

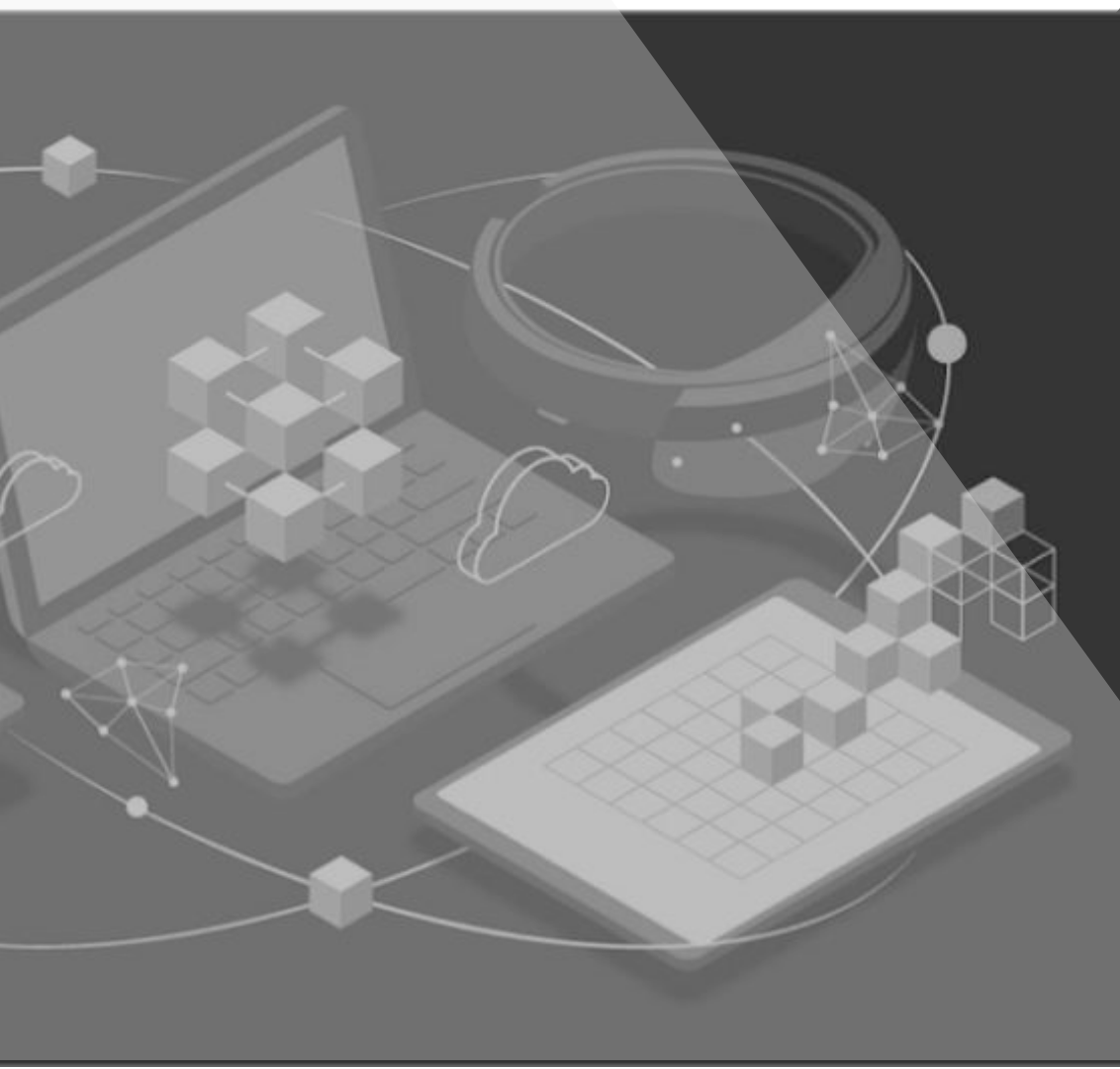


Backend dan frontend iot dashboard

Nodered influxdb backend dengan graphana

Table of contents

- 01 Pengenalan**
Pengenalan Dashboard IoT
- 02 Backend database**
Pengenalan Database
Influxdb
- 03 Node Red advanced**
Backend NodeRed
- 04 Dashboard**
Grafana Dashboard



01

Pengenalan



Goal

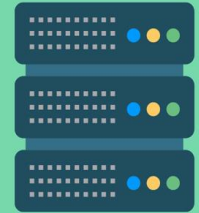
- Memahami sistem front end, back end, and full-stack development.
- Menggunakan database sqlite dan influxdb
- Menggunakan fungsi pada node-red
- Menampilkan data time line pada dashboard grafana



**front end, back end,
and full-stack
development.**



FRONT END



BACK END

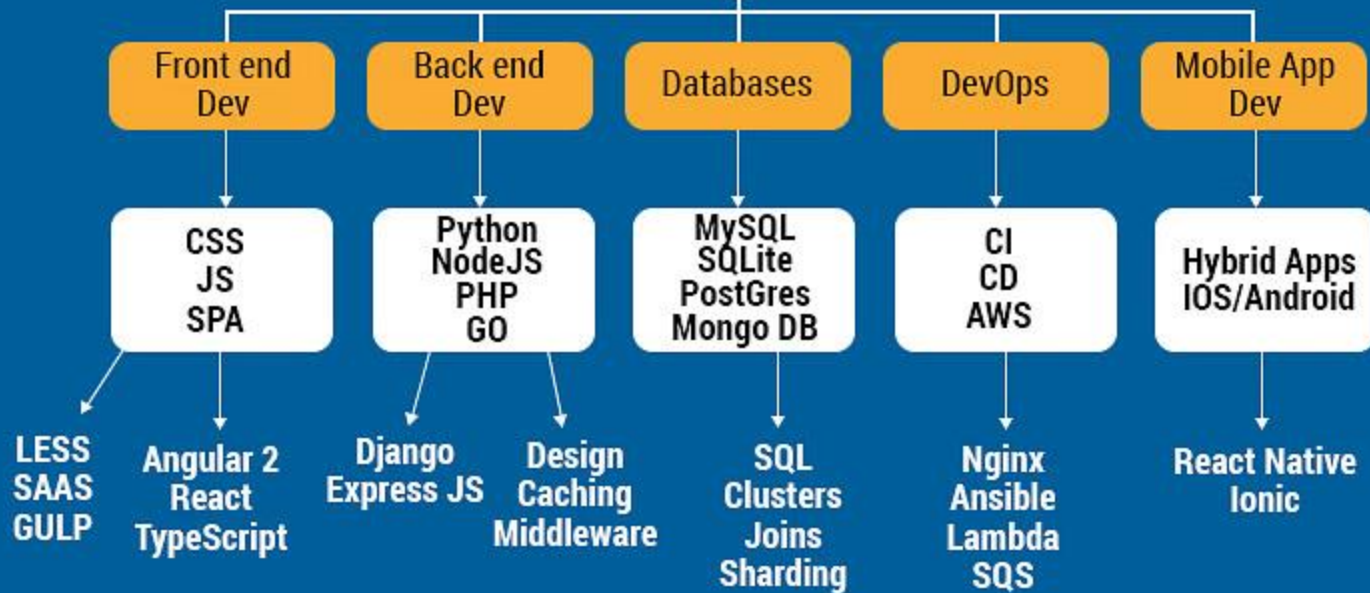
FRONT END

1. Memastikan kelayakan teknis desain dari UI/UX
2. Mengoptimalkan kecepatan dari *website*/aplikasi
3. Menerapkan elemen visual situs web atau aplikasi
4. Menggunakan desain responsif dalam pembuatan antarmuka pengguna situs web atau aplikasi
5. Menguji situs web atau aplikasi untuk kegunaan
6. Memecahkan masalah kode apa pun yang tidak berfungsi
7. Meningkatkan arsitektur visual situs web atau aplikasi
8. Memastikan bahwa semua kelayakan dari tampilan *website* sebelum mengirimkannya ke tim back end
9. Berkolaborasi tim lainnya

Back end

- 1.Mengelola dan mengembangkan sumber daya API (Application Programming Interface) yang berfungsi di seluruh perangkat
- 2.Membuat sistem pemrosesan pembayaran yang menyimpan data yang diperlukan dengan aman
- 3.Selalu perbarui aplikasi web, aman, dan cepat
- 4.Memantau status server
- 5.Menerapkan algoritma dan memecahkan masalah yang terkait dengan sistem server atau data base
- 6.Mengembangkan CMS (Content Management System);
- 7.Mendukung pengembangan frontend dengan komunikasi yang jelas dan dokumentasi yang baik
- 8.Menyimpan dan mengelola data secara efektif

FULL STACK DEVELOPER



Influxdb

Node-red untuk query for influxdb



InfluxDB

InfluxDB is an open-source high-performance time series database (TSDB) that can store large amounts of data per second. Each data point you submit to the database is associated with a particular timestamp. So, it is ideal for IoT datalogging projects like storing data from your weather station sensors.



influxdb

IOTstack Main Menu

```
-> Build Stack <-  
  Docker Commands  
Miscellaneous Commands  
  Backup and Restore  
  Native Installs  
    Exit
```

Service to install

- Nodered
- Portainer-ce
- Mosquito
- Influxdb
- Grafana

Install windows influxdb (service)

- <https://portal.influxdata.com/downloads/>
- `wget https://dl.influxdata.com/influxdb/releases/influxdb-1.8.10_windows_amd64.zip -UseBasicParsing -OutFile influxdb-1.8.10_windows_amd64.zip`
- `Expand-Archive .\influxdb-1.8.10_windows_amd64.zip -DestinationPath 'C:\Program Files\InfluxData\influxdb\'`

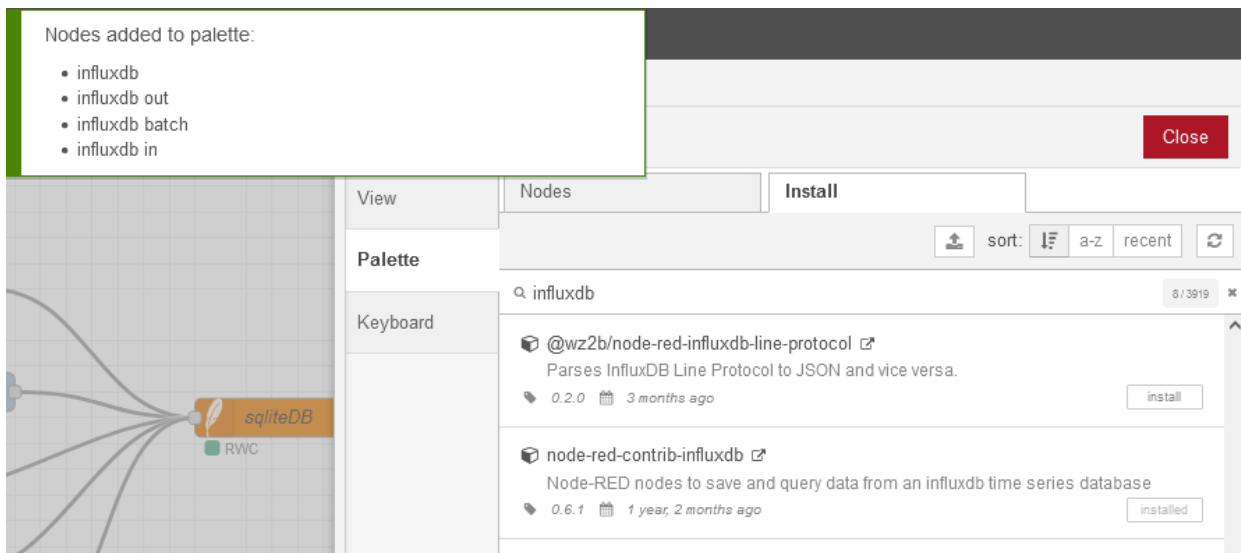
```
PS C:\Users\User> wget https://dl.influxdata.com/influxdb/releases/influxdb-1.8.10_windows_amd64.zip -UseBasicParsing -OutFile influxdb-1.8.10_windows_amd64.zip
PS C:\Users\User> X|
```

```
PS C:\Users\User\Documents\home\Project\FGD Tel-U 2022\influxdb-1.8.10-1> .\influxd.exe
```

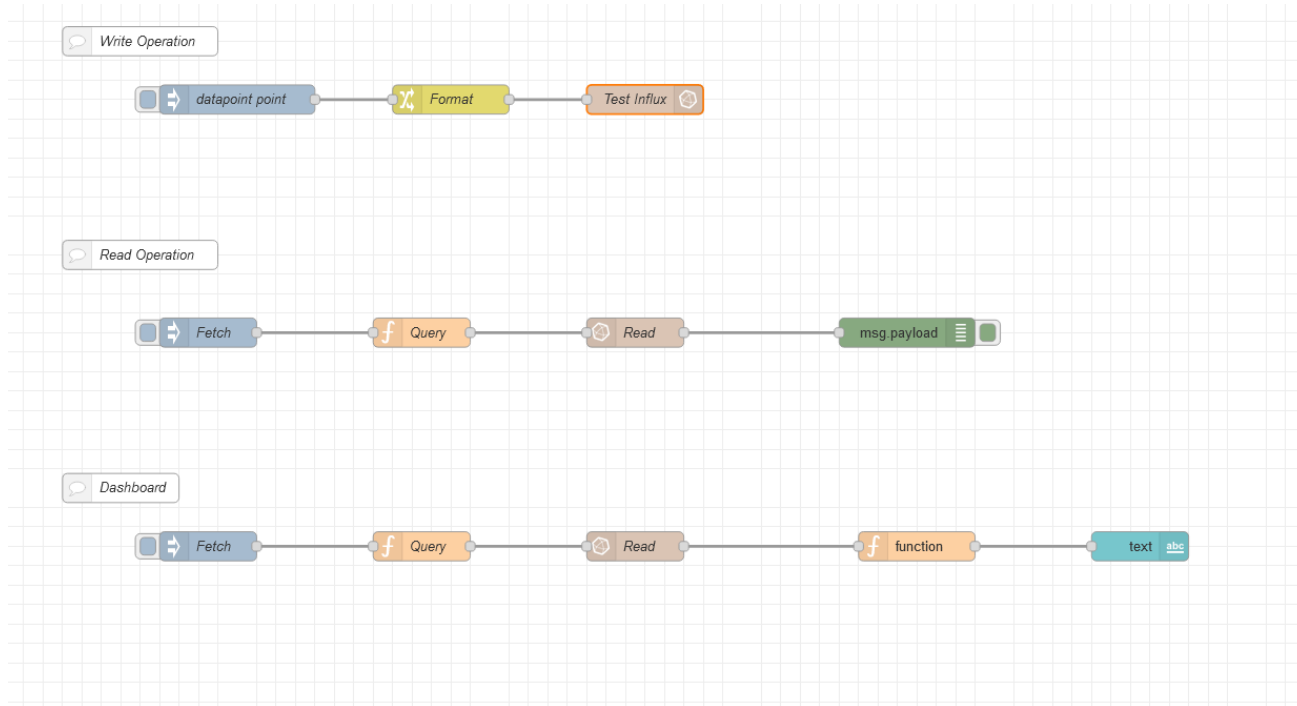
```
88888888      .d888 888      88888888b. 8888888b.
888      d88P" 888      888  "Y88b 888  "88b
888      888 888      888  888 888 .88P
888 888888b. 8888888 888 888 888 888 888 88888888K.
888 888 "88b 888 888 888 888 Y8bd8P' 888 888 888 "Y88b
888 888 888 888 888 888 888 X88K 888 888 888 888
888 888 888 888 888 Y88b 888 .d8"8b. 888 .d88P 888 d88P
88888888 888 888 888 888 "Y88888 888 888 8888888P" 88888888P"
```

Install influxdb

Install node-red-contrib-influxdb from install palette



Influxdb node red tutorial



Sql query

```
PS C:\Users\User\Documents\home\Project\FGD Tel-U 2022\influxdb-1.8.10-1> .\influx.exe
Connected to http://localhost:8086 version 1.8.10
InfluxDB shell version: 1.8.10
>
>
```

```
> show databases
```

```
name: databases
```

```
name
```

```
----
```

```
_internal
```

```
sensordata
```

```
> use sensordata
```

```
Using database sensordata
```

```
> select * from "sensordata"
```

```
name: sensordata
```

```
time
```

```
timestamp
```

```
type
```

```
value
```

```
----
```

```
-----
```

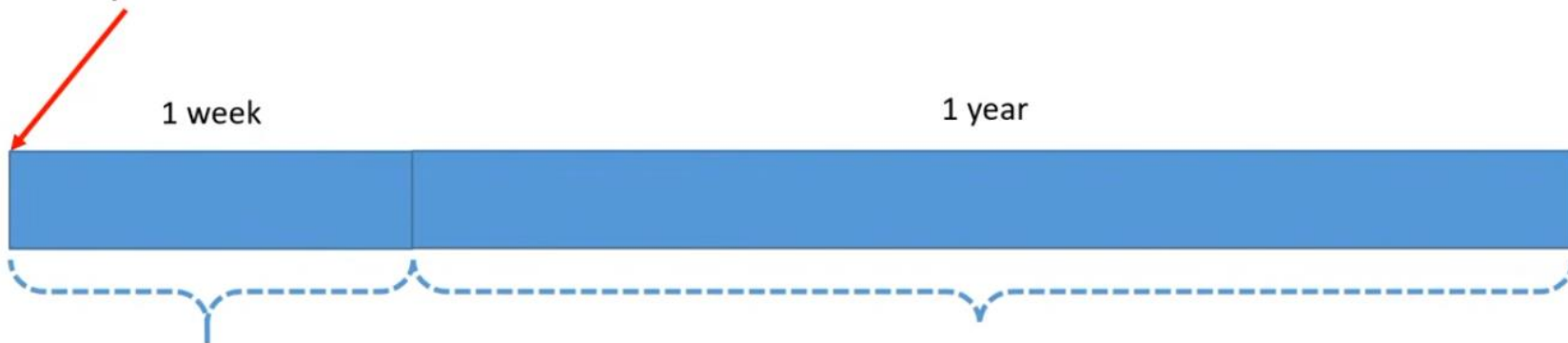
```
----
```

```
-----
```

```
1655866970748378600 1655866970745 point 14.454086270623922
```

```
1655866991794885500 1655866991792 point 6.214789227636464
```


Data every 2 minutes



1 week = 7d x 24h x 30 = 5'040 data sets

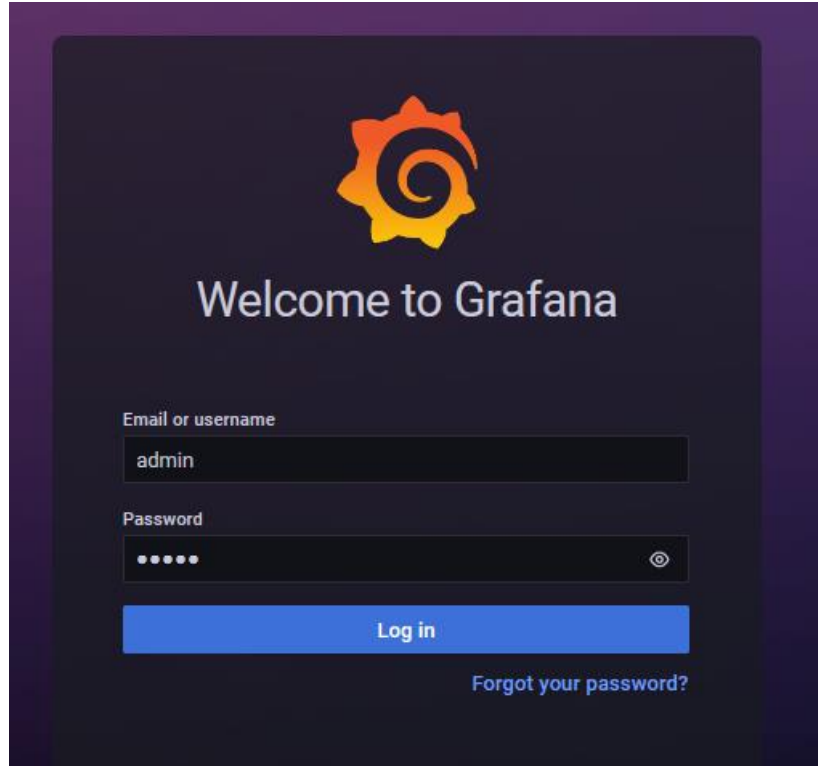
1 year = 52 weeks = 52 x 5'040 = 262'080 data sets

The background of the image features a large, dark gray triangle pointing upwards, centered on the page. To the left of this triangle is a light gray area, and to the right is a white area. A thin, dark gray diagonal line runs from the top right corner towards the center of the dark triangle.

GRAFANA

Dashboard

Open `http://localhost:3000`



Username : admin
Password : admin
Configure datasource

Database connection

HTTP

URL ⓘ

127.0.0.1:8086

Access

Server (default) ▾

Help >

Allowed cookies ⓘ

New tag (enter key to add)

Timeout ⓘ

Timeout in seconds

Auth

Basic auth

☐

With Credentials ⓘ

☐

TLS Client Auth

☐

With CA Cert ⓘ

☐

Skip TLS Verify

☐

Forward OAuth Identity ⓘ

☐

Custom HTTP Headers

+ Add header

InfluxDB Details

Database Access

Setting the database for this datasource does not deny access to other databases. The InfluxDB query syntax allows switching database with `USE` or `SHOW MEASUREMENTS ON _internal` or `SELECT * FROM "_internal"."database" LIMIT 10`

To support data isolation and security, make sure appropriate permissions are configured in InfluxDB.

Database

sensordata

Dashboard



Thanks

Do you have any question?

hasbiida@gmail.com



CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, infographics & images by **Freepik**