**Membuat Tampilan Interface  
 Web Dashboard IoT**

**Disusun Untuk Memenuhi Tugas Mata Kuliah :**

**Internet Of Things**

**Dosen Pengampu : Rachmad Andri Atmoko, S.ST, M.T, MCF**



**Disusun Oleh :**

1. Rifqi Athallah G.R(233140701111008)

PROGRAM STUDI TEKNOLOGI INFORMASI

FAKULTAS VOKASI

UNIVERSITAS BRAWIJAYA

2025

**Abstract**

Percobaan ini bertujuan untuk menghubungkan sensor suhu dan kelembaban DHT22 dengan mikrokontroler ESP32 dan mengirimkan data suhu serta kelembaban melalui API yang dibangun menggunakan Laravel. Data yang diterima oleh API tersebut kemudian disimpan dalam database MySQL. Dalam percobaan ini, digunakan Arduino uno untuk mengembangkan dan meng-upload kode ke ESP32, serta Ngrok untuk membuat URL publik agar API dapat diakses dari jaringan eksternal. Setelah data berhasil dikirimkan dari ESP32, hasilnya diverifikasi dengan memeriksa output pada Serial Monitor dan memastikan data yang diterima tersimpan dengan benar dalam database MySQL.

Selain itu, eksperimen ini juga membangun sebuah Web Dashboard berbasis Laravel Blade untuk menampilkan data yang diterima dari sensor secara real-time. Web dashboard ini memungkinkan pengguna untuk memantau data suhu dan kelembaban yang dikirimkan oleh ESP32 secara interaktif melalui antarmuka pengguna yang mudah dipahami.

Dengan adanya percobaan ini, diharapkan dapat memberikan pemahaman yang lebih dalam mengenai pengimplementasian Internet of Things (IoT), pengolahan data sensor secara real-time, serta pengembangan API menggunakan Laravel dan visualisasi data dalam bentuk dashboard berbasis web.

# **BAB I**

# **PENDAHULUAN**

## **1.1** **Latar Belakang**

Perkembangan teknologi Internet of Things (IoT) telah membawa perubahan signifikan dalam cara kita berinteraksi dengan dunia fisik. Dengan menggunakan IoT, perangkat fisik seperti sensor dapat terhubung dengan internet untuk mengirimkan data secara real-time ke server dan database untuk analisis lebih lanjut. Salah satu contoh aplikasi IoT adalah ESP32, sebuah mikrokontroler dengan konektivitas WiFi yang memungkinkan perangkat untuk mengirimkan data ke server secara real-time.

Dalam percobaan ini, sensor DHT22 digunakan untuk mengukur suhu dan kelembaban udara, dan data yang diperoleh dari sensor ini akan dikirimkan ke server Laravel melalui API yang dibangun. Data tersebut akan disimpan dalam database MySQL dan dapat dilihat dalam bentuk dashboard berbasis web. Selain itu, untuk membuat API Laravel dapat diakses secara eksternal, Ngrok digunakan untuk melakukan tunneling agar API dapat diakses melalui internet.

Percobaan ini bertujuan untuk memberikan pemahaman lebih dalam mengenai cara menghubungkan sensor DHT22 dengan ESP32, mengirimkan data ke API Laravel, serta menampilkan data tersebut dalam web dashboard menggunakan Laravel Blade.

## **1.2 Tujuan eksperimen**

* Menghubungkan sensor DHT22 dengan ESP32 untuk mendapatkan data suhu dan kelembaban.
* Mengirimkan data suhu dan kelembaban tersebut melalui API Laravel yang dibangun untuk diterima di server.
* Menyimpan data yang diterima dalam database MySQL dan memverifikasi bahwa data tersimpan dengan benar.
* Menampilkan data suhu dan kelembaban pada Web Dashboard IoT menggunakan Laravel Blade.

# 

# **BAB II**

**Metodologi**

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

* ESP32
* Breadboard
* Kabel Jumper
* Sensor DHT22
* PlatformIO
* Laravel untuk API
* MySQL untuk database
* Ngrok untuk membuat URL publik
* Visual Studio Code
* Kabel USB Type-C

**2.2 Implementation Steps**

1. Membuat API di Laravel

* Install Laravel dan buat API untuk menerima data suhu dan kelembaban dari ESP32.
* Jalankan server Laravel dengan perintah: php artisan serve --host=0.0.0.0 --port=8080

1. Jalankan Ngrok dengan perintah berikut untuk mendapatkan URL yang dapat diakses secara publik:

ngrok http --scheme=http 8080

1. **Wiring Kabel**

* Hubungkan sensor DHT22 ke ESP32 seperti berikut ini:
  + **VCC** dari DHT22 ke **3V** pada ESP32
  + **GND** dari DHT22 ke **GND** pada ESP32
  + **Data** dari DHT22 ke pin **GPIO4** pada ESP32

1. Modifikasi Kode dalam main.cpp
2. Edit file platformio.ini untuk menambahkan dependensi yang diperlukan:

Pada monitor\_port menggunakan COM3

monitor\_speed 115200

dan install libary pendukung tambahan pada arduino uno

adafruit/DHT sensor library@^1.4.4

adafruit/Adafruit Unified [Sensor@^1.1.14](mailto:Sensor@%5e1.1.14)

**Proses Upload dan Simulasi**  
Lakukan verifikasi dan upload kode ke ESP32 menggunakan PlatformIO. Setelah upload selesai, buka Serial Monitor di Visual Studio Code. Data suhu dan kelembaban dari sensor DHT22 akan dikirimkan setiap 5 detik.

**Verifikasi di MySQL**  
Setelah data berhasil dikirimkan, buka database MySQL dan periksa apakah data yang diterima dari ESP32 telah tersimpan dalam tabel yang sesuai.

# **BAB III**

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

**3.1 Hasil Percobaan**

Setelah menghubungkan sensor DHT22 dengan Arduino Uno dan mengupload kode yang sesuai, data suhu dan kelembaban berhasil dibaca oleh Arduino Uno. Data tersebut kemudian dikirimkan ke server Laravel melalui API menggunakan HTTP POST dengan bantuan modul ESP8266 (untuk WiFi), yang dihubungkan ke Arduino Uno. Proses pengiriman data ke server Laravel berhasil dilakukan dengan interval pengiriman data setiap 5 detik.

Data yang diterima dari sensor DHT22 berhasil ditampilkan pada Serial Monitor di Arduino IDE, dan setelah memverifikasi data pada database MySQL, hasilnya menunjukkan bahwa suhu dan kelembaban berhasil tersimpan dalam tabel yang sesuai.

Selain itu, setelah API Laravel berjalan dengan menggunakan Ngrok, data yang dikirimkan oleh Arduino Uno dapat diakses secara publik melalui URL yang diberikan oleh Ngrok. Web Dashboard berbasis Laravel Blade berhasil menampilkan data suhu dan kelembaban dalam bentuk tabel, dan pengguna dapat memantau data secara real-time.

**3.2 Pembahasan**

Percobaan ini berhasil membuktikan bahwa Arduino Uno, dengan bantuan modul ESP8266 untuk konektivitas WiFi, dapat mengirimkan data dari sensor DHT22 ke server Laravel menggunakan API. Berikut adalah beberapa hal yang perlu diperhatikan dalam percobaan ini:

1. Pengiriman Data dengan HTTP POST: Data suhu dan kelembaban berhasil dikirimkan menggunakan HTTP POST dari Arduino Uno ke API Laravel. ESP8266 digunakan untuk menyediakan konektivitas WiFi kepada Arduino Uno. Penggunaan ESP8266 memungkinkan Arduino Uno untuk mengirimkan data secara real-time melalui jaringan WiFi.
2. Penyimpanan Data di MySQL: Setelah data diterima oleh server Laravel melalui API, data suhu dan kelembaban disimpan dalam database MySQL. Data tersebut disimpan dalam format yang sesuai dan dapat diakses untuk analisis lebih lanjut. Proses ini menunjukkan bahwa Arduino Uno dapat berfungsi sebagai perangkat IoT yang menghubungkan sensor fisik dengan sistem berbasis cloud.
3. Web Dashboard dengan Laravel Blade: Data yang diterima berhasil ditampilkan dalam web dashboard menggunakan Laravel Blade. Dashboard ini menampilkan suhu dan kelembaban secara real-time dalam bentuk tabel yang mudah dibaca. Pengguna dapat memantau data yang diperoleh dari sensor DHT22 kapan saja melalui antarmuka web.

**3.3 Kesimpulan**

Percobaan ini berhasil menunjukkan penerapan sistem Internet of Things (IoT) dengan menghubungkan sensor suhu dan kelembaban DHT22 ke Arduino Uno dan mengirimkan data suhu serta kelembaban ke API Laravel. Data yang diterima oleh API Laravel berhasil disimpan dalam database MySQL dan ditampilkan secara real-time pada web dashboard berbasis Laravel Blade.

# **BAB IV**

**Experimental Results (Hasil Eksperimen)**

* 1. **Experimental Results**

Install composer require maatwebsite/excel di terminal project, disini harus disesuiakan untuk laravel anda dan php -v version. Untuk versi ini tidak dapat digunakan di laravel 11 dan php versi 8.2 keatas

Disini saya menggunakan laravel 11 dan php versi 8.2, pada versi composer require maatwebsite/excel yang di instal (v1.1.5) sudah tidak lagi didukung atau "abandoned".

Meng-upgrade ke versi maatwebsite/excel yang lebih baru yang mendukung phpoffice/phpspreadsheet.

Ketik perintah composer require maatwebsite/excel:^3.1 --with-all-dependencies (Ini akan memastikan Anda mendapatkan versi terbaru yang kompatibel dengan PHP 8.)

Cek apakah paket maatwebsite/excel sudah terinstal dan tersedia di file composer.json Anda. Coba jalankan perintah berikut untuk melihat apakah paket sudah terpasang: composer show maatwebsite/excel.

Jika masih belum terpasang pergi ke file composer.json coba lihat apakah sudah ada berikut ini.

"require": {

"maatwebsite/excel": "^3.1.50",

...

}

Jika dapat permasalahan berikut ini,

Problem 1 - phpoffice/phpspreadsheet[1.29.7, ..., 1.29.10] require ext-gd \* -> it is missing from your system. Install or enable PHP's gd extension. - maatwebsite/excel 3.1.62 requires phpoffice/phpspreadsheet ^1.29.7 -> satisfiable by phpoffice/phpspreadsheet[1.29.7, 1.29.8, 1.29.9, 1.29.10]. - Root composer.json requires maatwebsite/excel 3.1.62 -> satisfiable by maatwebsite/excel[3.1.62]. To enable extensions, verify that they are enabled in your .ini files: - C:\xampp\php\php.ini You can also run php --ini in a terminal to see which files are used by PHP in CLI mode.

Pada file php -> php.ini hapus ; pada extension=gd, setalah itu hapus dulu yang composer remove maatwebsite/excel. Baru install dengan perintah berikut ini.

1.composer require maatwebsite/excel:^3.1.62

2. composer show maatwebsite/excel

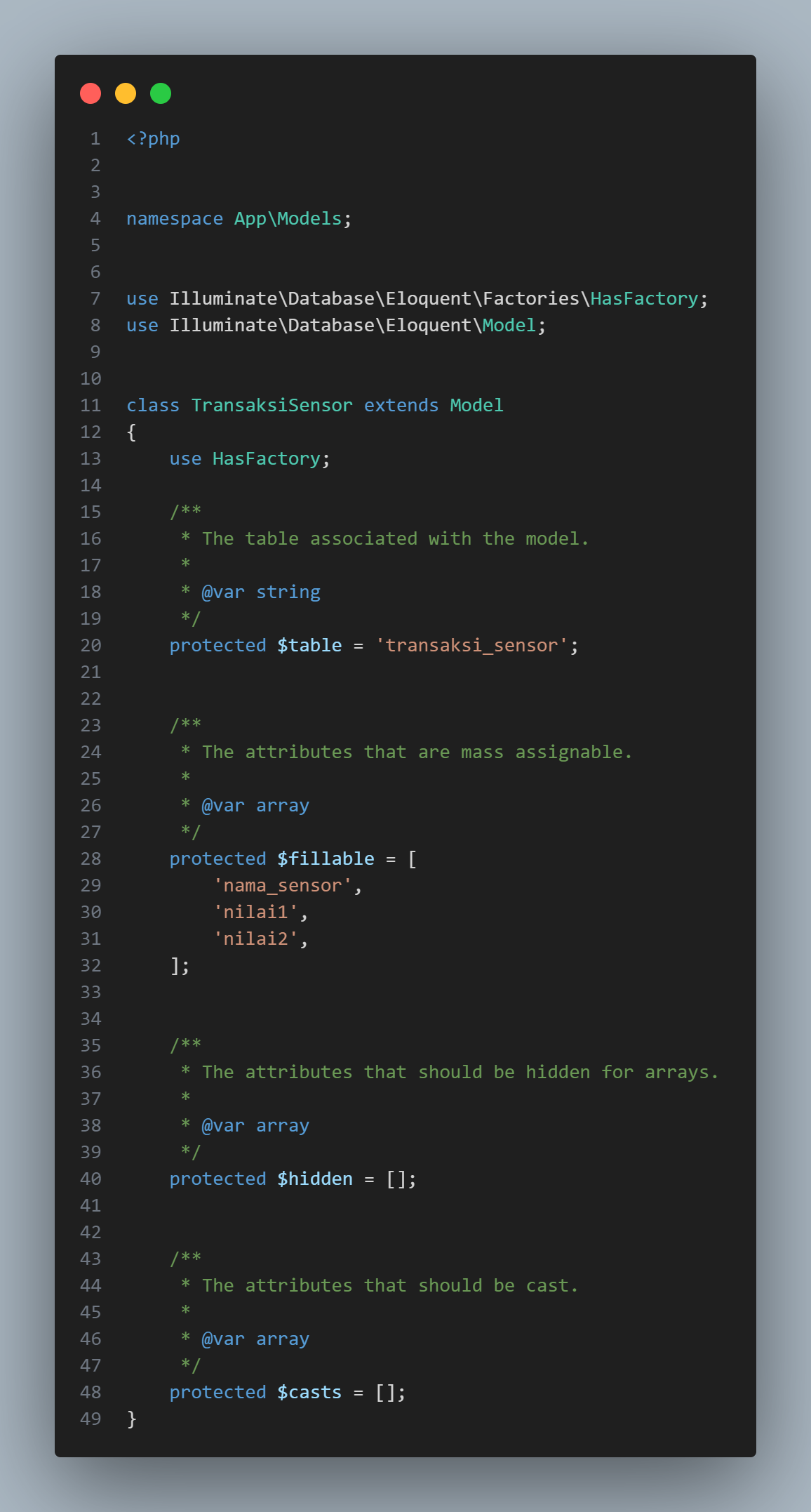
3. php artisan make:controller GraphController

Edit file dan sesuaikan.



4. php artisan make:export TransaksiSensorExport --model=TransaksiSensor

Edit file dan sesuaikan.



1. Edit file web.php



7.setelah itu edit file graph.blade.php untuk melihat hasil dhasbord nya.

8. php artisan serve

