LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)

Fakultas Vokasi, Universitas Brawijaya

Praktik Simulasi Relay, Button, & LED

Aimar Faris Jati Wibowo

Fakultas Vokasi, Universitas Brawijaya

aimar.faris17@gmail.com

Abstract (Abstrak)

Simulasi penggunaan relay, tombol (button), dan LED merupakan bagian penting dalam pemahaman dasar sistem kontrol elektronik. Praktik ini bertujuan untuk mensimulasikan cara kerja relay sebagai saklar elektronik yang dikendalikan oleh tombol untuk menyalakan atau mematikan LED. Dalam simulasi ini, Wokwi digunakan sebagai platform untuk menguji rangkaian tanpa perangkat keras fisik.

Metode yang digunakan meliputi perancangan rangkaian, pemrograman mikrokontroler, serta analisis respon sistem terhadap input dari tombol. Hasil simulasi menunjukkan bahwa relay dapat dikendalikan secara efektif melalui tombol, yang kemudian mengontrol status LED. Praktik ini memberikan pemahaman lebih dalam mengenai konsep relay, logika kontrol sederhana, serta implementasi pada sistem otomatisasi dasar.

Keywords—Simulasi, Relay, Button, LED, Wokwi, Mikrokontroler

1. Introduction (Pendahuluan)

Dalam era digital dan otomatisasi saat ini, sistem lalu lintas yang efisien dan cerdas menjadi salah satu aspek penting dalam mengatasi kemacetan dan meningkatkan keselamatan di jalan raya. Salah satu teknologi yang dapat diterapkan untuk mendukung sistem lalu lintas modern adalah Internet of Things (IoT). Dengan pemanfaatan IoT, lampu lalu lintas dapat dikontrol dan dipantau secara real-time, memberikan solusi yang lebih adaptif terhadap kondisi lalu lintas.

Pada proyek ini, dirancang dan dikembangkan logika traffic light berbasis IoT dengan menggunakan aplikasi Wokwi sebagai simulator perangkat keras. Wokwi memungkinkan pengujian dan simulasi perangkat mikrokontroler tanpa memerlukan perangkat fisik, sehingga mempermudah proses pengembangan dan pengujian. Selain itu, GitHub digunakan sebagai wadah untuk berbagi kode sumber, mendokumentasikan perkembangan proyek, serta memfasilitasi kolaborasi dalam pengembangan sistem.

1.1 Latar belakang praktikum IoT yang dilakukan

Dalam sistem elektronik dan otomasi, relay digunakan sebagai saklar elektronik yang dapat dikendalikan dengan sinyal listrik. Penggunaan relay memungkinkan kontrol perangkat dengan arus tinggi menggunakan sinyal dari mikrokontroler atau rangkaian tegangan rendah. Selain itu, tombol (button) berfungsi sebagai input yang memungkinkan pengguna untuk mengontrol status relay, sementara LED digunakan sebagai indikator visual untuk menunjukkan status operasi sistem.

Pemahaman mengenai interaksi antara relay, tombol, dan LED sangat penting dalam pengembangan sistem kendali otomatis seperti kontrol lampu, sistem keamanan, serta peralatan rumah tangga pintar. Dengan berkembangnya teknologi, simulasi berbasis perangkat lunak seperti Wokwi memungkinkan praktisi dan pelajar untuk menguji dan menganalisis rangkaian elektronik tanpa memerlukan perangkat keras fisik.

1.2 Tujuan eksperimen

Eksperimen ini bertujuan untuk:

⦁ Mensimulasikan cara kerja relay dalam mengontrol LED dengan tombol sebagai pemicu.

⦁ Memahami proses pengendalian relay menggunakan mikrokontroler dalam simulasi.

⦁ Menganalisis kinerja sistem berbasis simulasi untuk memahami konsep dasar otomasi sederhana.

2. Methodology (Metodologi)

2.1 Tools & Materials (Alat dan Bahan)

Adapun perangkat yang digunakan dalam praktikum ini adalah sebagai berikut:

⦁ Arduino Uno

⦁ Relay 5V

⦁ LED (Light Emitting Diode)

⦁ Resistor 220Ω dan 10kΩ

⦁ Button (Push Button)

⦁ Breadboard

⦁ Kabel jumper

⦁ Software Arduino IDE

2.2 Implementation Steps (Langkah Implementasi)

Adapun Langkah-langkah dalam penyusunan sistem, pengkodean, dan pengujian adalah sebagai berikut:

⦁ Proses implementasi eksperimen ini dilakukan dalam beberapa tahapan sebagai berikut:

⦁ Persiapan Rangkaian

⦁ Sambungkan LED dengan resistor 220Ω ke pin digital Arduino.

⦁ Hubungkan relay ke Arduino dan pastikan input daya sesuai dengan spesifikasi.

⦁ Sambungkan button dengan pull-up resistor 10kΩ agar dapat membaca input dengan stabil.

⦁ Pemrograman Arduino

⦁ Tulis kode untuk membaca input dari button.

⦁ Implementasikan logika untuk menyalakan dan mematikan relay berdasarkan input button.

⦁ Tambahkan LED sebagai indikator status relay.

⦁ Pengujian dan Analisis

⦁ Upload program ke Arduino dan jalankan simulasi.

⦁ Amati perubahan status relay dan LED saat button ditekan.

⦁ Evaluasi respons sistem dan pastikan fungsinya sesuai dengan ekspektasi.

3. Results and Discussion (Hasil dan Pembahasan)

3.1 Experimental Results (Hasil Eksperimen)

Setelah menjalankan simulasi, hasil yang diperoleh menunjukkan bahwa:

⦁ Relay dapat dikendalikan dengan input dari button menggunakan Arduino.

⦁ LED menyala dan mati sesuai dengan kondisi relay, menandakan sistem berjalan dengan baik.

⦁ Pull-up resistor membantu memastikan pembacaan input button yang stabil tanpa noise.

⦁ Simulasi ini memberikan pemahaman dasar tentang integrasi relay, button, dan LED dalam sistem kendali berbasis mikrokontroler.

Kode Program:

#include <Arduino.h>

// Define pin numbers

const int ButtonPin = 19; // GPIO19 connected to the pushbutton

const int LedPin = 18; // GPIO18 connected to the LED

const int RelayPin = 23; // GPIO23 connected to the relay module

void setup() {

// Set pin modes

pinMode(ButtonPin, INPUT\_PULLUP); // Set the button pin as an input with an internal pull-up resistor

pinMode(LedPin, OUTPUT); // Set the LED pin as an output

pinMode(RelayPin, OUTPUT); // Set the relay pin as an output

// Initialize the outputs to be OFF

digitalWrite(LedPin, LOW);

digitalWrite(RelayPin, LOW);

}

void loop() {

// Read the state of the button

int buttonState = digitalRead(ButtonPin);

// Check if the button is pressed

// Since the button is wired to pull the pin LOW when pressed, we check for LOW

if (buttonState == LOW) {

digitalWrite(LedPin, HIGH); // Turn on the LED

digitalWrite(RelayPin, HIGH); // Turn on the relay

} else {

digitalWrite(LedPin, LOW); // Turn off the LED

digitalWrite(RelayPin, LOW); // Turn off the relay

}

}

Diagrma Skematik Wokwi :

{

"version": 1,

"author": "Aimar",

"editor": "wokwi",

"parts": [

{ "type": "board-esp32-devkit-c-v4", "id": "esp", "top": 9.6, "left": -81.56, "attrs": {} },

{ "type": "wokwi-led", "id": "led1", "top": 130.8, "left": 90.2, "attrs": { "color": "red" } },

{

"type": "wokwi-pushbutton",

"id": "btn1",

"top": 73.4,

"left": 144,

"attrs": { "color": "green", "xray": "1" }

},

{ "type": "wokwi-relay-module", "id": "relay1", "top": -38.2, "left": 115.2, "attrs": {} }

],

"connections": [

[ "esp:TX", "$serialMonitor:RX", "", [] ],

[ "esp:RX", "$serialMonitor:TX", "", [] ],

[ "relay1:VCC", "esp:3V3", "red", [ "h-220.8", "v57.6" ] ],

[ "relay1:GND", "esp:GND.2", "black", [ "h-96", "v47.6" ] ],

[ "relay1:IN", "esp:23", "green", [ "h-57.6", "v47.8" ] ],

[ "btn1:2.l", "esp:19", "green", [ "h0" ] ],

[ "btn1:1.l", "esp:GND.2", "green", [ "h-96", "v-48" ] ],

[ "led1:A", "esp:18", "green", [ "v0", "h9.6", "v-57.6" ] ],

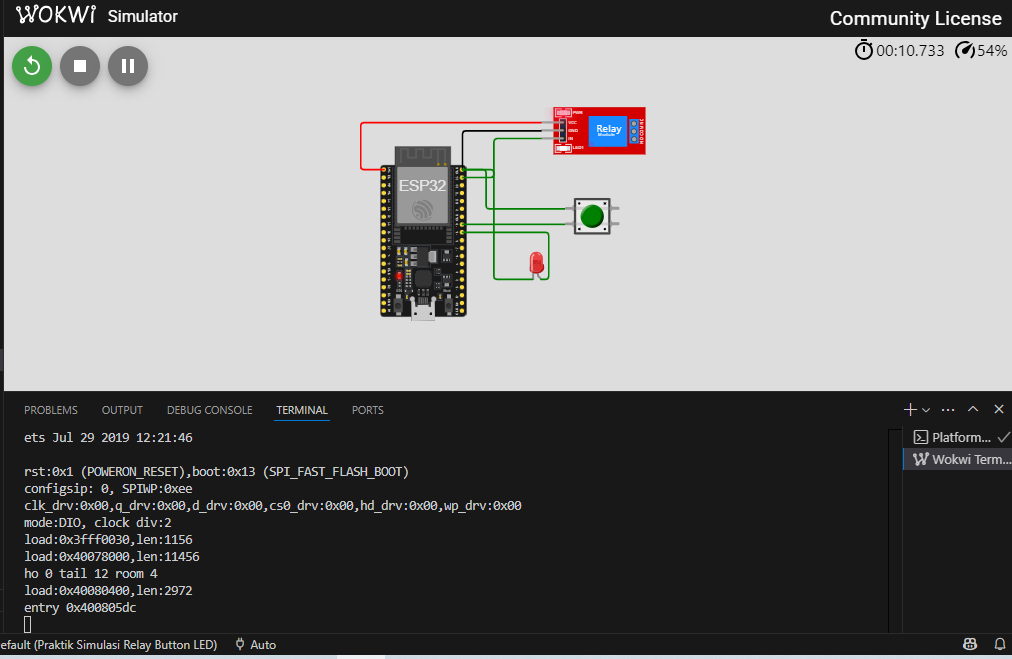
[ "led1:C", "esp:GND.2", "green", [ "v0", "h-47.6", "v-134.4" ] ]

],

"dependencies": {}

}

Gambar Simulasi Wokwi



LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)

Fakultas Vokasi, Universitas Brawijaya

Praktik Simulasi Sensor Jarak

(Ultrasonic)

Aimar Faris Jati Wibowo

Fakultas Vokasi, Universitas Brawijaya

aimar.faris17@gmail.com

Abstract (Abstrak)

(Font: Times New Roman, 10pt, Justified, Maksimal 250 Kata)

Tuliskan ringkasan eksperimen yang dilakukan, hasil utama, dan kesimpulan singkat.

Contoh:

Praktik simulasi sensor jarak ultrasonik bertujuan untuk memahami prinsip kerja sensor ultrasonik dalam mengukur jarak suatu objek serta implementasinya dalam berbagai aplikasi. Sensor ultrasonik bekerja dengan mengirimkan gelombang ultrasonik dan mengukur waktu yang dibutuhkan gelombang untuk dipantulkan kembali setelah mengenai objek. Dalam praktik ini, simulasi dilakukan menggunakan perangkat lunak yang memungkinkan visualisasi pengukuran jarak secara real-time. Hasil dari simulasi ini menunjukkan bahwa sensor ultrasonik memiliki tingkat akurasi yang tinggi dalam mendeteksi jarak dengan variasi faktor seperti sudut pantulan dan jenis permukaan objek. Praktik ini memberikan wawasan mendalam mengenai pemanfaatan sensor ultrasonik dalam bidang otomasi, robotika, dan sistem keamanan.

Keywords—sensor ultrasonik, pengukuran jarak, simulasi, otomasi, robotika.

1. Introduction (Pendahuluan)

Sensor ultrasonik merupakan salah satu jenis sensor yang banyak digunakan untuk mengukur jarak suatu objek tanpa kontak fisik. Sensor ini bekerja dengan mengirimkan gelombang ultrasonik melalui pemancar (transmitter) dan menerima pantulan gelombang tersebut melalui penerima (receiver). Waktu tempuh gelombang dari pemancar ke objek dan kembali ke penerima digunakan untuk menghitung jarak objek dengan sensor.

Dalam berbagai aplikasi, sensor ultrasonik digunakan dalam sistem otomasi, robotika, kendaraan otonom, serta perangkat keamanan seperti sistem parkir otomatis dan deteksi hambatan. Keunggulan sensor ini adalah kemampuannya dalam mendeteksi objek dengan akurasi tinggi tanpa terpengaruh oleh warna atau pencahayaan lingkungan.

Praktik simulasi sensor jarak ultrasonik bertujuan untuk memahami cara kerja sensor ini secara lebih mendalam serta menguji efektivitasnya dalam berbagai kondisi. Dengan menggunakan simulasi, peserta dapat mengamati respons sensor terhadap objek dengan jarak dan karakteristik permukaan yang berbeda, sehingga dapat mengidentifikasi faktor-faktor yang mempengaruhi kinerja sensor.

Melalui praktik ini, diharapkan peserta memperoleh wawasan tentang implementasi sensor ultrasonik dalam berbagai bidang serta memahami cara meningkatkan efisiensi dan akurasi pengukuran jarak menggunakan sensor ini.

1.1 Latar belakang praktikum IoT yang dilakukan

Dalam dunia teknologi dan otomasi, sensor memiliki peran penting dalam berbagai aplikasi, terutama dalam pengukuran jarak dan deteksi objek. Salah satu sensor yang banyak digunakan untuk tujuan ini adalah sensor ultrasonik. Sensor ultrasonik bekerja dengan memancarkan gelombang suara berfrekuensi tinggi dan mengukur waktu yang dibutuhkan gelombang tersebut untuk dipantulkan kembali oleh objek. Teknologi ini digunakan dalam berbagai bidang seperti robotika, kendaraan otonom, sistem keamanan, serta perangkat elektronik lainnya.

Pentingnya pemahaman tentang sensor ultrasonik mendorong perlunya praktik simulasi untuk mempelajari cara kerja serta faktor-faktor yang mempengaruhi kinerjanya. Dengan simulasi, pengguna dapat mengeksplorasi berbagai kondisi operasional sensor tanpa harus menggunakan perangkat keras secara langsung. Hal ini tidak hanya menghemat biaya, tetapi juga mempermudah dalam melakukan analisis dan pengujian sebelum implementasi di dunia nyata.

Selain itu, pemahaman yang baik tentang sensor ultrasonik sangat bermanfaat dalam pengembangan teknologi berbasis sensor, seperti sistem parkir otomatis, drone navigasi, dan alat bantu bagi penyandang disabilitas. Oleh karena itu, melalui praktik simulasi ini, diharapkan peserta dapat memahami prinsip kerja sensor ultrasonik, cara penggunaannya, serta tantangan yang mungkin dihadapi dalam implementasinya.

1.2 Tujuan eksperimen

Eksperimen ini bertujuan untuk:

⦁ Memahami Prinsip Kerja Sensor Ultrasonik

⦁ Mengidentifikasi Faktor yang Mempengaruhi Kinerja Sensor

⦁ Mensimulasikan Penggunaan Sensor Ultrasonik dalam Berbagai Kondisi

⦁ Mengembangkan Pemahaman tentang Aplikasi Sensor Ultrasonik

⦁ Meningkatkan Kemampuan dalam Menggunakan Perangkat Simulasi

2. Methodology (Metodologi)

Metodologi yang digunakan dalam eksperimen ini terdiri dari beberapa tahapan utama, yaitu:

1. Persiapan

- Menentukan perangkat lunak simulasi yang akan digunakan, seperti Proteus, Tinkercad, atau MATLAB.

- Menyusun rangkaian simulasi dengan memasukkan komponen sensor ultrasonik (misalnya HC-SR04), mikrokontroler (Arduino atau sejenisnya), dan tampilan hasil pengukuran (LCD atau serial monitor).

2. Implementasi Simulasi

- Mengonfigurasi sensor ultrasonik dalam perangkat simulasi.

- Menuliskan kode pemrograman untuk membaca data jarak dari sensor dan menampilkannya pada monitor atau layar simulasi.

3. Pengumpulan Data

- Merekam hasil pengukuran sensor dalam berbagai kondisi.

- Menganalisis keakuratan dan konsistensi data yang diperoleh dari simulasi.

4. Analisis dan Evaluasi

- Membandingkan hasil pengukuran simulasi dengan teori yang ada mengenai sensor ultrasonik.

- Mengevaluasi kesalahan pengukuran dan penyebabnya.

2.1 Tools & Materials (Alat dan Bahan)

Adapun perangkat yang digunakan dalam praktikum ini adalah sebagai berikut:

⦁ Arduino Uno

⦁ Sensor Ultrasonic HC-SR04

⦁ Resistor (opsional)

⦁ Breadboard

⦁ Kabel jumper

⦁ Software Arduino IDE

2.2 Implementation Steps (Langkah Implementasi)

Langkah-langkah implementasi eksperimen ini adalah sebagai berikut:

Proses implementasi eksperimen ini dilakukan dalam beberapa tahapan sebagai berikut:

⦁ Persiapan Rangkaian

⦁ Hubungkan trigPin sensor ke pin digital 5 pada Arduino.

⦁ Hubungkan echoPin sensor ke pin digital 18 pada Arduino.

⦁ Pastikan koneksi VCC dan GND sensor tersambung dengan benar.

⦁ Pemrograman Arduino

⦁ Konfigurasi trigPin sebagai OUTPUT dan echoPin sebagai INPUT.

⦁ Kirim sinyal trigger selama 10 mikrodetik untuk memancarkan gelombang ultrasonik.

⦁ Hitung waktu pantulan gelombang dengan fungsi pulseIn() untuk mendapatkan jarak objek.

⦁ Konversi hasil pengukuran ke satuan cm dan tampilkan di Serial Monitor.

⦁ Pengujian dan Analisis

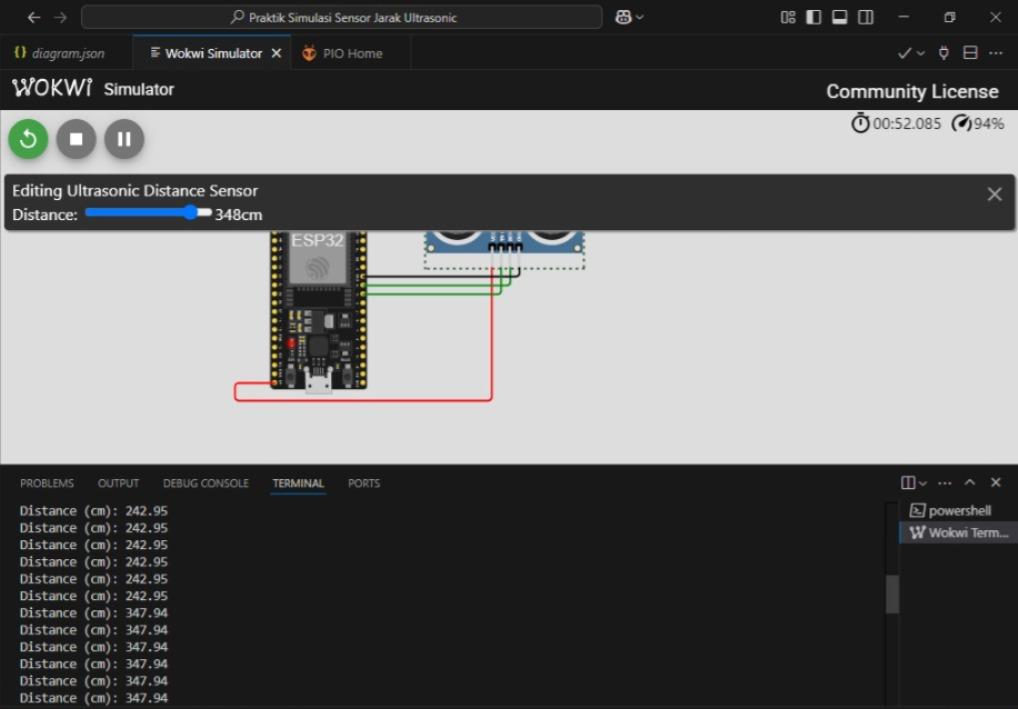
⦁ Upload program ke Arduino dan jalankan simulasi.

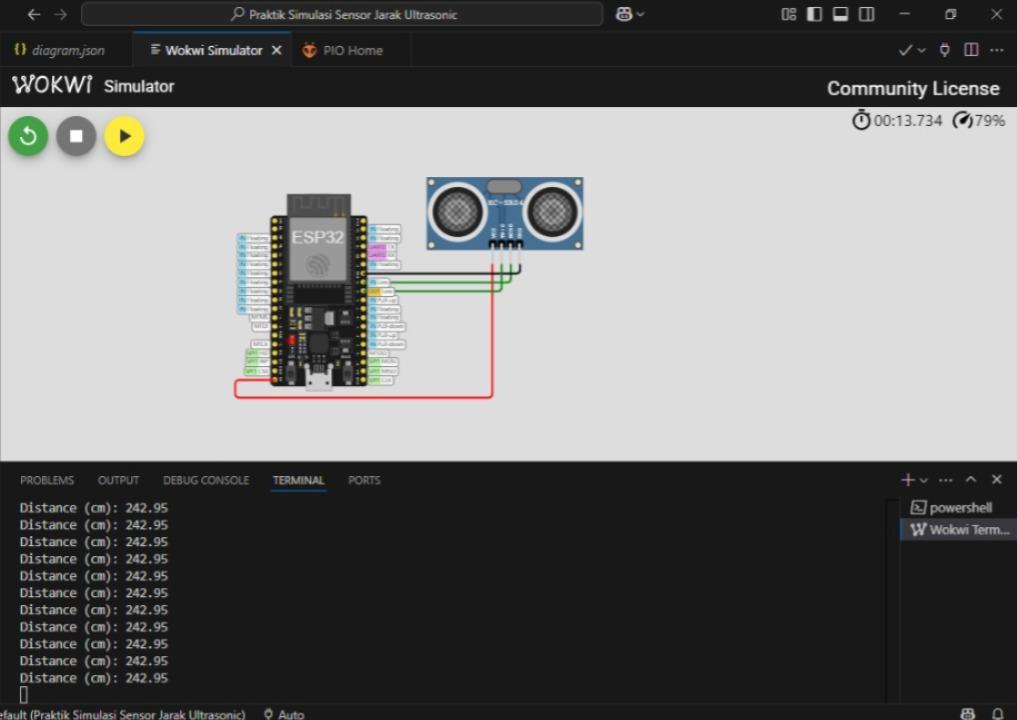
⦁ Letakkan objek di depan sensor dan amati perubahan jarak yang ditampilkan.

⦁ Bandingkan hasil pengukuran dengan pengukuran manual untuk melihat akurasi sensor.

3. Results and Discussion (Hasil dan Pembahasan)

3.1 Experimental Results (Hasil Eksperimen)





Kode Program

#include <Arduino.h>

const int trigPin = 18;

const int echoPin = 19;

//define sound speed in cm/uS

#define SOUND\_SPEED 0.034

#define CM\_TO\_INCH 0.393701

long duration;

float distanceCm;

float distanceInch;

void setup() {

Serial.begin(115200); // Starts the serial communication

pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output

pinMode(echoPin, INPUT); // Sets the echoPin as an Input

}

void loop() {

// Clears the trigPin

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

// Sets the trigPin on HIGH state for 10 micro seconds

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

// Reads the echoPin, returns the sound wave travel time in microseconds

duration = pulseIn(echoPin, HIGH);

// Calculate the distance

distanceCm = duration \* SOUND\_SPEED/2;

// Convert to inches

distanceInch = distanceCm \* CM\_TO\_INCH;

// Prints the distance in the Serial Monitor

Serial.print("Distance (cm): ");

Serial.println(distanceCm);

// Serial.print("Distance (inch): ");

// Serial.println(distanceInch);

delay(1000);

}

code Diagram.json

{

"version": 1,

"author": "Anonymous maker",

"editor": "wokwi",

"parts": [

{ "type": "board-esp32-devkit-c-v4", "id": "esp", "top": 0, "left": 0, "attrs": {} },

{

"type": "wokwi-hc-sr04",

"id": "ultrasonic1",

"top": -17.7,

"left": 168.7,

"attrs": { "distance": "320" }

}

],

"connections": [

[ "esp:TX", "$serialMonitor:RX", "", [] ],

[ "esp:RX", "$serialMonitor:TX", "", [] ],

[ "ultrasonic1:VCC", "esp:5V", "red", [ "v144", "h-278.4", "v-19.2" ] ],

[ "ultrasonic1:ECHO", "esp:19", "green", [ "v0" ] ],

[ "ultrasonic1:GND", "esp:GND.3", "black", [ "v0" ] ],

[ "ultrasonic1:TRIG", "esp:18", "green", [ "v0" ] ]

],

"dependencies": {}

}

LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)

Fakultas Vokasi, Universitas Brawijaya

Praktik Pembuatan API

Menggunakan Laravel 11 dan Ngrok

Aimar Faris Jati Wibowo

Fakultas Vokasi, Universitas Brawijaya

aimar.faris17@gmail.com

ABSTRACT

Perkembangan teknologi web semakin pesat, terutama dalam pembuatan Application Programming Interface (API) yang memungkinkan komunikasi antar sistem secara efisien. Laravel, sebagai salah satu framework PHP yang populer, menyediakan fitur bawaan untuk membangun API dengan mudah dan cepat. Dalam praktik ini, digunakan Laravel 11 untuk membuat API yang dapat diakses melalui internet menggunakan Ngrok, sebuah alat yang memungkinkan tunneling ke localhost.

Penelitian ini bertujuan untuk memahami konsep dasar API dalam Laravel 11, mengimplementasikan metode RESTful API, serta menguji konektivitas API melalui Ngrok. Tahapan yang dilakukan meliputi instalasi Laravel 11, konfigurasi database, pembuatan endpoint API menggunakan Laravel Resource Controller, serta pengujian API menggunakan Postman. Ngrok digunakan untuk menyediakan public URL yang memungkinkan API diakses dari luar jaringan lokal.

Hasil praktik menunjukkan bahwa kombinasi Laravel 11 dan Ngrok dapat digunakan untuk mengembangkan API yang cepat dan fleksibel, serta memudahkan pengujian tanpa perlu deploy ke server langsung. Dengan metode ini, pengembang dapat mempercepat proses pengujian API, terutama dalam pengembangan aplikasi berbasis microservices atau mobile.

Kata Kunci: API, Laravel 11, Ngrok, RESTful API, Web Development.

Latar Belakang

API (Application Programming Interface) menjadi elemen penting dalam pengembangan aplikasi modern karena memungkinkan pertukaran data antar sistem secara efisien. Laravel 11, sebagai framework PHP yang populer, menyediakan berbagai fitur untuk membangun RESTful API dengan mudah dan terstruktur.

Namun, dalam proses pengujian, API yang berjalan di localhost sulit diakses dari luar jaringan. Untuk mengatasi hal ini, digunakan Ngrok, yang memungkinkan API mendapatkan public URL sementara tanpa perlu deploy ke server.

Praktik ini bertujuan untuk memahami pembuatan API dengan Laravel 11 serta menguji aksesibilitasnya menggunakan Ngrok, guna mempercepat proses pengembangan dan integrasi aplikasi.

Tujuan Eksperimen

⦁ Membangun API menggunakan Laravel 11 untuk mengelola data transaksi sensor.

⦁ Menghubungkan API dengan database MySQL untuk penyimpanan dan pengambilan data sensor.

⦁ Menguji API menggunakan Postman sebelum diintegrasikan dengan perangkat IoT

⦁ Mengonline-kan API menggunakan Ngrok agar dapat diakses dari jaringan publik.

Metodologi

⦁ Alat dan Bahan

⦁ Laptop/PC dengan sistem operasi Windows/Linux/MacOS

⦁ Composer untuk instalasi Laravel

⦁ PHP versi 8.x atau lebih tinggi

⦁ Laravel 11 sebagai framework pengembangan API

⦁ Postman untuk pengujian API

⦁ XAMPP atau MySQL Server sebagai database backend

⦁ Ngrok untuk membuka akses API secara publik

⦁ Langkah Implementasi

Proses implementasi eksperimen ini dilakukan dalam beberapa tahapan sebagai berikut:

⦁ Instalasi Laravel 11

⦁ Menggunakan perintah composer create-project --prefer-dist laravel/laravel:^11.0 laravel-11

⦁ Mengatur database MySQL dengan nama iot\_25 dan mengonfigurasi file .env

⦁ Pembuatan Model dan Migrasi Database

⦁ Membuat model TransaksiSensor dengan php artisan make:model TransaksiSensor -m

⦁ Menyesuaikan skema tabel di file migrasi

⦁ Pembuatan API Controller dan Resource

⦁ Membuat API controller php artisan make:controller Api/TransaksiSensorController

⦁ Mengatur fungsi CRUD untuk mengelola data sensor

⦁ Pengujian API dengan Postman

⦁ Menjalankan server Laravel dengan php artisan serve

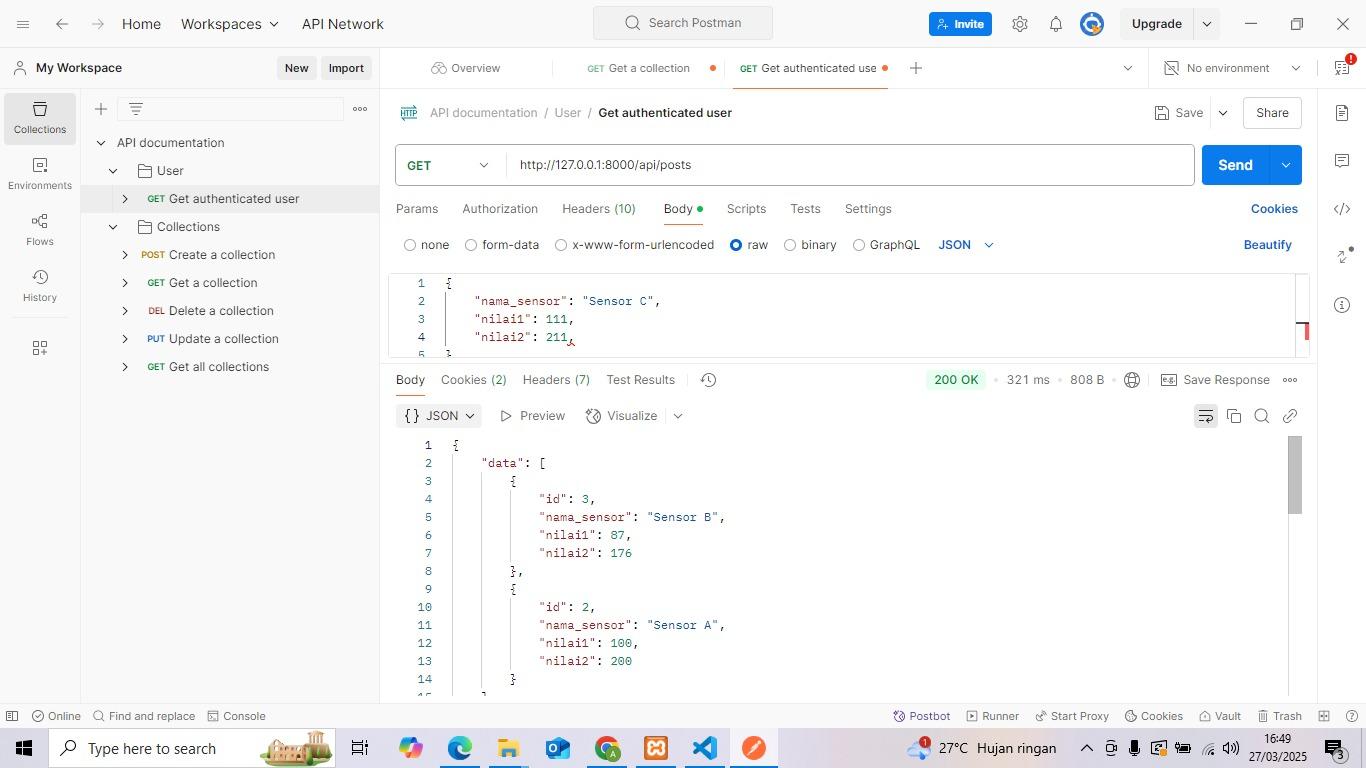
⦁ Menggunakan Postman untuk menguji endpoint GET, POST, PUT, dan DELETE

⦁ Mengonline-kan API dengan Ngrok

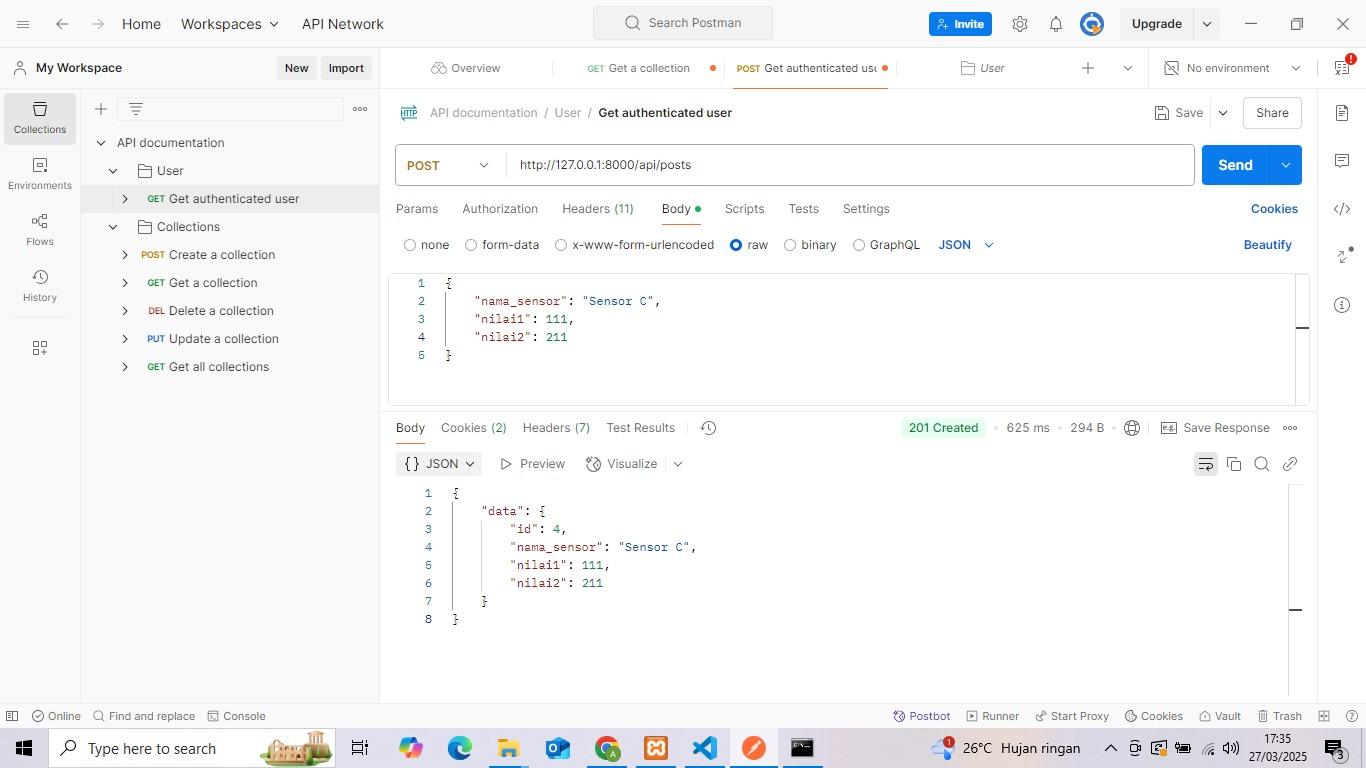
⦁ Menjalankan Ngrok menggunakan perintah ngrok http http://localhost:8000

⦁ Menggunakan URL publik yang diberikan oleh Ngrok untuk mengakses API

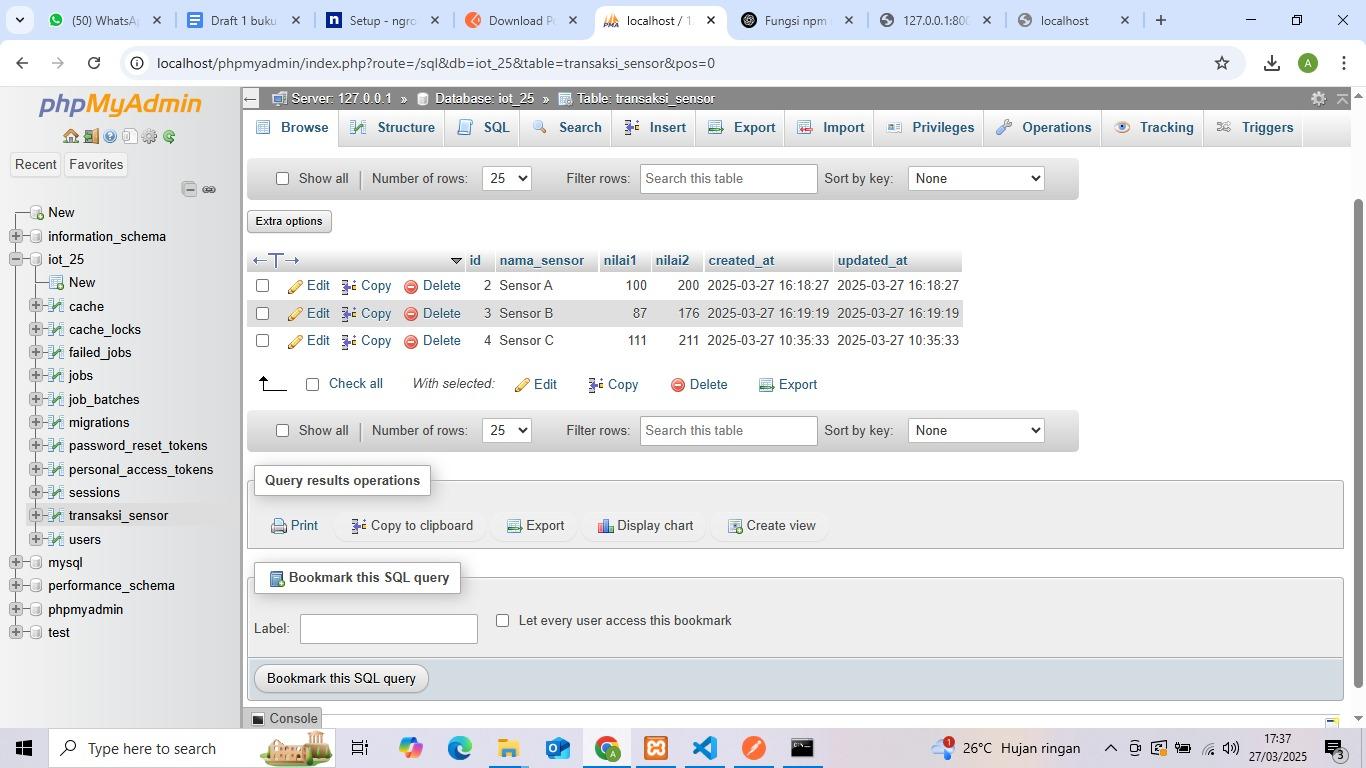
Hasil Eksperimen



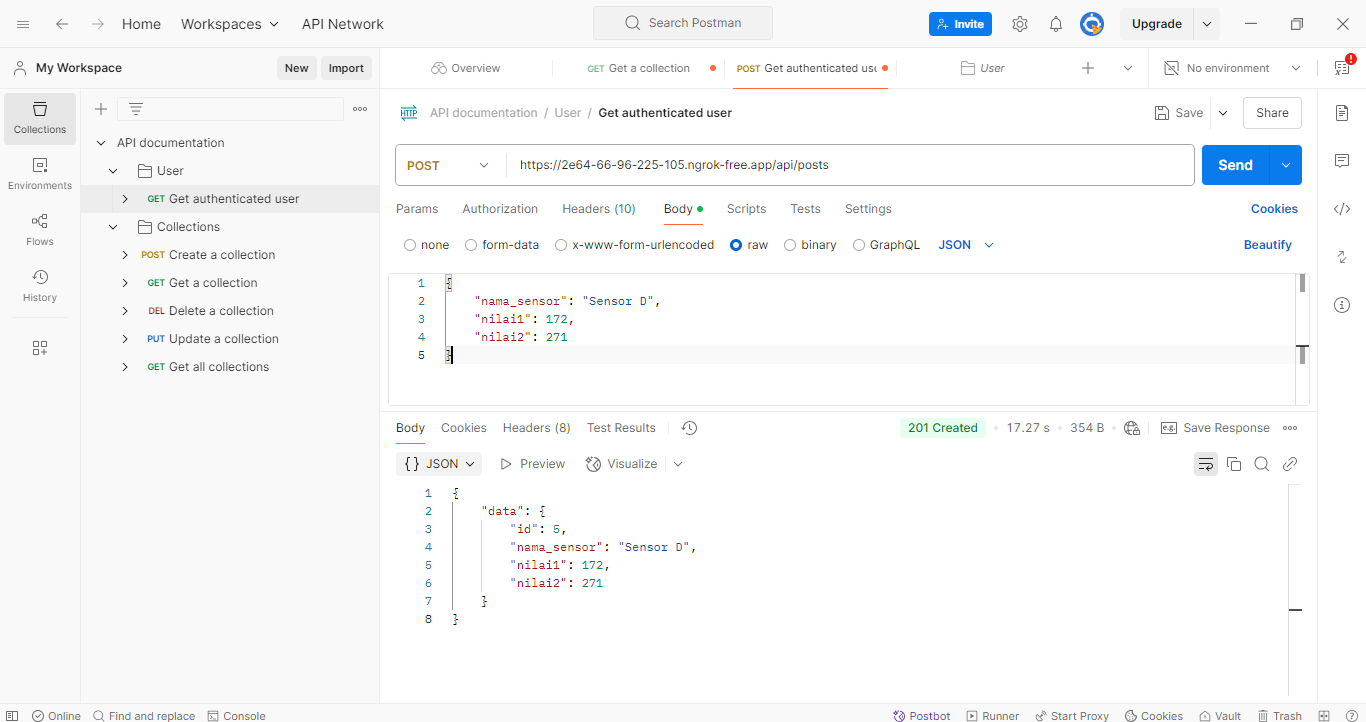
Tampilan sebelum menambahkan data pada Postman



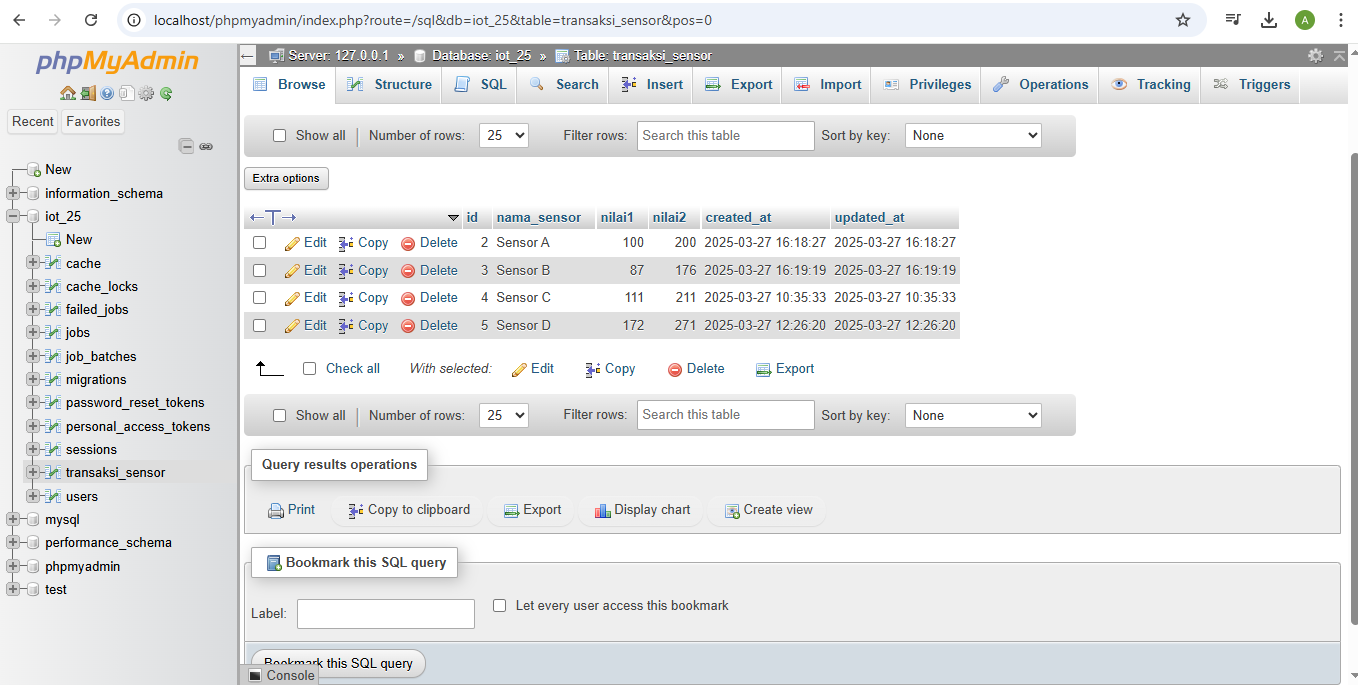
Tampilan setelah menambahkan data



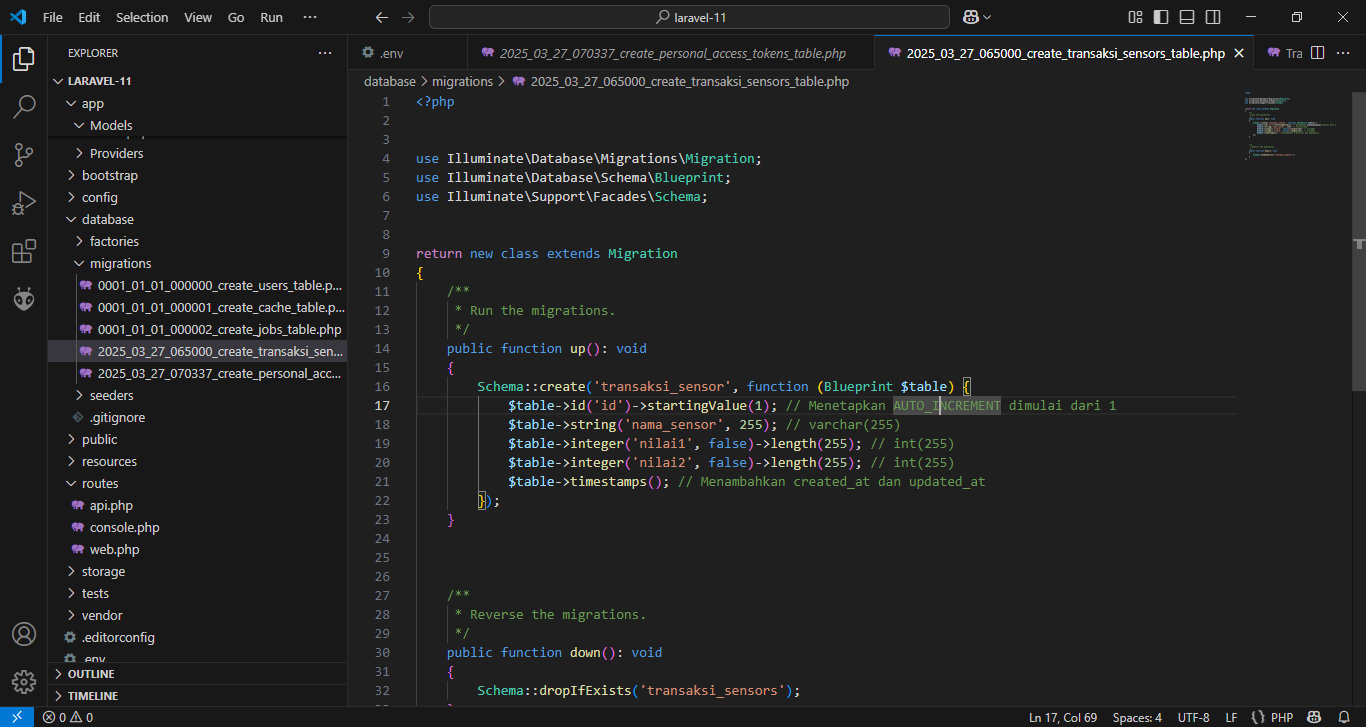
Data masuk pada Database



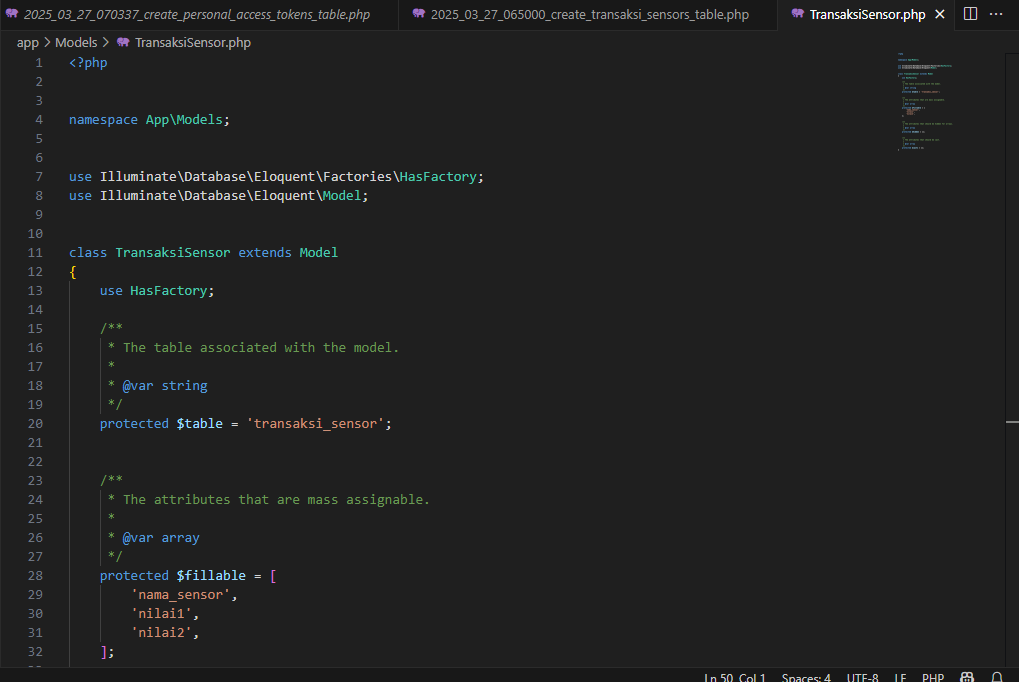
Tampilan input data setelah mengaktifkan ngrok.



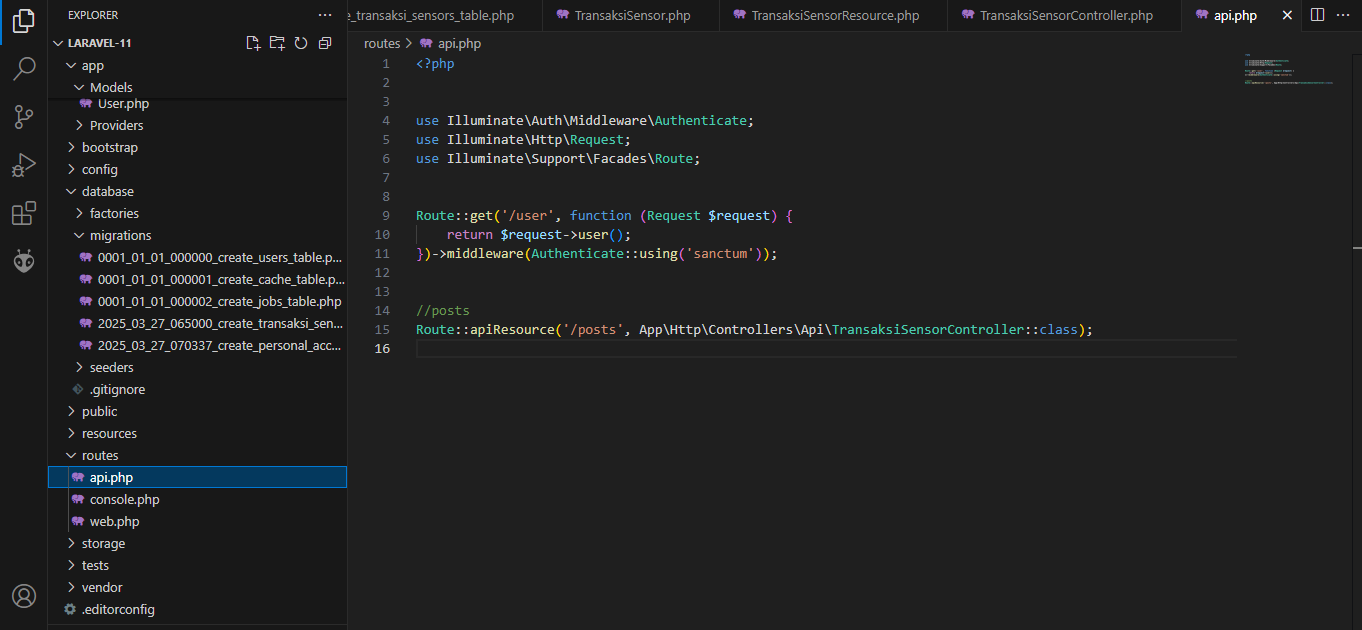
Tampilan Database Terbaru.



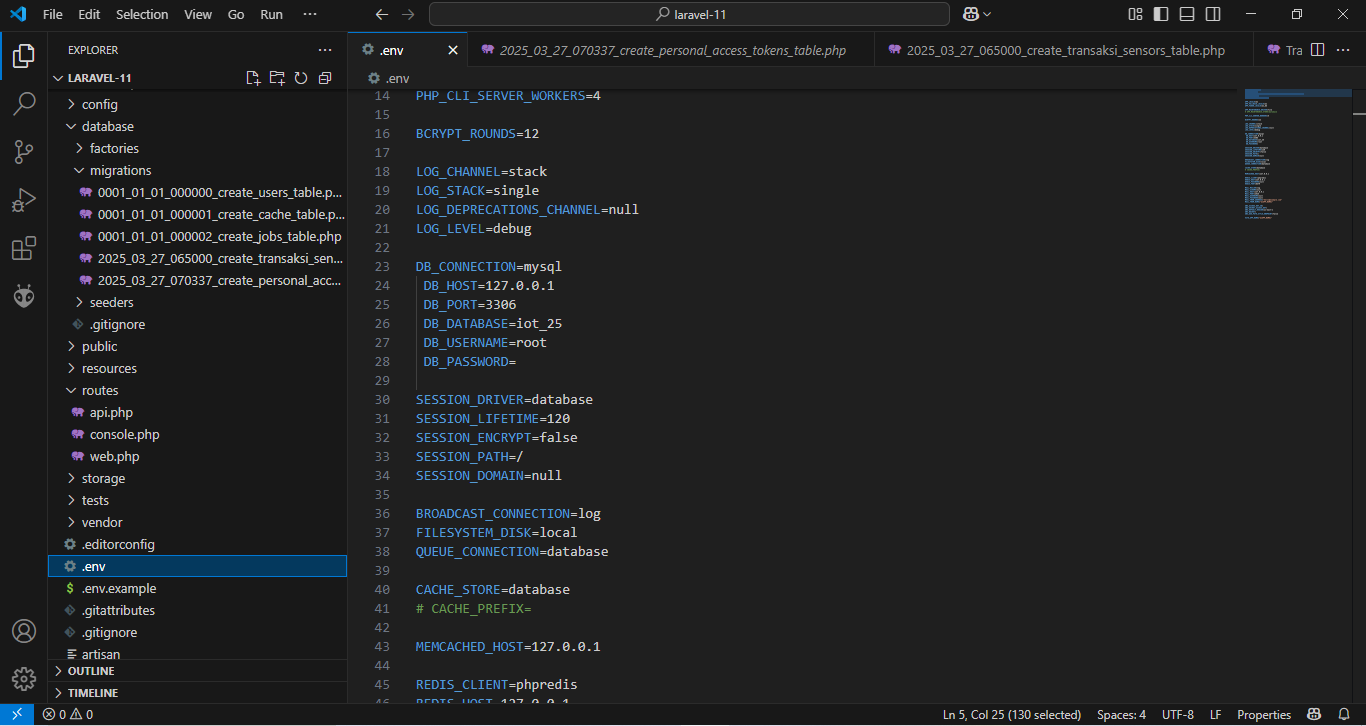
Tampilan code pada 2025\_03\_27\_065000\_create\_transaksi\_sensors\_table



Tampilan Kode pemrograman app/Models/TransaksiSensor.php



Tampilan api.php



Tampilan .env