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

**Fakultas Vokasi, Universitas Brawijaya**

**Membuat Tampilan Interface Web Dashboard IoT**

*Alvaro Zeka Ricardo*

*Fakultas Vokasi,Universitas Brawijaya*

*Email : alvarozeka7@student.ub.ac.id*

**Abstract** **:** Dalam era digital yang semakin maju, Internet of Things (IoT) memainkan peran penting dalam menghubungkan berbagai perangkat untuk memantau dan mengelola data secara real-time. Salah satu komponen krusial dalam sistem IoT adalah antarmuka (interface) yang memungkinkan pengguna untuk berinteraksi dengan data secara efisien. Penelitian atau proyek ini bertujuan untuk merancang dan membangun tampilan interface web dashboard IoT yang interaktif, responsif, dan informatif. Dashboard ini akan menampilkan data sensor secara visual dalam bentuk grafik, tabel, dan indikator lainnya guna mempermudah proses monitoring serta pengambilan keputusan oleh pengguna.

**Kata Kunci :API,Laravel,Php**

**1. Introduction**

**1.1 Latar belakang**

Perkembangan teknologi IoT telah memungkinkan integrasi antara perangkat keras dan perangkat lunak dalam berbagai sektor, seperti pertanian, industri, kesehatan, hingga rumah pintar. Data yang dihasilkan oleh perangkat IoT biasanya bersifat real-time dan dalam jumlah besar, sehingga memerlukan sistem pemantauan yang mampu menyajikan informasi secara jelas dan mudah dipahami. Oleh karena itu, keberadaan sebuah dashboard web menjadi kebutuhan yang penting untuk mendukung efektivitas sistem IoT secara keseluruhan.

Di sisi lain, tantangan dalam pembuatan dashboard web terletak pada desain antarmuka yang intuitif dan efisien, serta kemampuan untuk menyesuaikan tampilan data sesuai dengan kebutuhan pengguna. Antarmuka yang dirancang dengan baik akan meningkatkan pengalaman pengguna dan mempercepat respons terhadap kondisi yang dimonitor. Maka dari itu, pengembangan tampilan interface web dashboard untuk sistem IoT perlu dirancang secara cermat dengan mempertimbangkan aspek teknis dan kenyamanan pengguna.

**1.2 Tujuan eksperimen**

Tujuan utama dari pembuatan tampilan interface web dashboard ini adalah untuk menyediakan media visualisasi data yang dapat diakses secara online, yang mempermudah pengguna dalam memantau dan menganalisis data dari perangkat IoT. Dengan dashboard ini, pengguna dapat melihat perubahan data secara real-time, mengidentifikasi anomali, serta membuat keputusan yang lebih cepat dan tepat berdasarkan informasi yang tersaji.

Selain itu, proyek ini bertujuan untuk mengembangkan antarmuka pengguna yang modern, responsif, dan mudah digunakan, baik di perangkat desktop maupun mobile. Desain dashboard akan dirancang dengan prinsip user experience (UX) dan user interface (UI) yang optimal, sehingga pengguna dari berbagai latar belakang teknis dapat mengakses dan memahami informasi tanpa hambatan yang berarti.

**2. Methodology (Metodologi)**

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

Alat yang digunakan :

Laptop

Software yang digunakan :

VSCode

Laravel 11

XAMPP

**2.2 Implementation Steps (Langkah Implementasi)**

* Membuka Laravel yang telah dipakai sebelumnya
* Buka terminal dan jalankan code

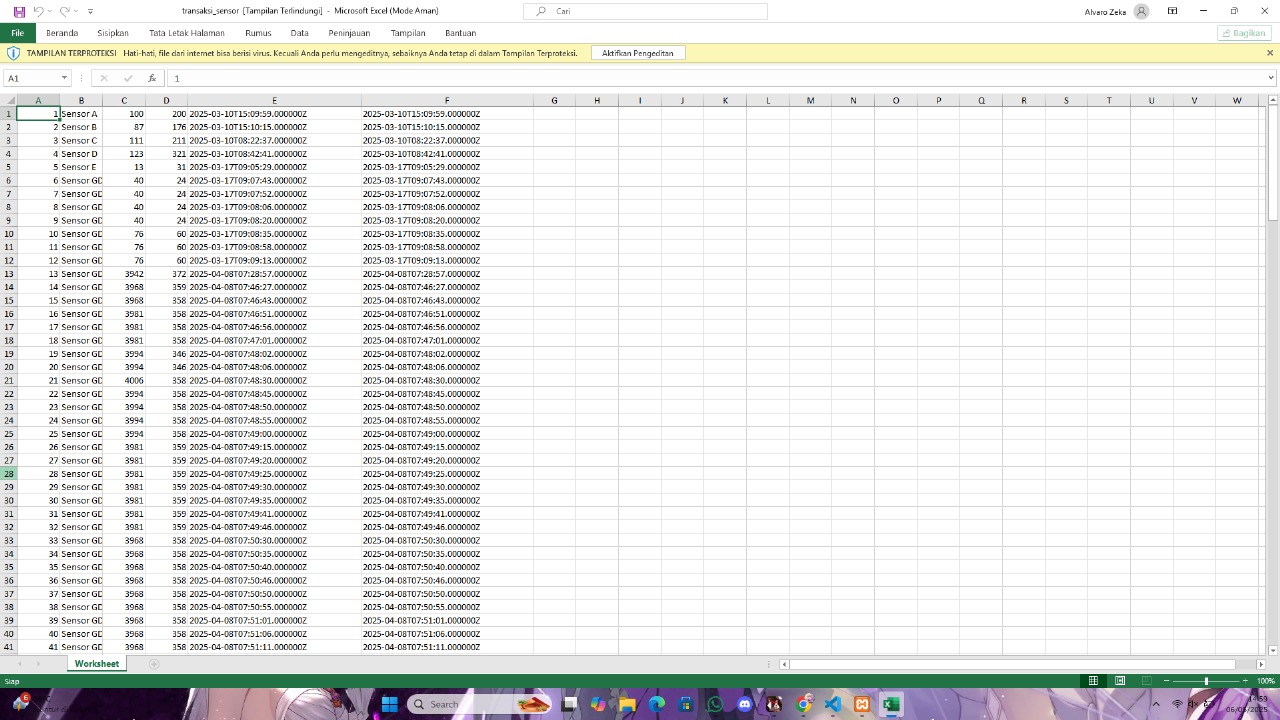
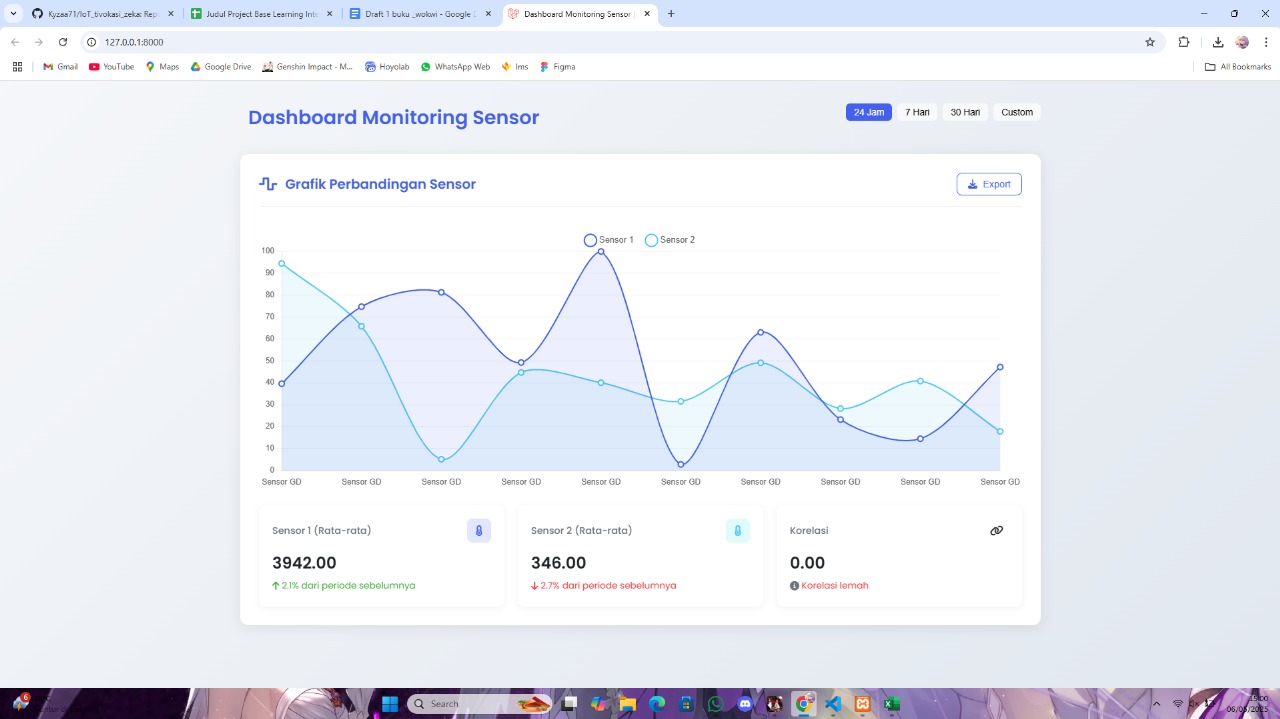
Composer require maatwebsite/excel dan

Php artisan make:Controller GraphController

* Tambahkan code ke GraphController
* Jalankan di terminal php artisan make:export TransaksiSensorExport --model=TransaksiSensor
* Edit file web.php
* Tambahkan program ke Graph.blade.php
* Jalankan php artisan serve
* Buka web dan export excel

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

**3.1 Experimental Result**



**4. Appendix (Lampiran, jika diperlukan)**

Kode Program:

GraphController :

<?php

namespace App\Http\Controllers;

use App\Exports\TransaksiSensorExport;

use Maatwebsite\Excel\Facades\Excel;

use App\Models\TransaksiSensor;

class GraphController extends Controller

{

    /\*\*

     \* Menampilkan grafik transaksi sensor.

     \*

     \* @return \Illuminate\View\View

     \*/

    public function index()

    {

        // Mengambil data transaksi sensor

        $transaksiSensors = TransaksiSensor::latest()->take(10)->get();

        // Mengambil data label

        $labels = $transaksiSensors->pluck('nama\_sensor');

        // Mengambil data nilai1 dan nilai2 untuk grafik

        $dataNilai1 = $transaksiSensors->pluck('nilai1');

        $dataNilai2 = $transaksiSensors->pluck('nilai2');

        return view('graph', compact('labels', 'dataNilai1', 'dataNilai2'));

    }

    /\*\*

     \* Mengunduh data transaksi sensor dalam format Excel

     \*

     \* @return \Symfony\Component\HttpFoundation\BinaryFileResponse

     \*/

    public function exportToExcel()

    {

        return Excel::download(new TransaksiSensorExport, 'transaksi\_sensor.xlsx');

    }

}

TransaksiSensorExports :

<?php

namespace App\Exports;

use App\Models\TransaksiSensor;

use Maatwebsite\Excel\Concerns\FromCollection;

class TransaksiSensorExport implements FromCollection

{

    /\*\*

    \* @return \Illuminate\Support\Collection

    \*/

    public function collection()

    {

        return TransaksiSensor::all();

    }

}

Web.php :

<?php

use Illuminate\Support\Facades\Route;

use App\Http\Controllers\GraphController;

Route::get('/', [GraphController::class, 'index'])->name('graph');

Route::get('/graph/export', [GraphController::class, 'exportToExcel'])->name('graph.export');