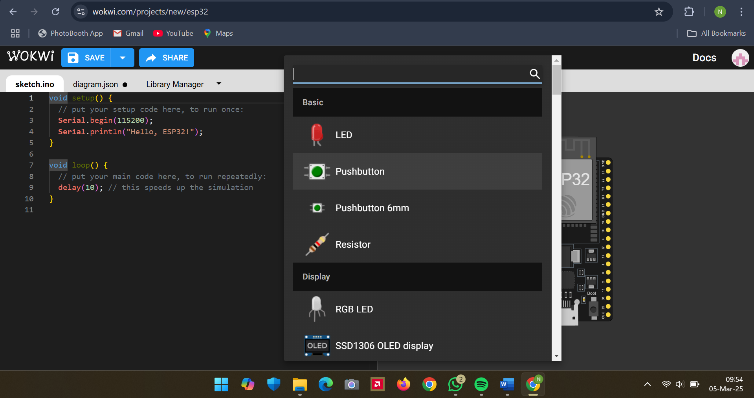
Laptop atau PC, koneksi internet, Browser Web (Wokwi.com), VS Code (editor kode), PlatformIO & Akun wokwi, kode program dalam bahasa C++, file diagram rangkaian (diagram.json).

* 1. **Implementation Steps (Langkah Implementasi)**

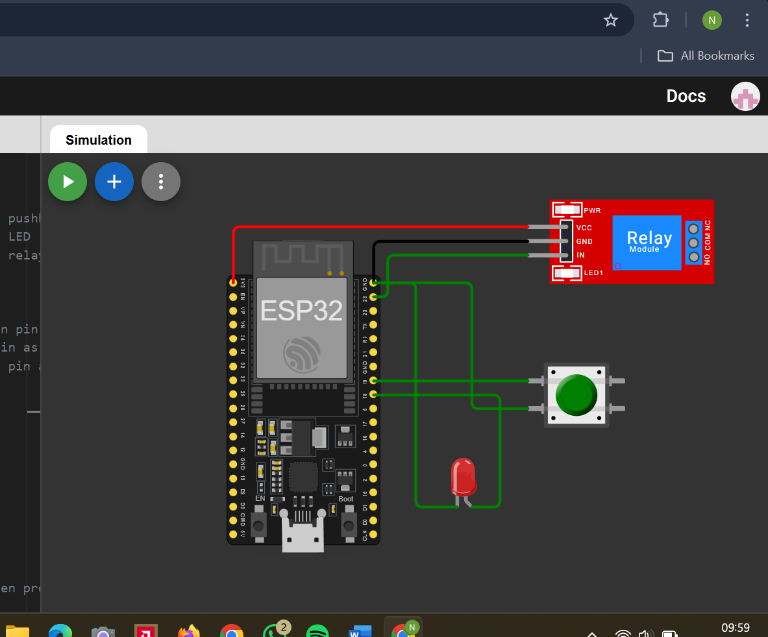
1. Masuk akun Wokwi.com, klik ESP32, lalu pilih template ESP32.



1. Tambahkan part LED, relay, push button ke simulasi.



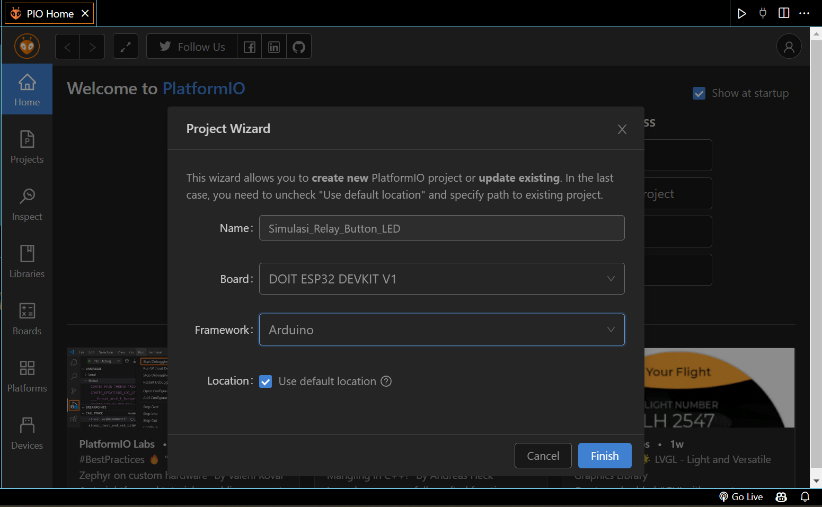
1. Hubungkan kabel relay ke ESP32, hubungkan VVC sensor ke pin 3V3 ESP32, lalu hubungkan pin data sensor ke pin digital 27, terakhir GND sensor ke GND ESP.



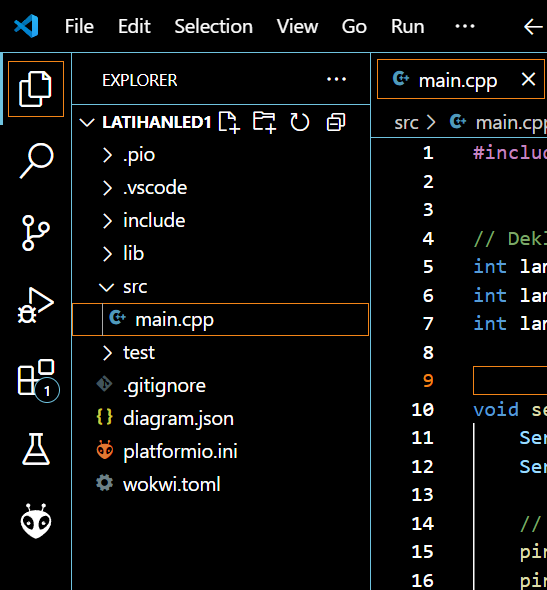
1. Masukkan kode program di bagian sketch.io.



1. Lalu isi parameter berikut :



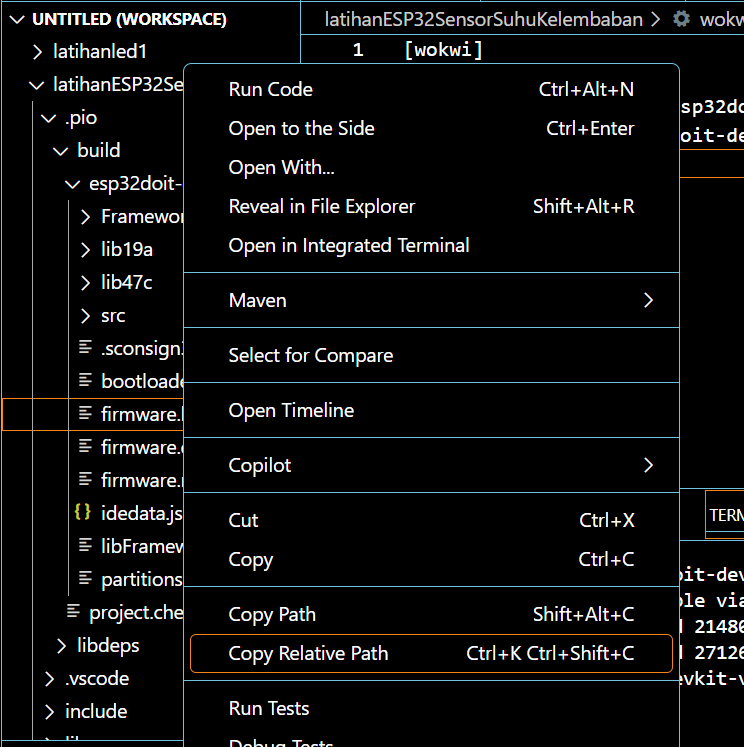
1. Salin kode pada Langkah 4 ke file main.cpp



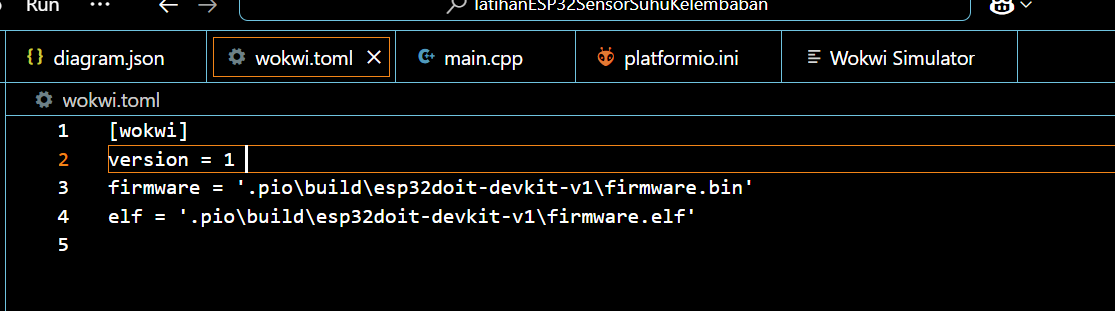
1. Lalu buat file diagram.json, untuk isi dari file copas dari diagram.json pada wokwi.com platform, setelah terisi jangan lupa untuk centang terlebih dahulu pada vscode sampai tertera “SUCCESS”.

****

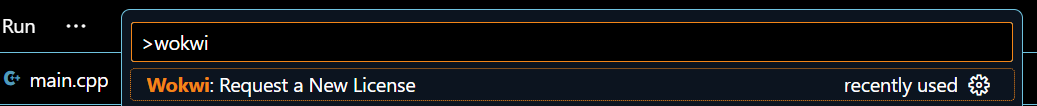
1. Lalu masuk pada .pio/build/esp32doit-devkit-v1 terdapat firmware.bin dan firmware.elf, klik kanan dua file tersebut lalu tekan “Copy Relative Path”.



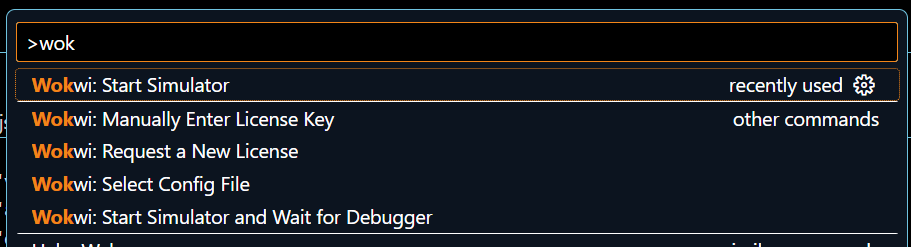
1. Salin relative path kedua file ke dalam file wokwi.toml, sebelum itu buat file wokwi.toml pada luar folder dan isi sesuai relative path pada bagian firmware dan elf.



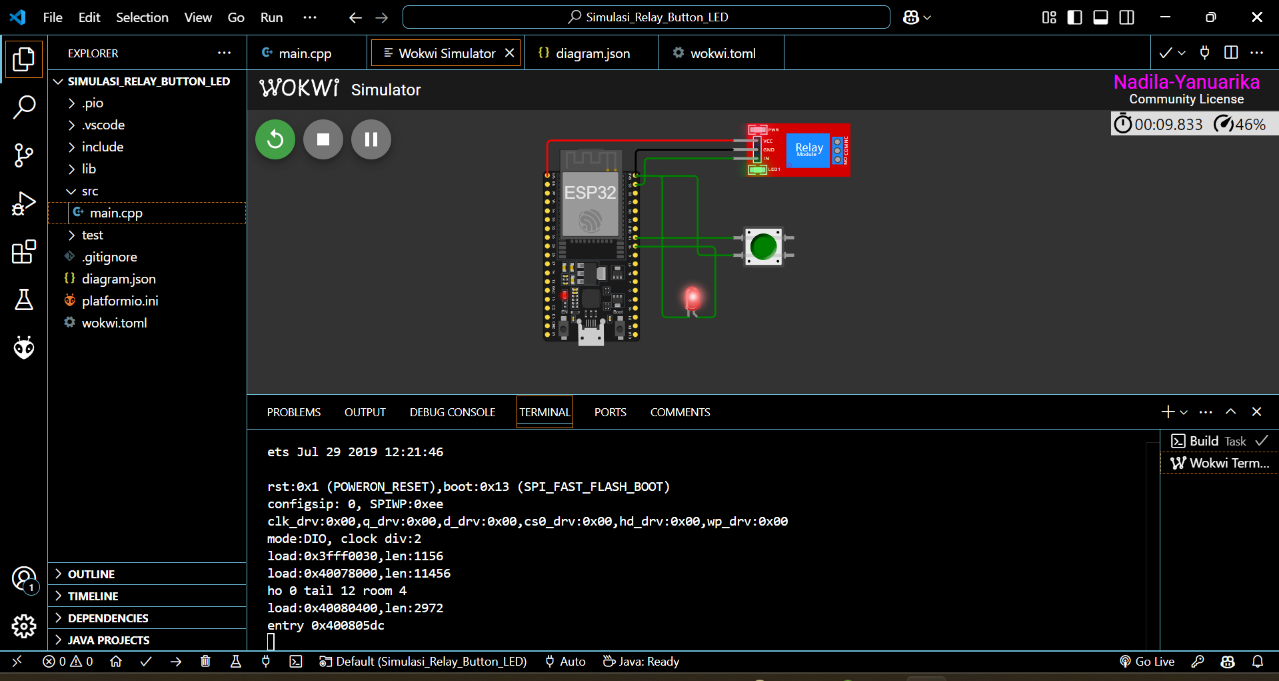
1. Langkah berikutnya lakukan request license ke wokwi.com



1. Selanjutnya menjalankan simulasi dengan mengetik pada command >Wokwi: Start Simulator



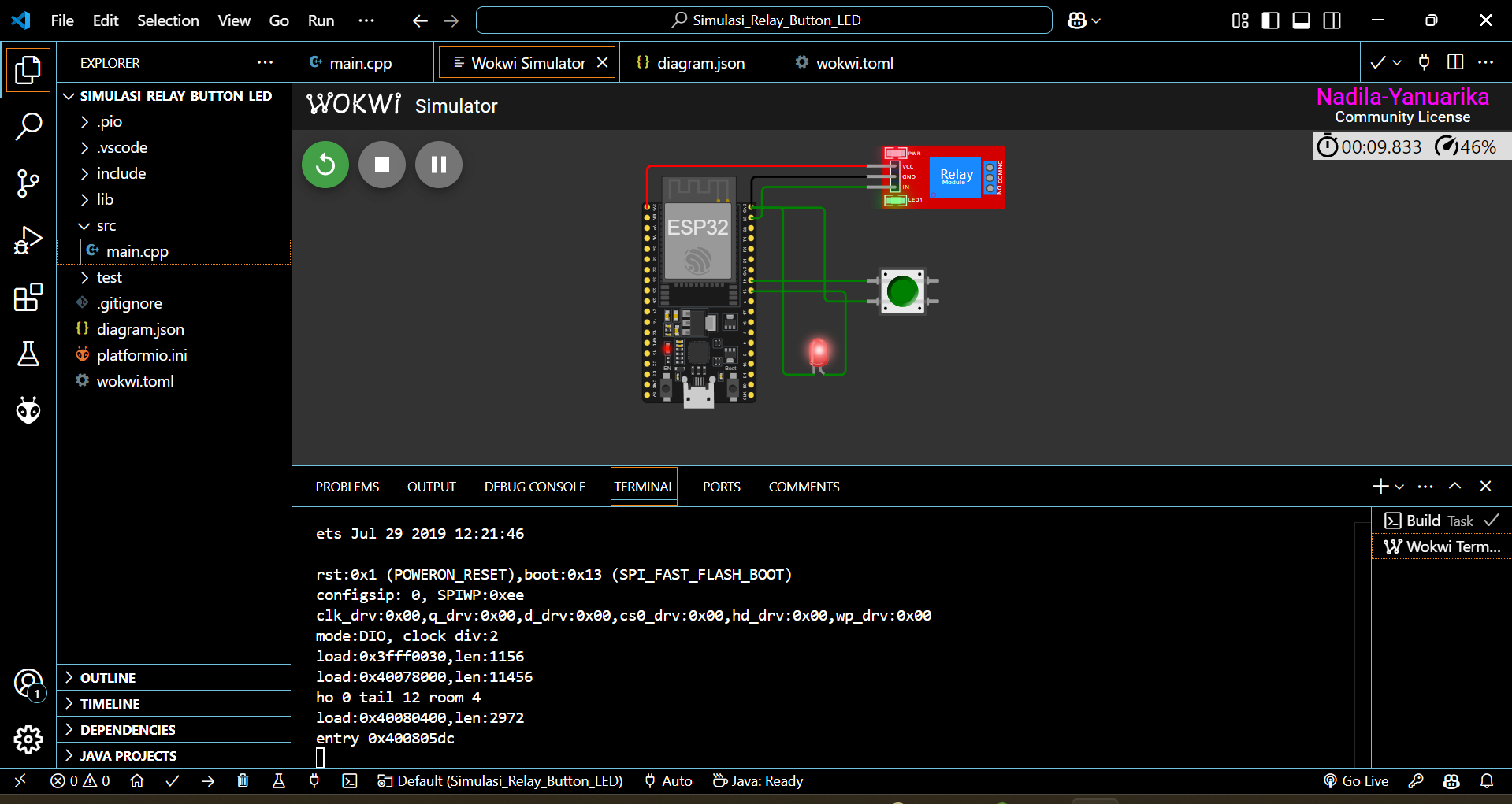
1. Terakhir simulasi sensor relay, button, dan LED akan berjalan sesuai koding dan diagram yang dirangkai.



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

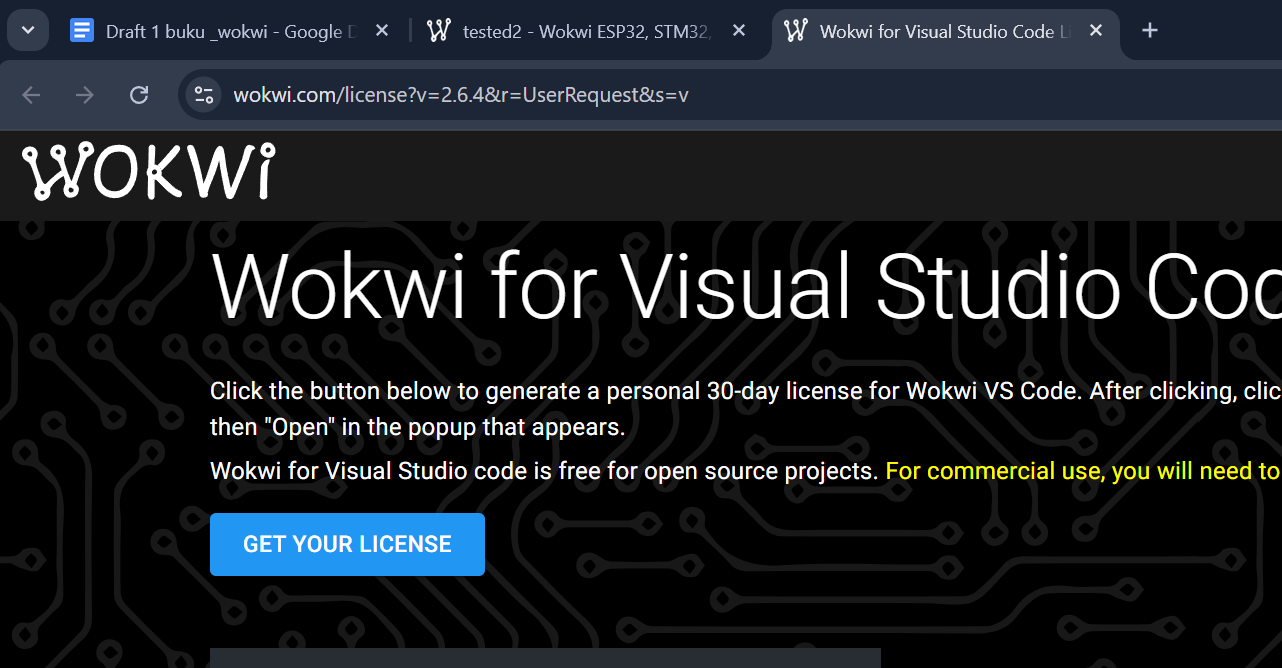
**3.1 Experimental Results (Hasil Eksperimen)**

Sensor suhu kelembaban berfungsi



**4. Appendix (Lampiran)**

Halaman Get Your License



LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)

Fakultas Vokasi, Universitas Brawijaya

**Praktik Simulasi Sensor Jarak (Ultrasonic)**

*Nadila Yanuarika Rimawati*

*Fakultas Vokasi, Universitas Brawijaya*

*Email:* [*nadiladila751@student.ub.ac.id*](mailto:nadiladila751@student.ub.ac.id)

**Abstract**

*Simulasi* dan pemrograman mikrokontroler berperan penting dalam pengembangan sistem otomatisasi. Praktikum ini bertujuan untuk mensimulasikan penggunaan *sensor jarak ultrasonik* *(HC-SR04)* menggunakan *ESP32* dalam lingkungan Wokwi. *Sensor ultrasonik* digunakan untuk mengukur jarak objek dan menampilkan hasil pembacaan melalui serial monitor. Implementasi ini bertujuan untuk memahami cara kerja sensor ultrasonik dalam mendeteksi objek serta bagaimana *ESP32* memproses data untuk aplikasi *otomatisasi*. Hasil simulasi menunjukkan bahwa sistem dapat membaca jarak hingga batas maksimal sensor, namun terdapat kemungkinan kendala pada akurasi pembacaan jika objek berada di luar jangkauan efektif.

**Kata kunci**: ESP32, Sensor Ultrasonik, HC-SR04, Simulasi, Otomatisasi.

**1. Introduction**

* 1. **Latar belakang**

Teknologi mikrokontroler berkembang pesat dan banyak digunakan dalam sistem otomatisasi, termasuk pengukuran jarak menggunakan sensor ultrasonik seperti HC-SR04, yang sering dimanfaatkan dalam parkir otomatis dan robotika. ESP32, dengan kemampuan pemrosesan dan konektivitasnya, dapat mengolah data dari sensor ini untuk berbagai keperluan. Praktikum ini mensimulasikan penggunaan sensor ultrasonik dengan ESP32 di Wokwi untuk memahami cara kerja sensor dalam mendeteksi jarak objek. Hasil eksperimen diharapkan memberikan pemahaman tentang penerapan sensor jarak serta manfaat simulasi sebelum implementasi pada perangkat keras nyata.

* 1. **Tujuan eksperimen**

1. Mensimulasikan penggunaan sensor ultrasonik HC-SR04 dengan ESP32 dalam lingkungan Wokwi.
2. Memahami cara kerja sensor ultrasonik dalam mengukur jarak objek.
3. Menguji kemampuan ESP32 dalam membaca dan memproses data sensor.
4. Mengamati cara kerja sistem dalam simulasi.

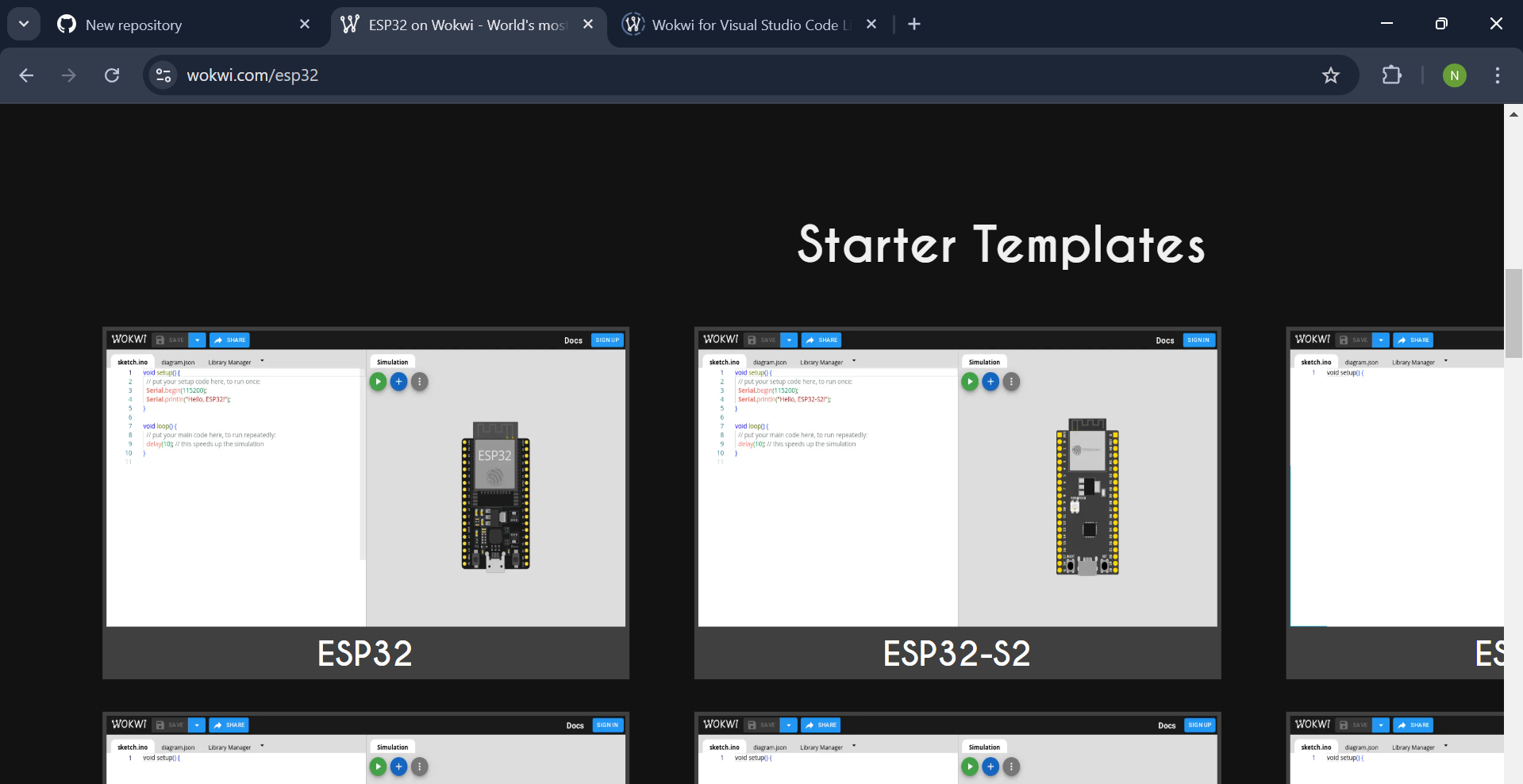
**2. Methodology**

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

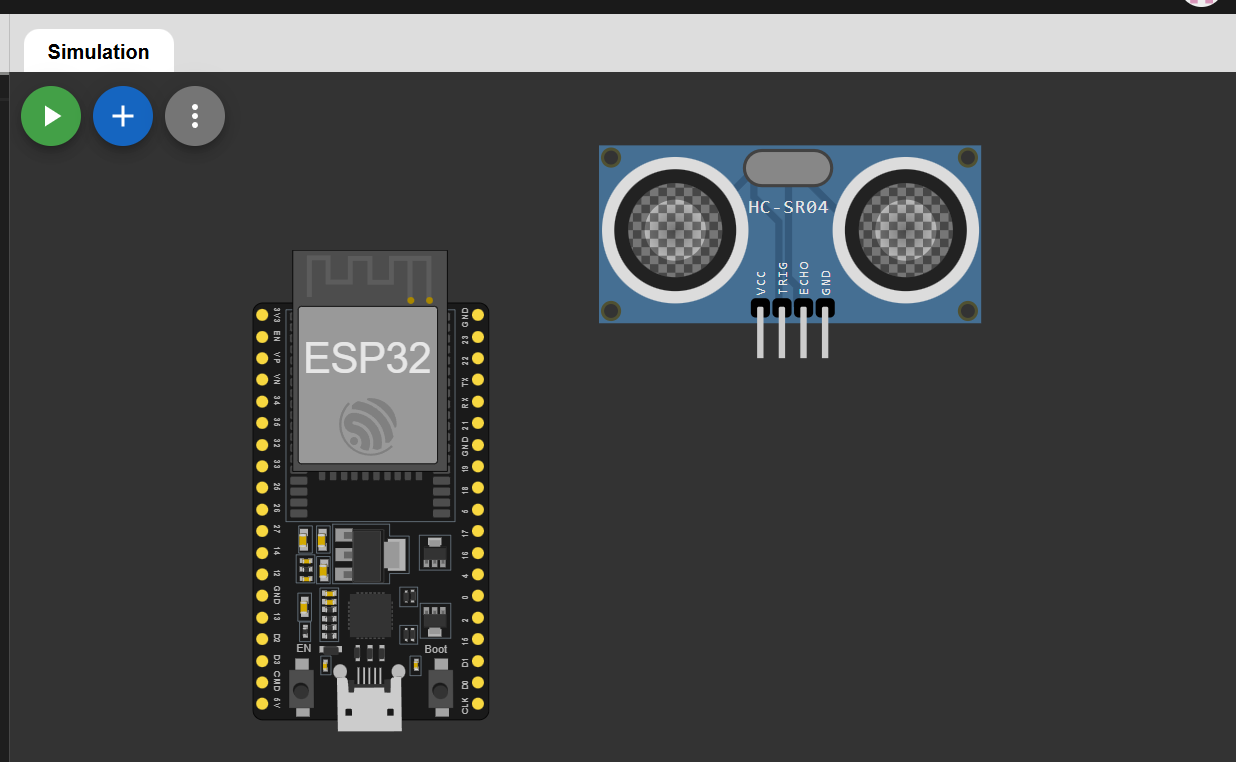
Laptop atau PC, koneksi internet, Browser Web (Wokwi.com), VS Code (editor kode), PlatformIO & Akun wokwi, kode program dalam bahasa C++, file diagram rangkaian (diagram.json).

* 1. **Implementation Steps (Langkah Implementasi)**

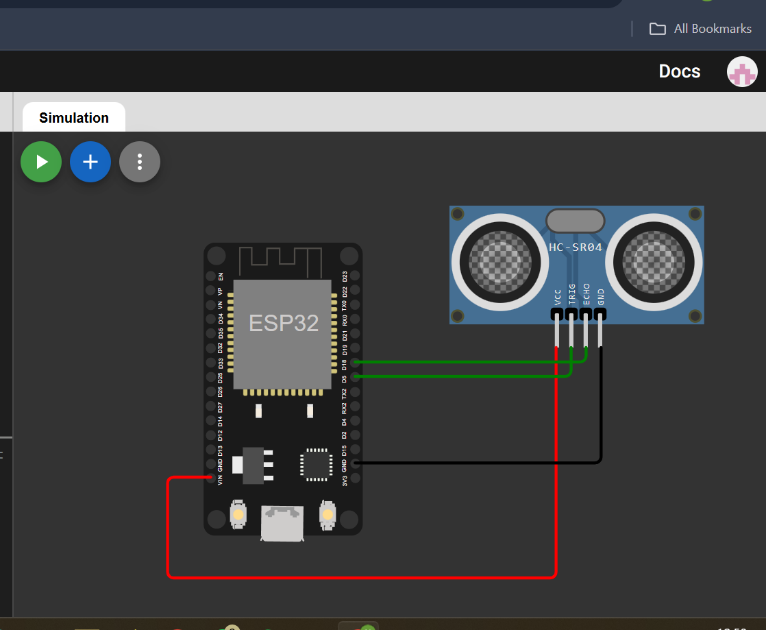
1. Masuk akun Wokwi.com, klik ESP32, lalu pilih template ESP32.



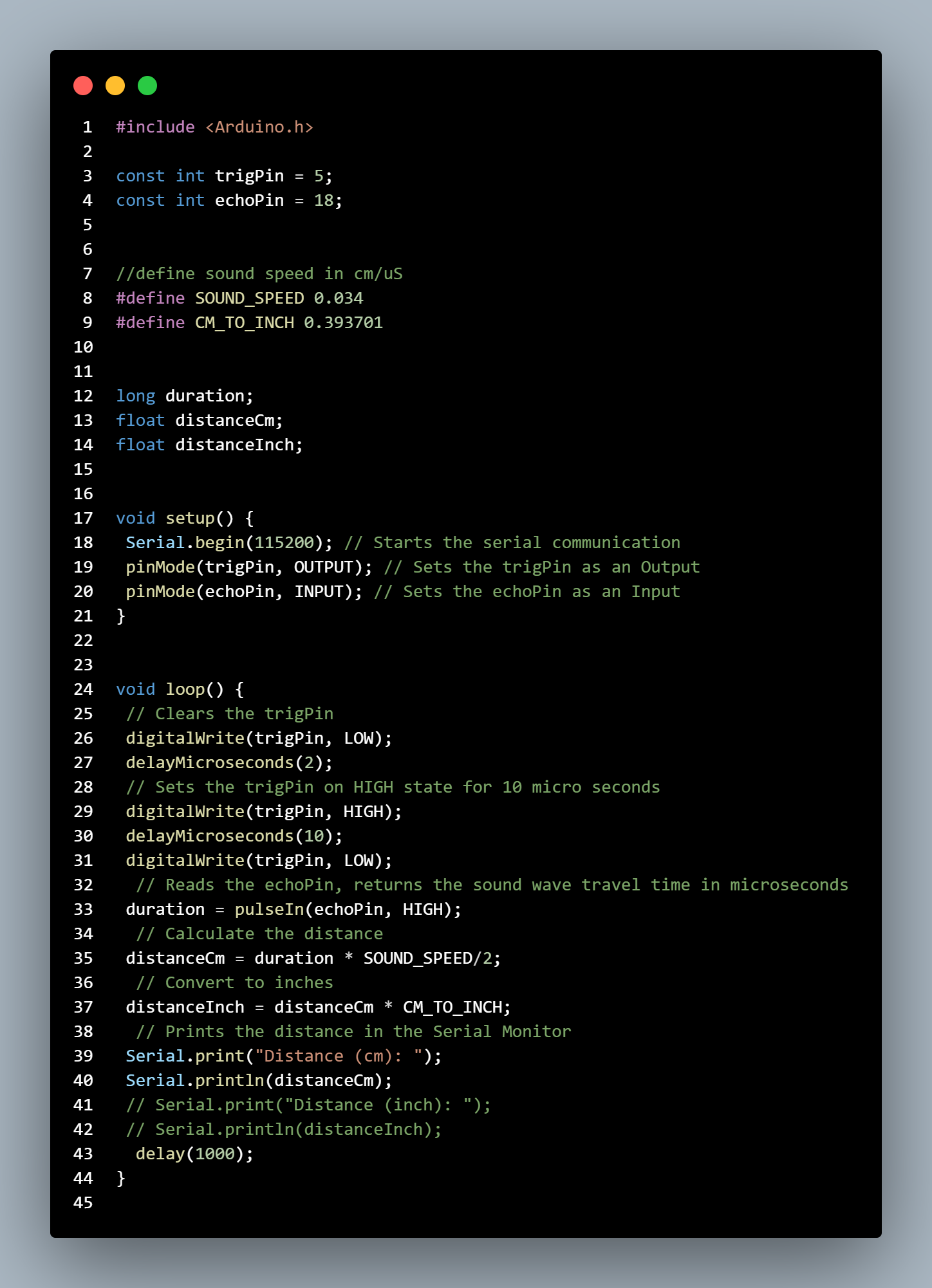
1. Tambahkan part sensor ultrasonik HC-SR04 ke simulasi.



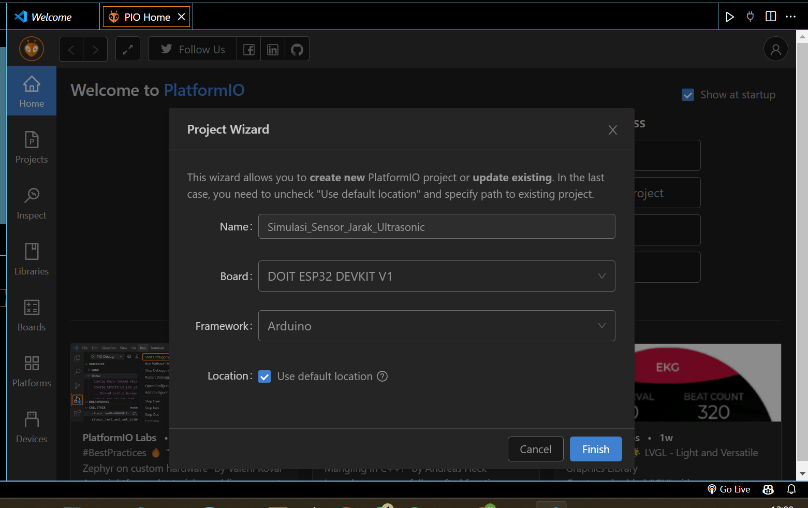
1. Hubungkan kabel ultrasonik HC-SR04 ke ESP32, hubungkan VVC sensor ke pin VIIN ESP32, lalu hubungkan pin TRIG sensor ke pin D5, selanjutnya hubungkan sensor ECHO ke pin D18, terakhir GND sensor ke GND ESP.



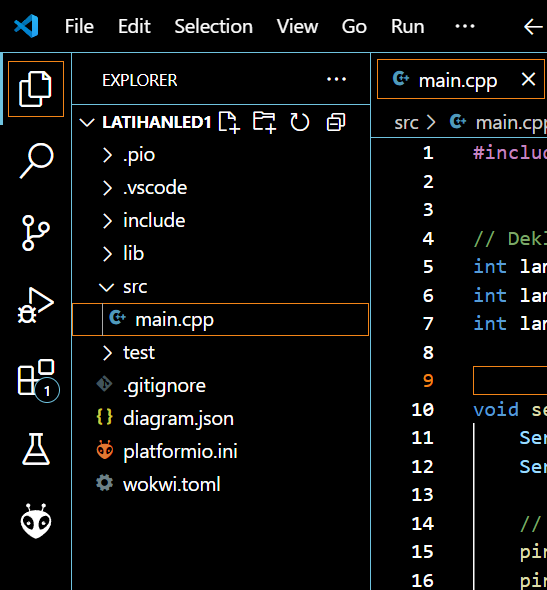
1. Masukkan kode program di bagian sketch.io.



1. Lalu isi parameter berikut :



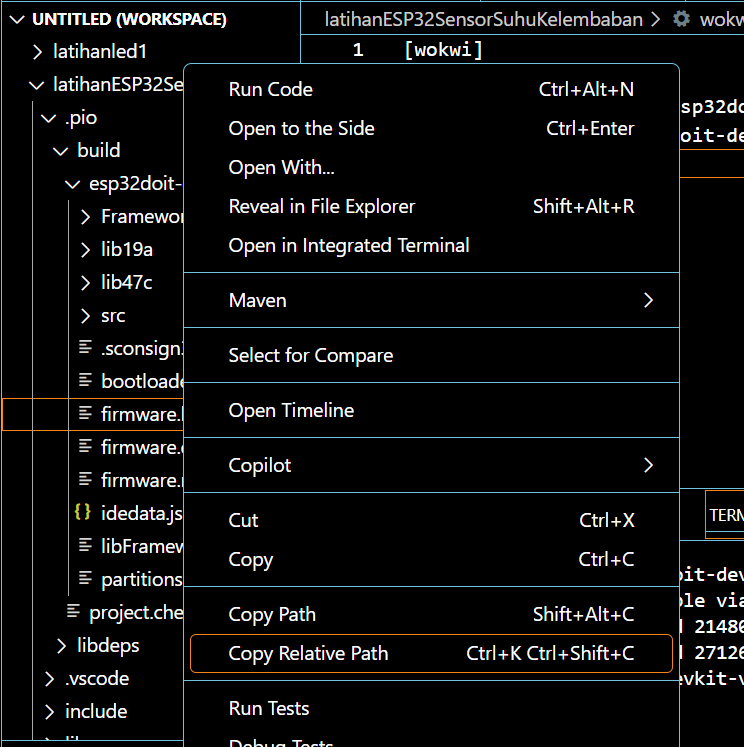
1. Salin kode pada Langkah 4 ke file main.cpp



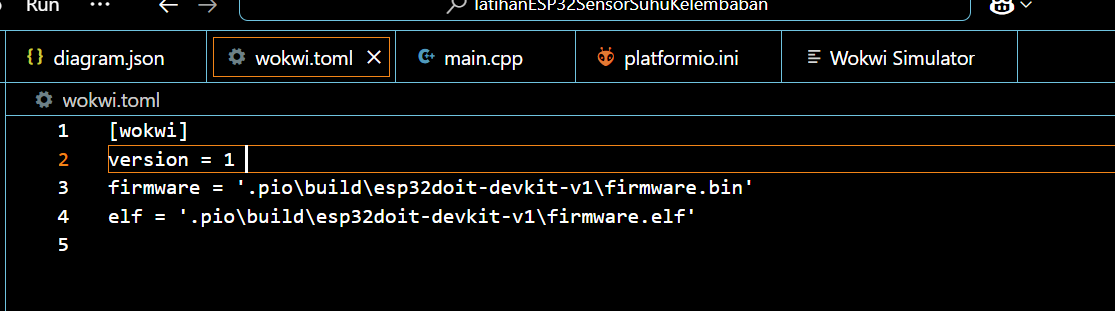
1. Lalu buat file diagram.json, untuk isi dari file copas dari diagram.json pada wokwi.com platform, setelah terisi jangan lupa untuk centang terlebih dahulu pada vscode sampai tertera “SUCCESS”.

****

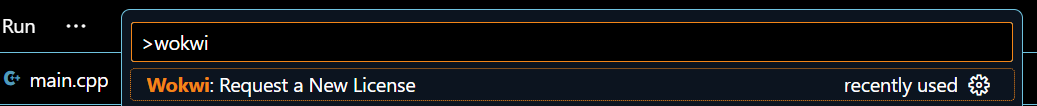
1. Lalu masuk pada .pio/build/esp32doit-devkit-v1 terdapat firmware.bin dan firmware.elf, klik kanan dua file tersebut lalu tekan “Copy Relative Path”.



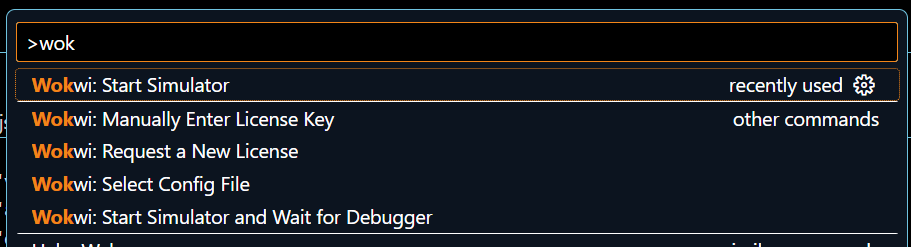
1. Salin relative path kedua file ke dalam file wokwi.toml, sebelum itu buat file wokwi.toml pada luar folder dan isi sesuai relative path pada bagian firmware dan elf.



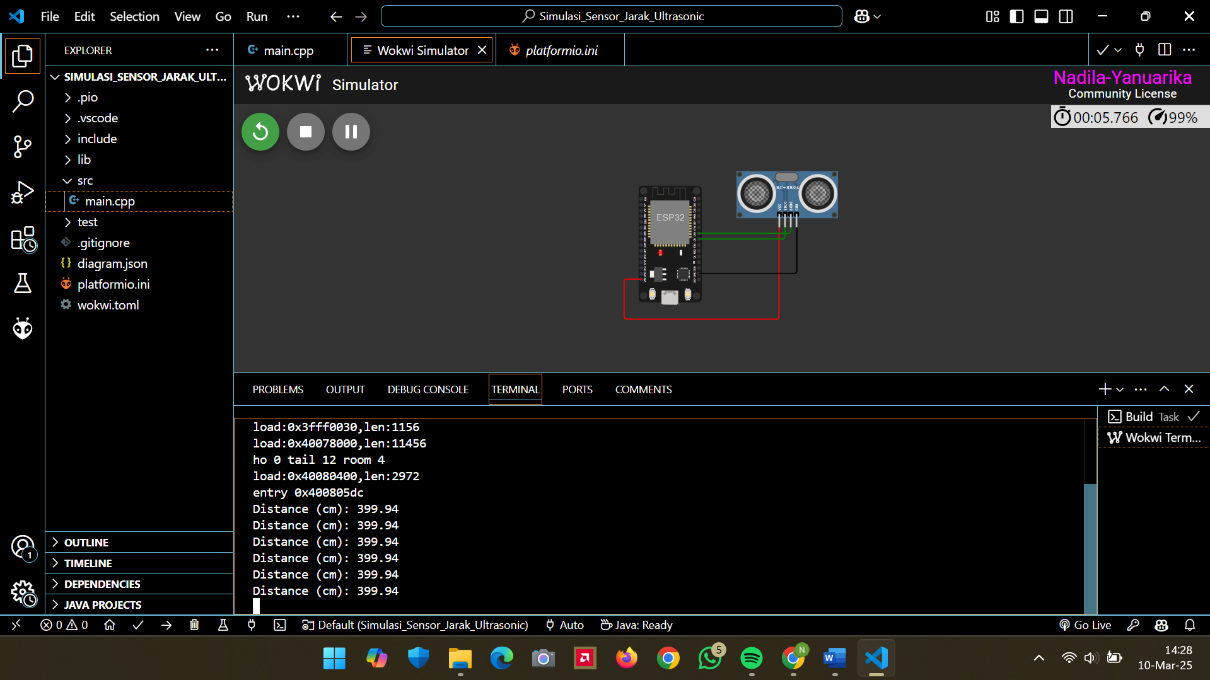
1. Langkah berikutnya lakukan request license ke wokwi.com



1. Selanjutnya menjalankan simulasi dengan mengetik pada command >Wokwi: Start Simulator



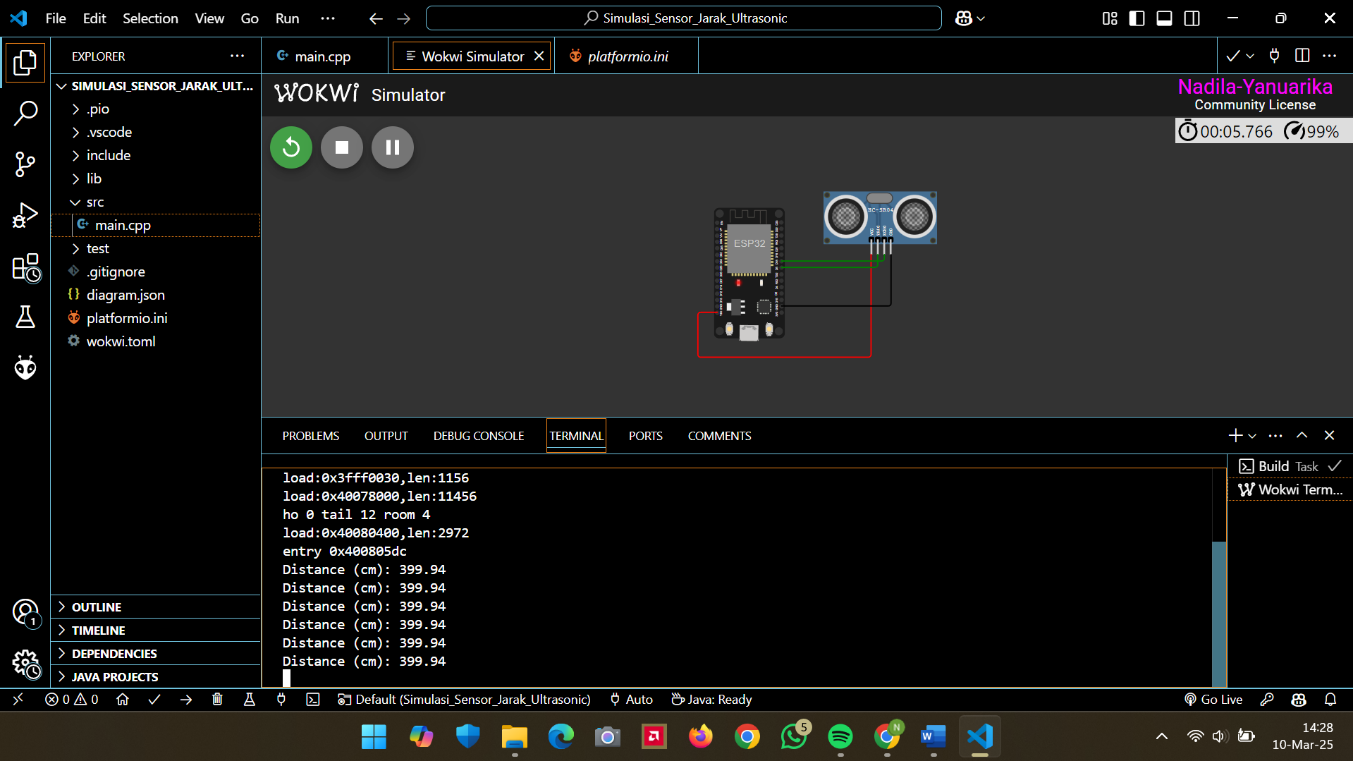
1. Terakhir simulasi sensor relay, button, dan LED akan berjalan sesuai koding dan diagram yang dirangkai.



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

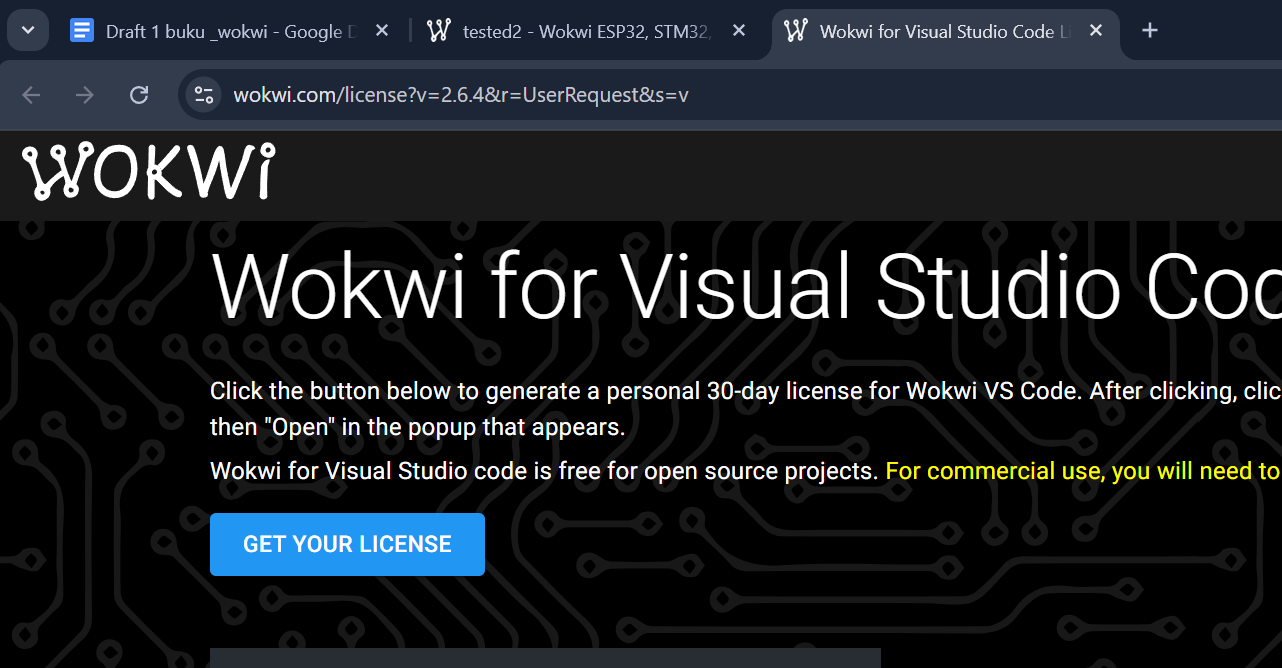
**3.1 Experimental Results (Hasil Eksperimen)**

Sensor suhu kelembaban berfungsi



**4. Appendix (Lampiran)**

Halaman Get Your License



LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)

Fakultas Vokasi, Universitas Brawijaya

**Praktik Pembuatan API   
Menggunakan Laravel 11 dan Ngrok**

*Nadila Yanuarika Rimawati*

*Fakultas Vokasi, Universitas Brawijaya*

*Email:* [*nadiladila751@student.ub.ac.id*](mailto:nadiladila751@student.ub.ac.id)

**Abstract**

*API (Application Programming Interface)* memungkinkan komunikasi antara sistem secara efisien. *Laravel 11* menyediakan fitur lengkap untuk membangun *API* dengan mudah, sementara *Ngrok* digunakan untuk mengaksesnya secara publik tanpa konfigurasi server tambahan. Dalam praktik ini, *API* dikembangkan dengan *Laravel 11* dan diuji menggunakan *Postman*. Hasilnya menunjukkan bahwa kombinasi *Laravel 11* dan *Ngrok* dapat mempercepat proses pengembangan dan *pengujian* *API*.

**Kata Kunci:** API, Laravel 11, Ngrok, Postman, Pengujian

**1. Introduction**

* 1. **Latar belakang**

API (Application Programming Interface) berperan penting dalam menghubungkan berbagai sistem dan aplikasi. Laravel 11 sebagai framework PHP modern menyediakan fitur yang memudahkan pengembangan API, seperti routing fleksibel, middleware keamanan, dan integrasi database yang efisien. Untuk mempermudah pengujian tanpa perlu hosting, digunakan Ngrok yang memungkinkan akses API secara publik. Praktik ini bertujuan untuk memahami cara membangun API dengan Laravel 11 serta menguji fungsionalitasnya menggunakan Postman, sehingga dapat diterapkan dalam pengembangan sistem yang lebih luas.

* 1. **Tujuan eksperimen**

1. Memahami dasar-dasar pengembangan API menggunakan Laravel 11.
2. Mengimplementasikan endpoint API yang dapat mengelola data dalam format JSON.
3. Menggunakan Ngrok untuk mengekspose API secara publik tanpa konfigurasi server tambahan.
4. Melakukan pengujian API menggunakan Postman untuk memastikan fungsionalitasnya.
5. Mempelajari konsep keamanan dan struktur yang baik dalam pengembangan API.

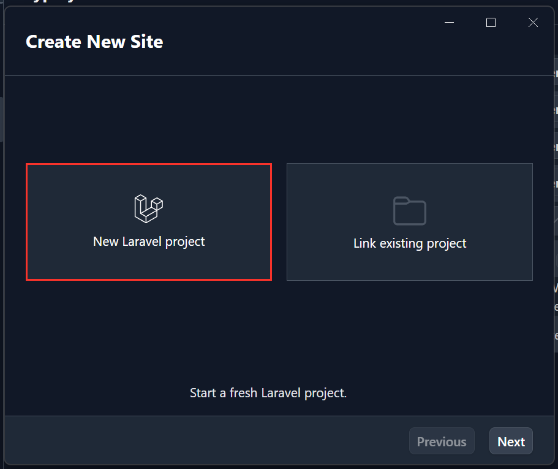
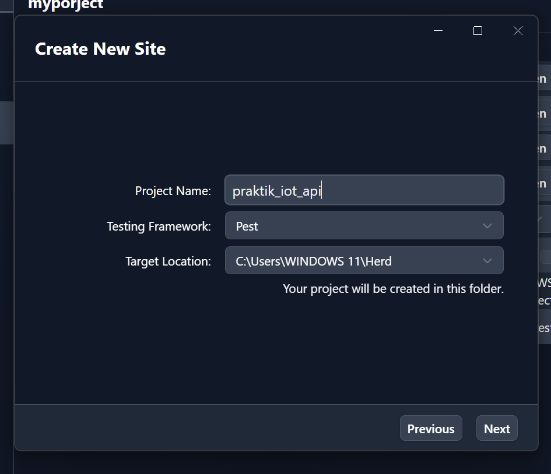
**2. Methodology**

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

Laptop, VS Code, XAMPP, phpMyAdmin, Herd, Laravel 11, Postman, Ngrok, dan koneksi internet

**2.2 Implementation Steps (Langkah Implementasi)**

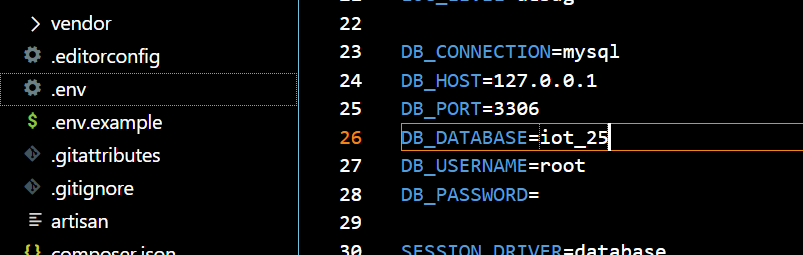
1. Membuat proyek baru laravel di herd

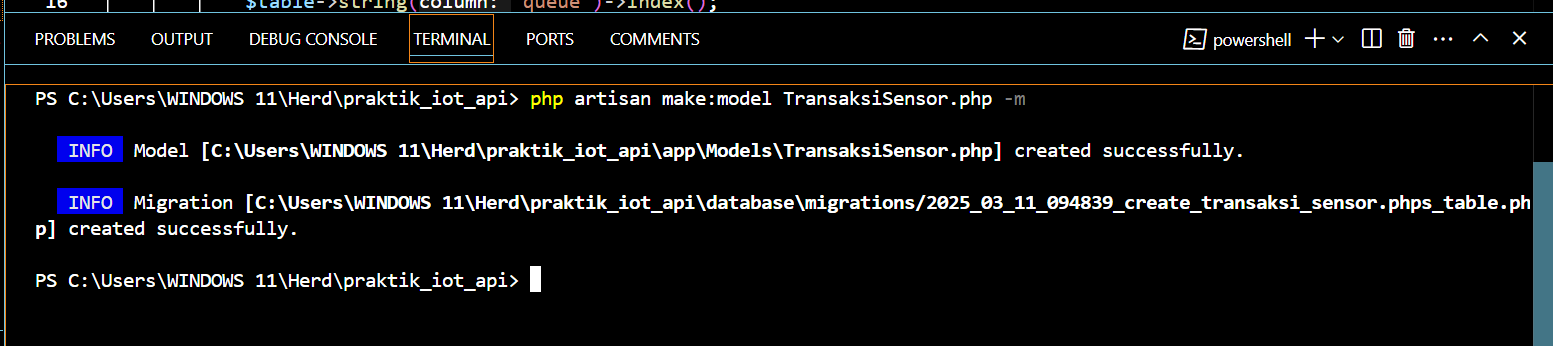
1. Buat database pada phpMyAdmin dengan nama database iot\_25



1. Edit nama database di file .env pada folder Laravel tadi



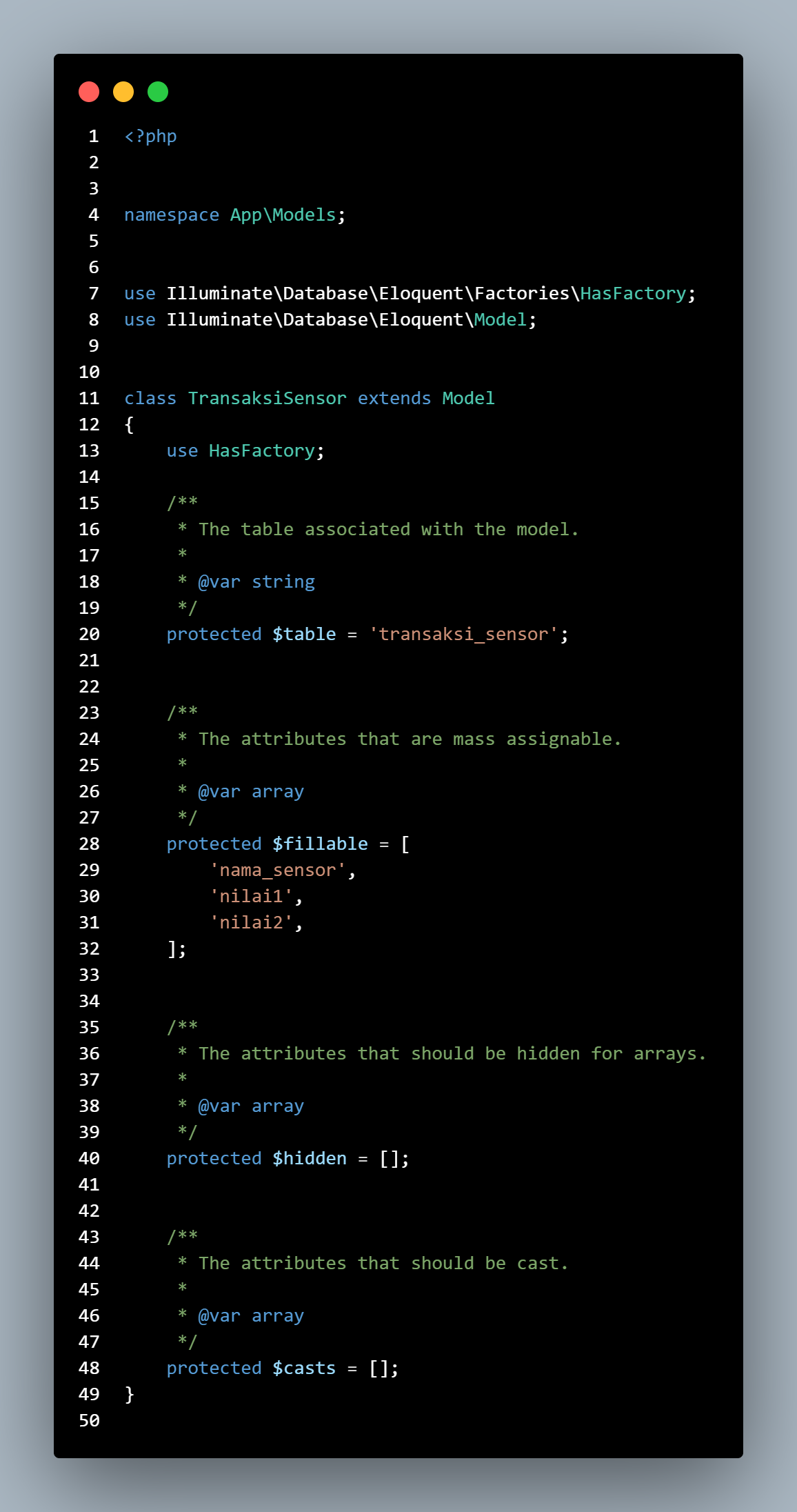
1. Buat file TransaksiSensor.php dengan perintah di terminal **php artisan make:model TransaksiSensor.php -m**

****

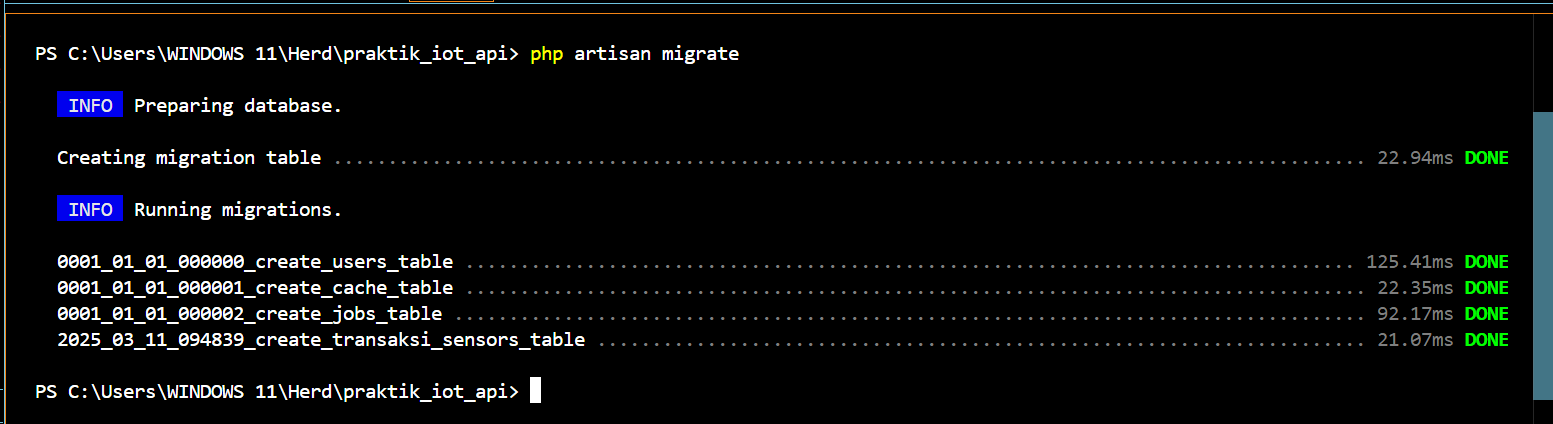
1. Ubah file database\migrations\2025\_03\_09\_13.42.16\_create\_transaksi\_sensors\_table.php dengan kode ini



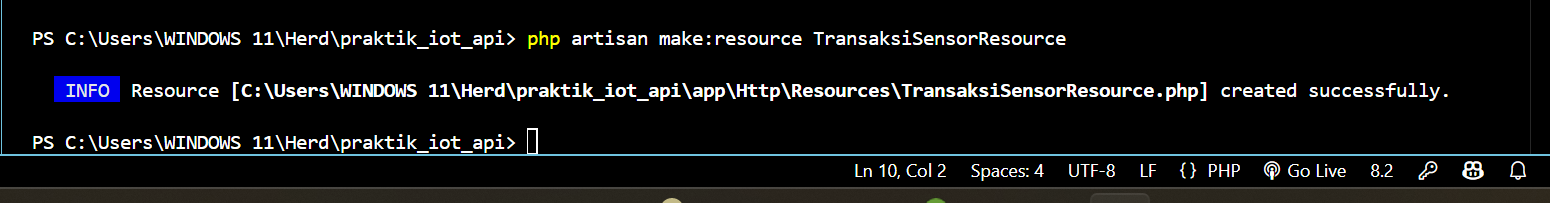
1. Ubah file app/Models/TransaksiSensor.php dengan kode ini



1. Jalankan perintah **php artisan migrate** di terminal untuk membuat tabel di database



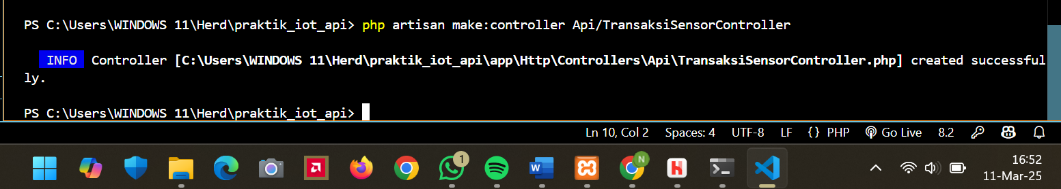
1. Buat resource dengan perintah di terminal **php artisan make:resource TransaksiSensorResource**



1. Ubah file app\Http\Resources\TransaksiSensorResource.php dengan kode ini



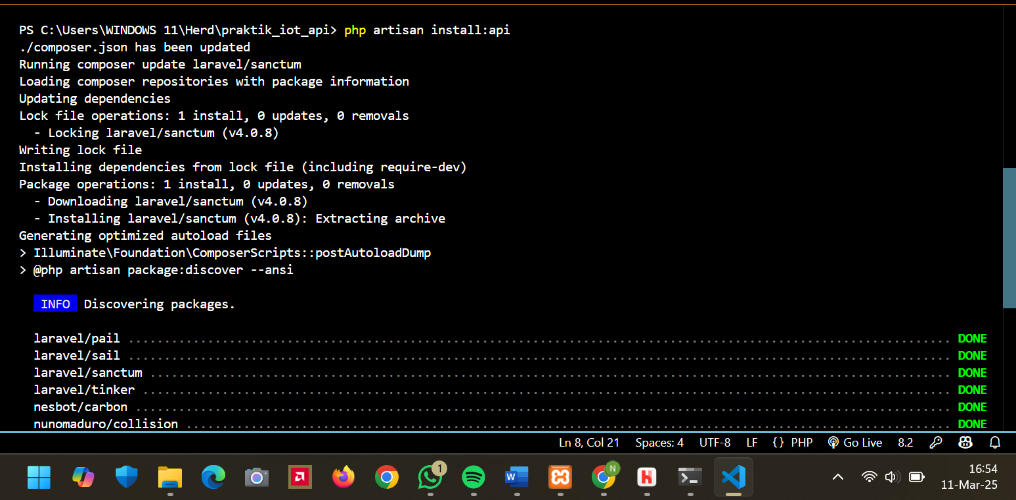
1. Buat API controller dengan perintah pada terminal**: php artisan make:controller Api/TransaksiSensorController**

****

1. Ubah isi file app/Http/Controllers/Api/TransaksiSensorController.php dengan kode ini



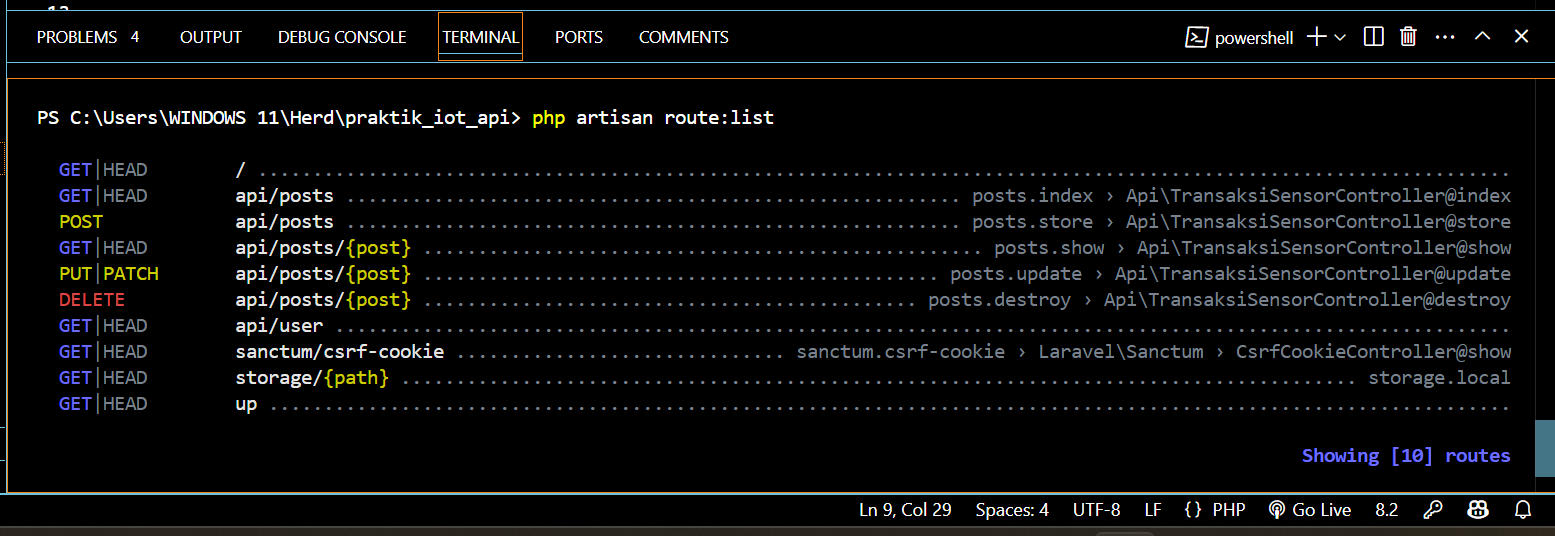
1. Membuat route khusus API dengan perintah di terminal **php artisan install:api** kemudian pilih **yes**

****

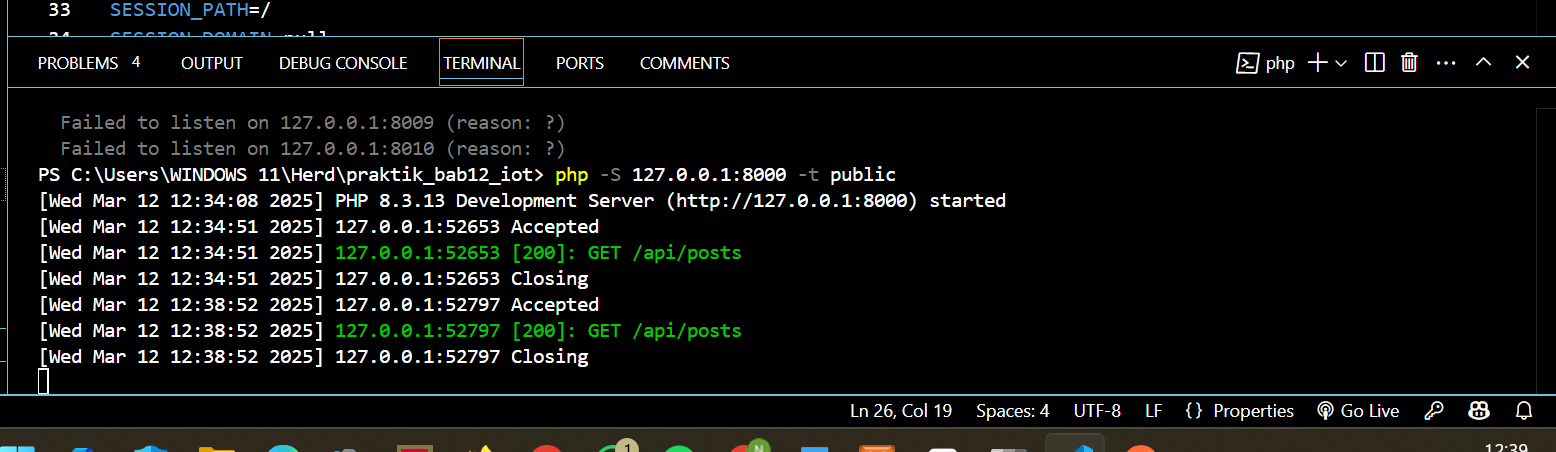
1. Ubah isi file routes/api.php dengan kode ini



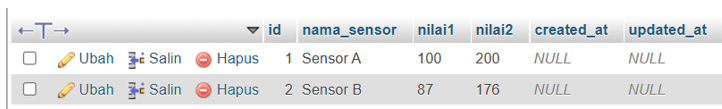
1. Masukkan perintah di terminal **php artisan route:list**, untuk memastikan route sudah terbentuk



1. Jalankan perintah **php artisan serve** atau jika tidak bisa gunakan **php -S 127.0.0.1:8000 -t public**, untuk mencoba mengakses API dan melakukan testing menggunakan tools Postman.



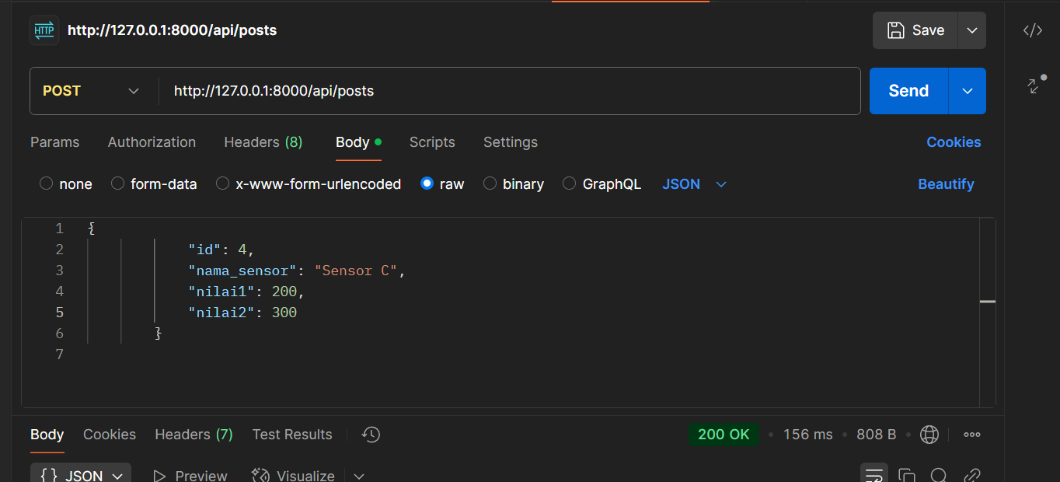
1. Pastikan ada tabel transaksi\_sensor pada database iot\_25



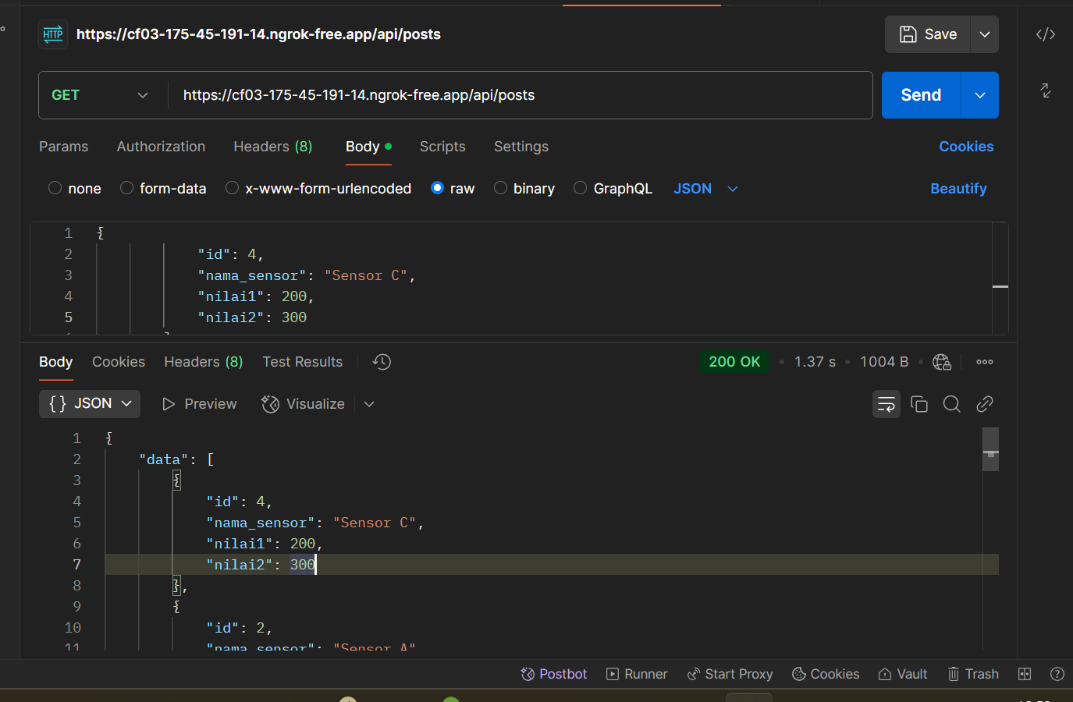
1. Pada Postman, pada bagian url, masukkan alamat server laravel dan route API-nya : http://localhost:8000/api/posts , lalu pilih method GET dan klik Send seperti gambar dibawah, lalu hasilnya akan muncul dalam format JSON. Maka API sudah berhasil mengambil data dari database.



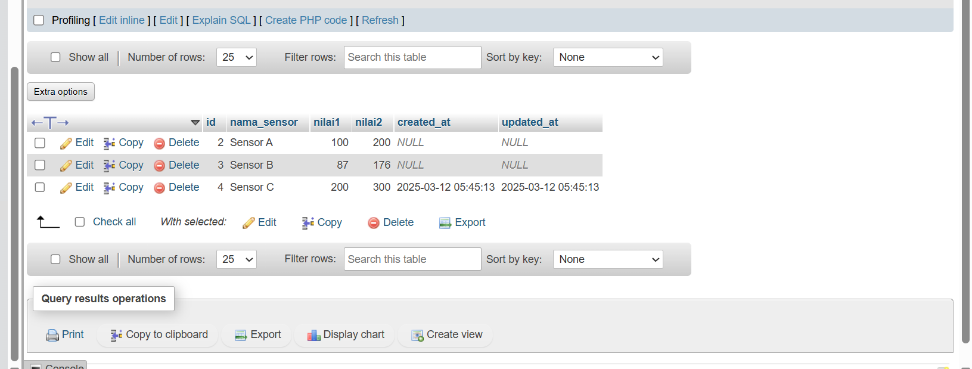
1. Lakukan percobaan menambah data di Body lalu isi dengan beda data, untuk cek Ganti method dengan POST



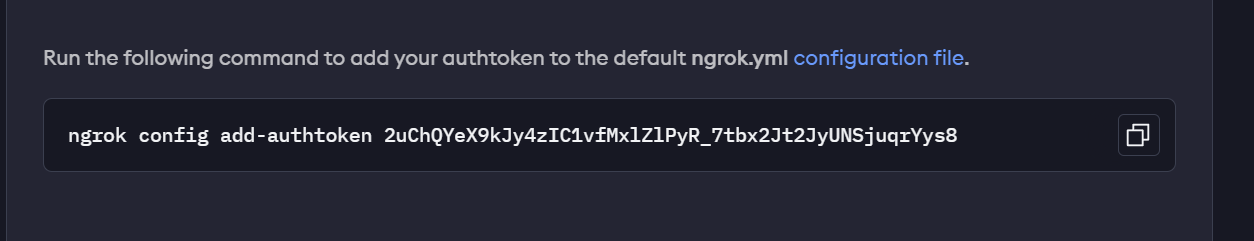
1. Lalu akan muncul ini



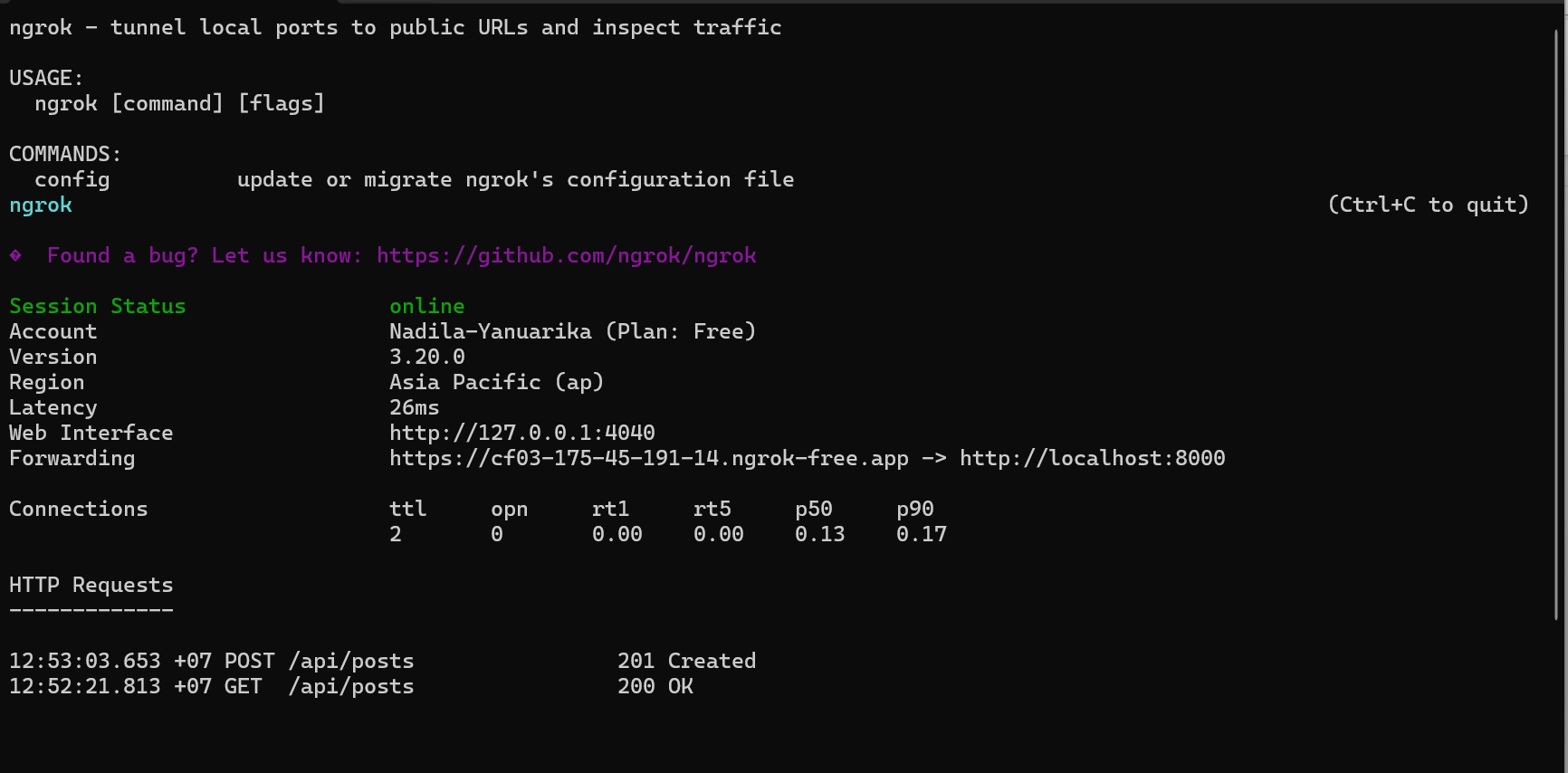
1. Lalu cek di phpMyAdmin akan otomatis bertambah



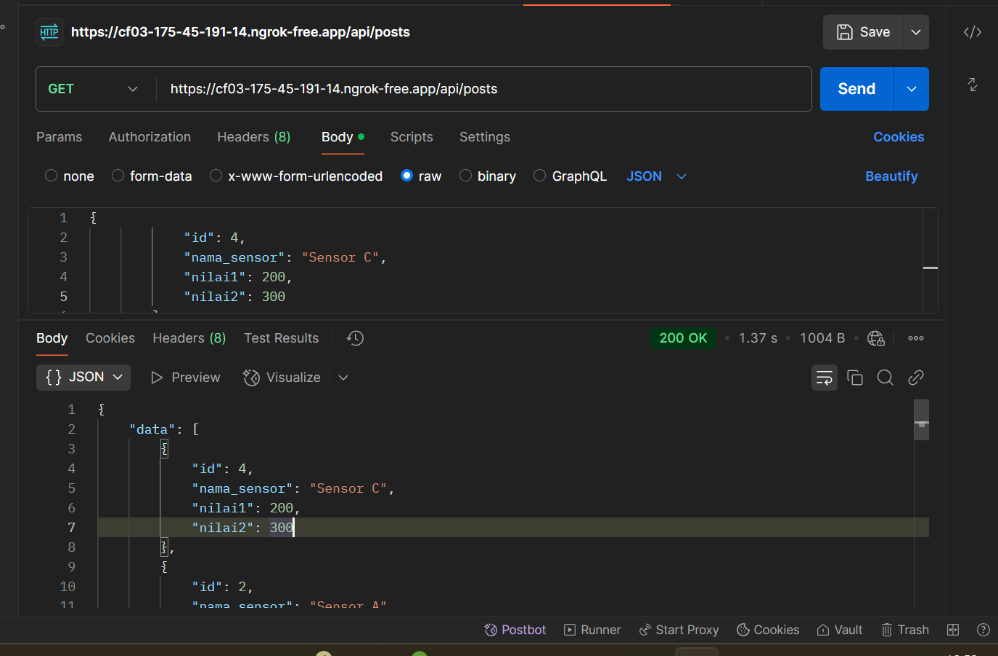
1. Download dan install Ngrok sesuai sistem operasi laptop, lalu register. Ini untuk mengonline-kan API menggunakan service Ngrok agar API dapat diakses melalui device iot atau simulasi wokwi iot. Ekstrak dan klik 2x untuk membuka cmd, lalu jalankan perintah konfigurasi yang ada pada akun Ngrok di cmd.



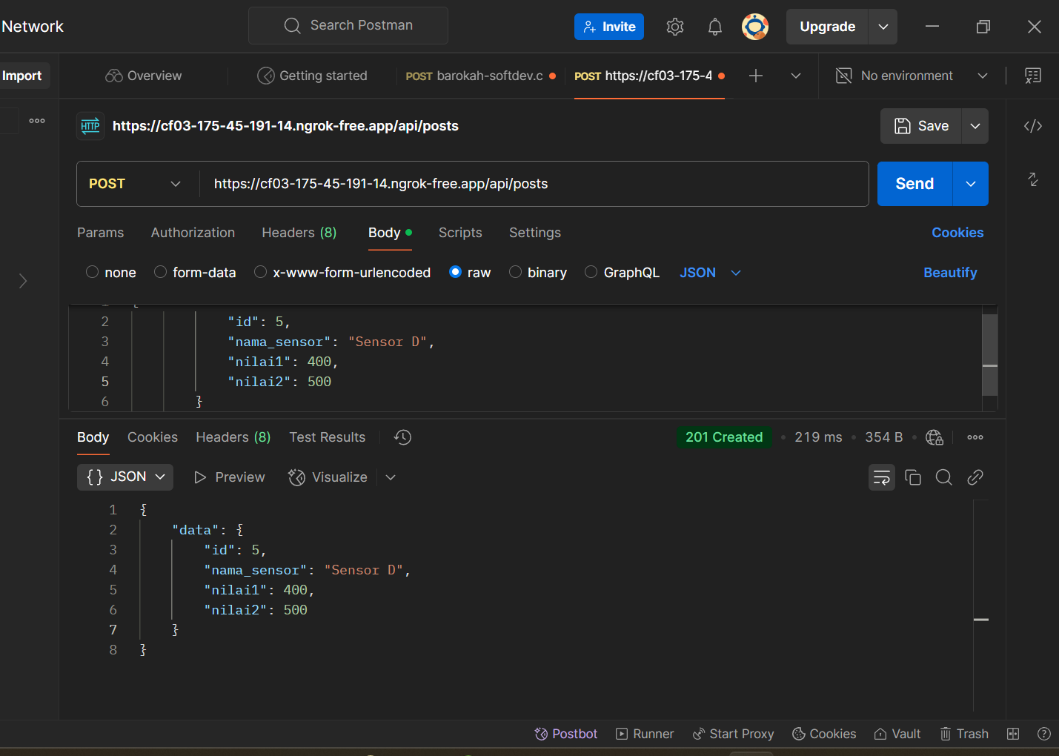
1. Kemudian menjalankan perintah ngrok http http://localhost:8000 pada cmd untuk meng-online-kan Laravel 11 melalui port 8000



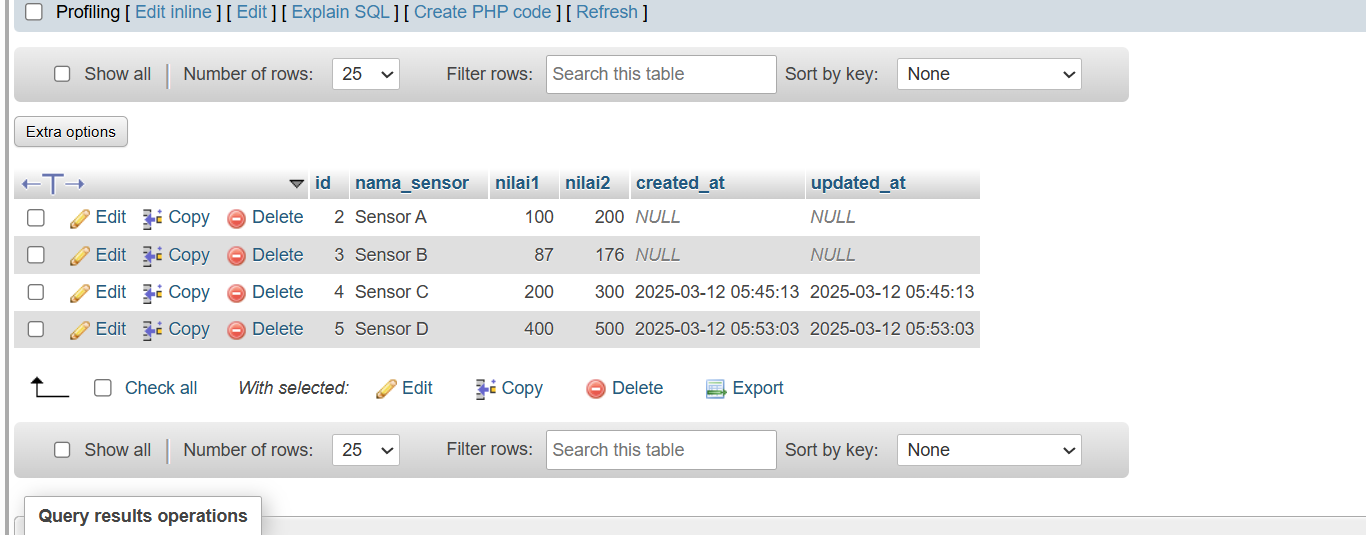
1. Melakukan percobaan untuk mengambil data dari database melalui Postman, dengan link yang telah disalin, gunakan method GET dan tambahkan route /api/posts (<https://cf03-175-45-191-14.ngrok-free.app/api/posts>)



1. Terakhir adalah melakukan perobaan untuk menambahkan data pada database, dengan menggunakan method POST, lalu pada Body, pilih raw dan format json



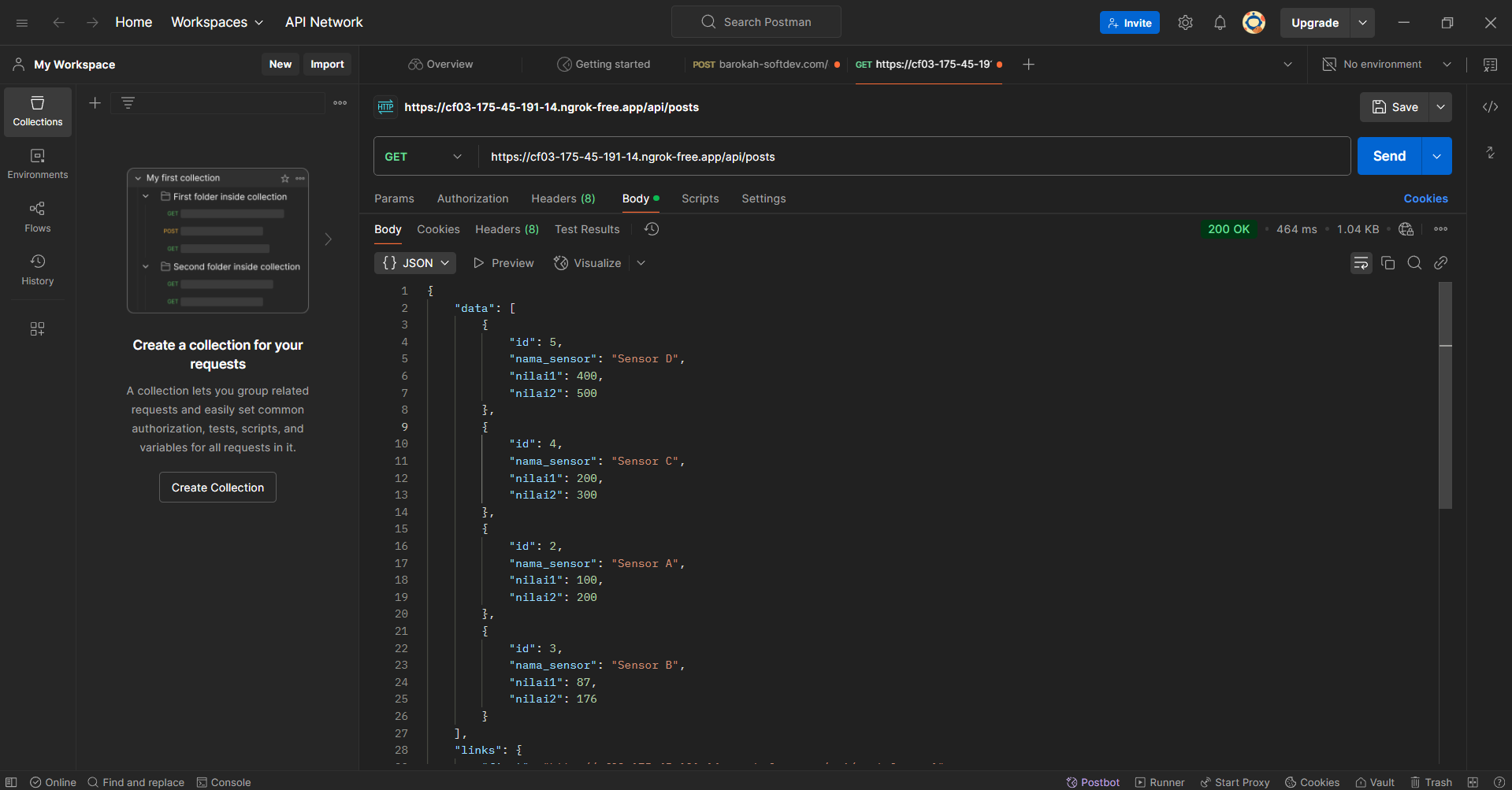
1. Cek database di phpMyAdmin

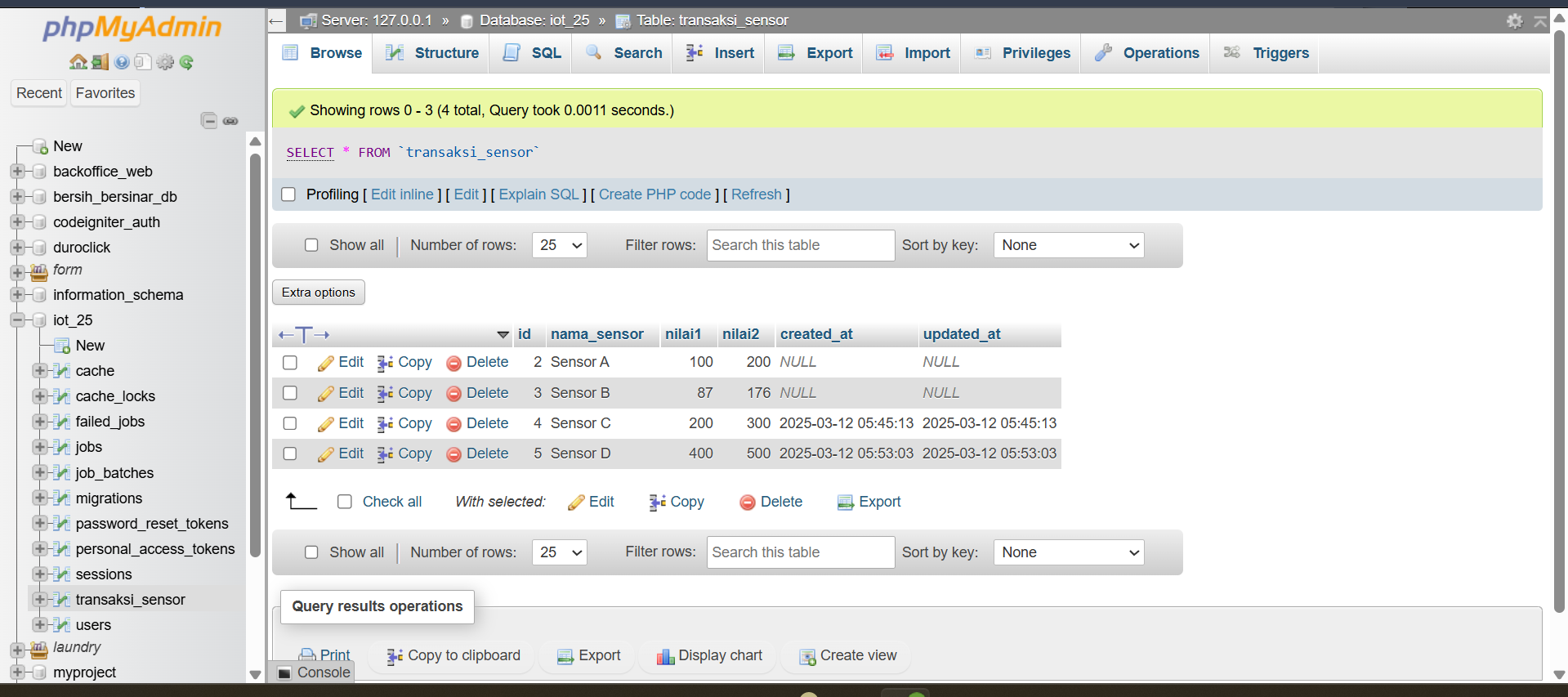


**3. Results and Discussion (Hasil dan Pembahasan)**

**3.1 Experimental Results (Hasil Eksperimen)**

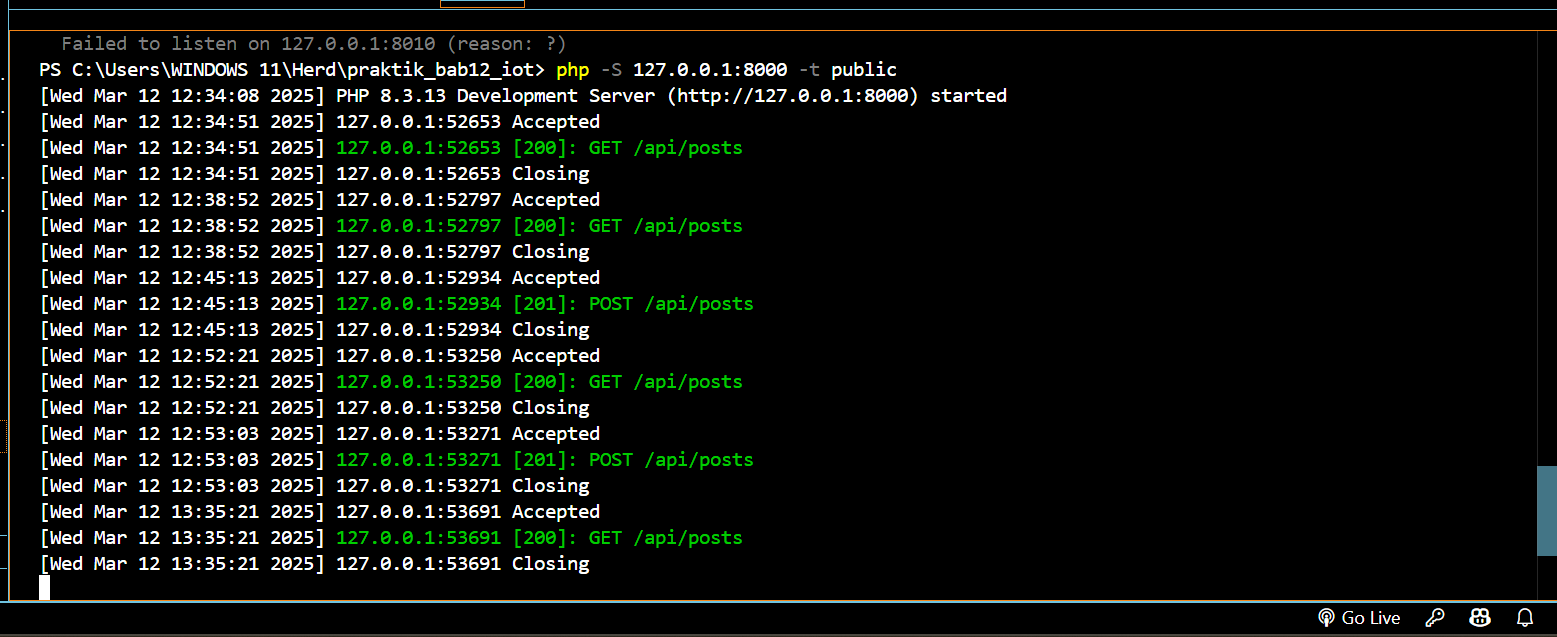
Data berhasil terhubung API dan database bisa berubah otomatis





**4. Appendix (Lampiran)**

Tampilan ketika serve aktif



Tampilan ketika http Ngrok berubah

