



## **USER MANUAL**

PERANCANGAN DASHBOARD UNTUK  
PENGAWASAN AKTIVITAS TRUK  
PADA PT CIPTA NIAGA SEMESTA (ANAK  
PERUSAHAAN MAYORA)

**Disusun oleh:**

DAVID RAHARJA

NIM. 825190027

PROGRAM STUDI SISTEM INFORMASI  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS TARUMANAGARA  
2023

## **DAFTAR ISI**

<b>USER MANUAL .....</b>	<b>1</b>
<b>DAFTAR ISI.....</b>	<b>2</b>
<b>SYSTEM REQUIREMENTS .....</b>	<b>3</b>
<b>PROSEDUR UNDUH PROGRAM DAN MEMASANG DASHBOARD.....</b>	<b>4</b>
<b>PENGUNAAN DASHBOARD.....</b>	<b>15</b>

## **SYSTEM REQUIREMENTS**

### **1.1 Minimum Perangkat Keras**

Penggunaan Dashboard untuk pengawasan aktivitas truk pada PT Cipta Niaga Semesta, diperlukan spesifikasi perangkat keras sebagai berikut:

1. Processors: Apple Macintosh Dual-Core; Intel EM64T or AMD64 Dual-Core  
Ubuntu Desktop 16.04 LTS & 18.04 LTS
2. RAM: 2 GB
3. Disk Space: 2.5 GB

### **1.2 Minimum Perangkat Lunak**

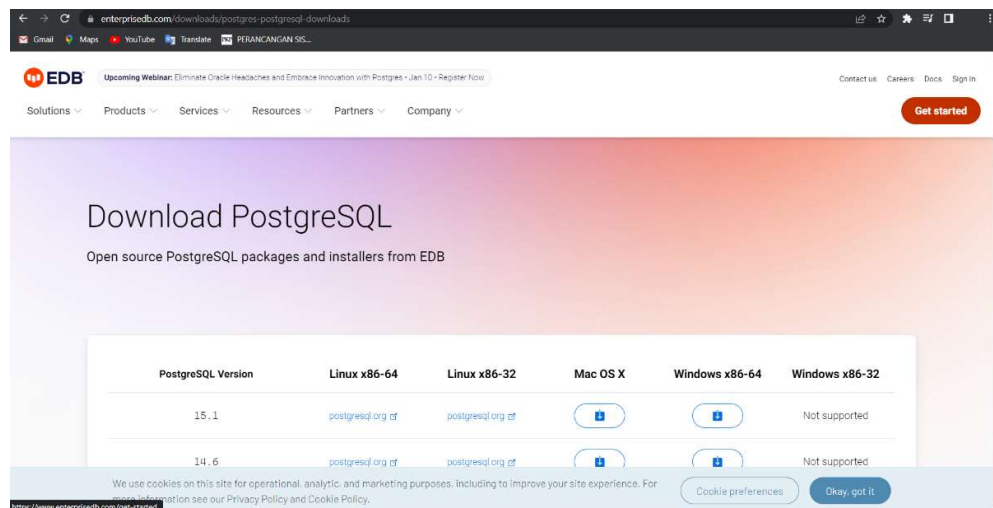
Penggunaan Dashboard untuk pengawasan aktivitas truk pada PT Cipta Niaga Semesta, diperlukan spesifikasi perangkat lunak sebagai berikut:

1. Operating system Windows 8 atau lebih
2. Web Browser

## PROSEDUR UNDUH PROGRAM DAN MEMASANG DASHBOARD

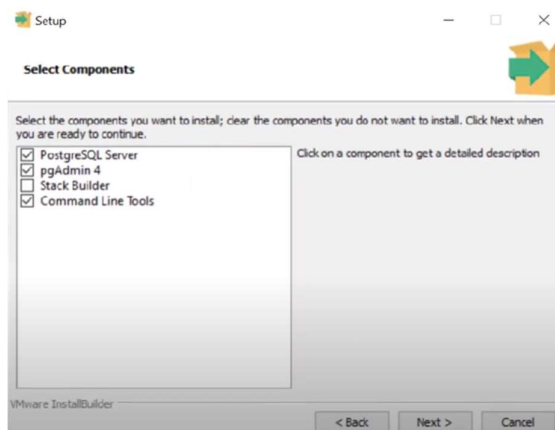
Berikut ini adalah langkah-lagkah dalam mengunduh program yang diperlukan agar *dashboard* berjalan dengan benar:

1. Mengunduh Postgresql dalam alamat *website* <https://www.enterprisedb.com/>. Halaman website dapat dilihat pada **Gambar 1**.



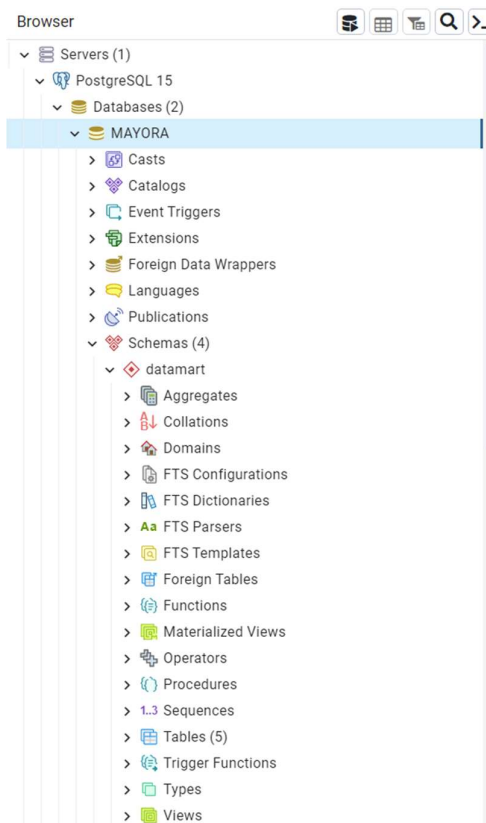
**Gambar 1** Halaman *Website* Unduh PostgreSQL

2. Membuka file .exe dan melakukan *setup* yang dapat dilihat pada **Gambar 2**.



**Gambar 2** *Setup* PostgreSQL

3. Setelah selesai setup, buka aplikasi pgadmin 4 dalam *search box* windows.
4. Masukkan password yang telah dipasang sebelumnya saat *setup*.
5. Membuat database baru “MAYORA” atau nama lain, kemudian membuat *schema* baru bernama “datamart”. Tahap ini dapat dilihat pada **Gambar 3**.



**Gambar 3** Database dan Schema Baru

6. Klik kanan datamart dan pilih “Query Tool” untuk membuka *query*.
7. Dalam *query* buat *create table* untuk tiap dimensi dan tabel fakta. **Gambar 4** menunjukkan *query* untuk dimensi *material*, **Gambar 5** menunjukkan *query* untuk dimensi *plant*, **Gambar 6** menunjukkan *query* untuk dimensi *time*, **Gambar 7** menunjukkan *query* untuk dimensi *vendor*, dan **Gambar 8** menunjukkan *query* untuk tabel fakta.

```

CREATE TABLE IF NOT EXISTS datamart.dim_material
(
    sk_material integer NOT NULL,
    material_code character varying(45) COLLATE pg_catalog."default",
    material_name character varying(75) COLLATE pg_catalog."default",
    material_type_code character varying(6) COLLATE pg_catalog."default",
    material_type_name character varying(75) COLLATE pg_catalog."default",
    CONSTRAINT dim_material_pkey PRIMARY KEY (sk_material)
)

```

**Gambar 4 Query Dimensi Material**

```

CREATE TABLE IF NOT EXISTS datamart.dim_plant
(
    sk_plant integer NOT NULL,
    plant_code character varying(10) COLLATE pg_catalog."default",
    plant_name character varying(75) COLLATE pg_catalog."default",
    plant_pengirim character varying(20) COLLATE pg_catalog."default",
    CONSTRAINT dim_plant_pkey PRIMARY KEY (sk_plant)
)

```

**Gambar 5 Query Dimensi Plant**

```

CREATE TABLE IF NOT EXISTS datamart.dim_time
(
    sk_waktu integer NOT NULL,
    hari double precision,
    kuartal text COLLATE pg_catalog."default",
    bulan double precision,
    year double precision,
    tanggal date,
    CONSTRAINT dim_time_pkey PRIMARY KEY (sk_waktu)
)

```

**Gambar 6 Query Dimensi Time**

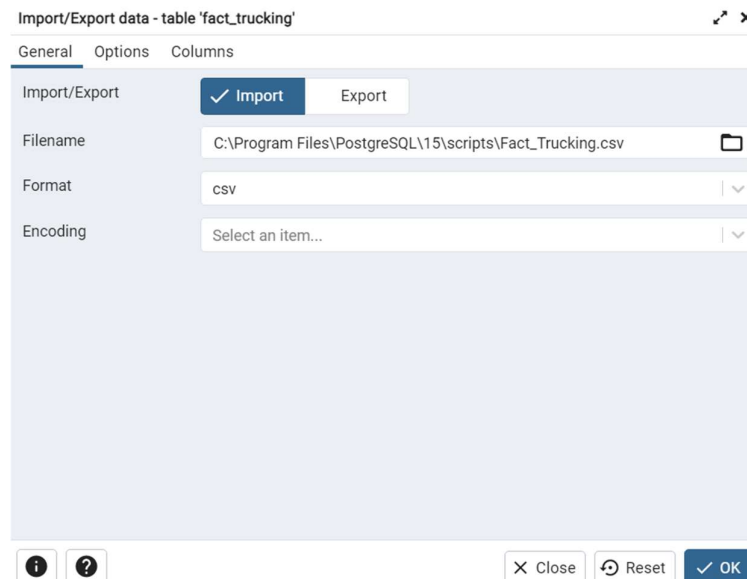
```
CREATE TABLE IF NOT EXISTS datamart.dim_vendor
(
    sk_vendor integer NOT NULL,
    vendor_code character varying(50) COLLATE pg_catalog."default",
    license_no character varying(45) COLLATE pg_catalog."default",
    jenis_mobil character varying(50) COLLATE pg_catalog."default",
    CONSTRAINT dim_vendor_pkey PRIMARY KEY (sk_vendor)
)
```

**Gambar 7 Query Dimensi Vendor**

```
CREATE TABLE IF NOT EXISTS datamart.fact_trucking
(
    sk_plant integer,
    sk_vendor integer,
    sk_material integer,
    sk_waktu integer,
    avg_lama_antri double precision,
    avg_lama_bongkar double precision,
    avg_lama_proses double precision,
    avg_lama_pending double precision,
    avg_qty_per_mobil double precision,
    mobil_datang double precision,
    barang_dibongkar double precision,
    persen_bongkar double precision
)
```

**Gambar 8 Query Tabel Fakta**

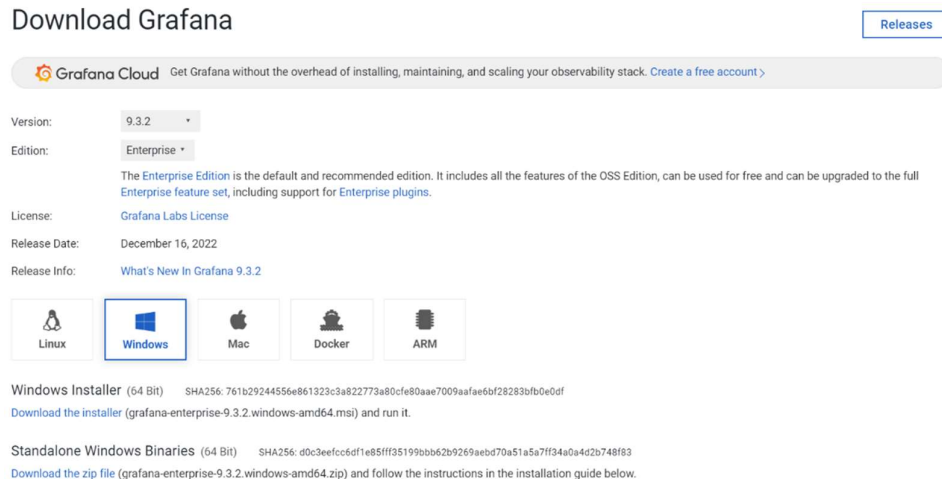
8. Setelah membuat semua tabel dimensi dan tabel fakta, dilanjutkan dengan mengimpor data ke dalam tabel-tabel tersebut. Ekspor data dari laptop penulis dalam bentuk .csv dan pindahkan ke komputer yang akan membuat *dashboard* pada *program files* PostgreSQL bagian *scripts*.
9. Dalam *tables* datamart klik kanan salah satu tabel dimensi atau tabel fakta dan pilih “import/export data”. **Gambar 9** menunjukkan *pop-up* import atau export data. Pilih file .csv yang ingin diimpor dan klik tombol “OK”.



**Gambar 9** *Import/Export Data*

10. Setelah selesai impor semua data ke tiap tabel dimensi dan fakta, periksa dalam *query* untuk melihat apakah data sudah sesuai.
11. Selanjutnya gunakan alamat *website* <https://grafana.com/grafana/download> untuk mengunduh Grafana. Halaman *website* dapat dilihat pada **Gambar 10**.

## Download Grafana

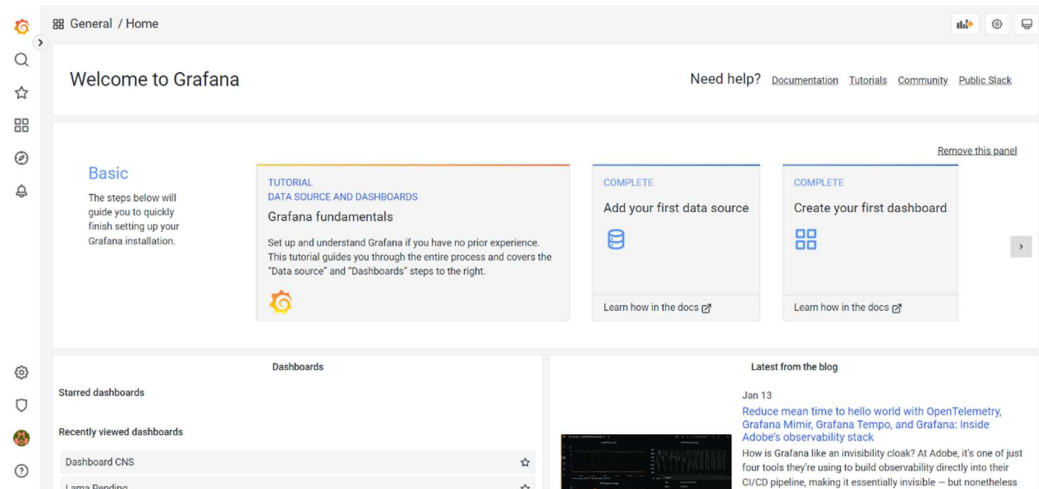


**Gambar 10** *Halaman Website Unduh Grafana*

12. Membuka aplikasi .msi untuk mengunduh Grafana dan lakukan *setup*.

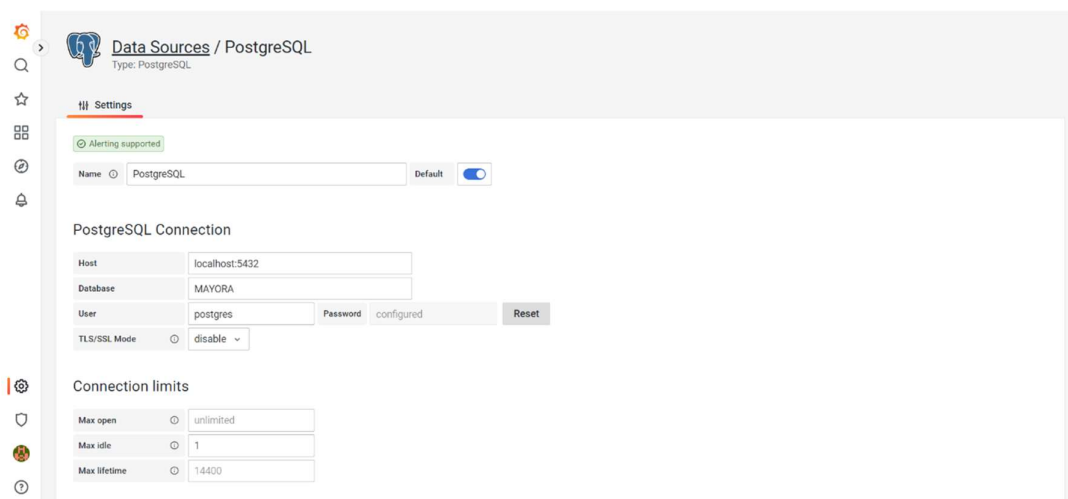


13. Setelah proses setup telah selesai, buka web browser pilihan dan masukkan <http://localhost:3000/>.
14. Dalam halaman Grafana klik setting dan pilih configuration. Tahap ini dapat dilihat pada **Gambar 11**.



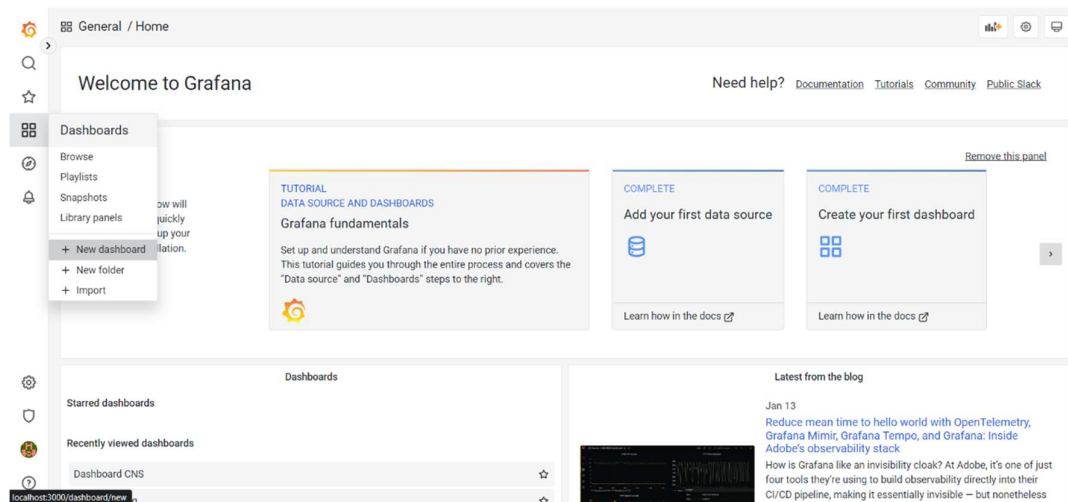
**Gambar 11** Halaman Grafana

15. Setelah itu, klik “Add data source” dan pilih PostgreSQL. Kemudian isi *setting data source* sesuai dengan PostgreSQL dan klik “Save & Test”. *Setting data source* dapat dilihat pada **Gambar 12**.



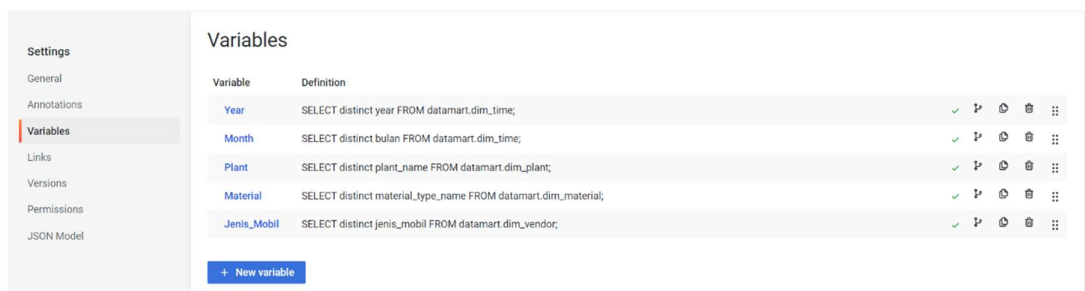
**Gambar 12** Setting Data Source

16. Selanjutnya, **Gambar 13** menunjukkan dropdown *dashboard* untuk membuat *dashboard* baru.



**Gambar 13** Grafana *New Dashboard*

17. Dalam dashboard baru, terdapat panel-panel yang dapat diatur untuk menampilkan data.
18. Sebelum memasang panel untuk menampilkan data, pasang *filter* terlebih dahulu dalam dashboard setting kemudian bagian variables yang dapat dilihat pada **Gambar 14**. Klik tombol “+ New variable” untuk mengisi nama, label, data source PostgreSQL, dan *query* sesuai dengan filter yang ingin dipasang.



**Gambar 14** *Dashboard Variables*

19. Selanjutnya membuat tampilan data pada setiap *panel*. **Gambar 15** menunjukkan *query* untuk menampilkan service level bongkar per bulan. **Gambar 16** menunjukkan *query* untuk menampilkan rata-rata lama proses.

**Gambar 17** menunjukkan *query* untuk menampilkan standard bongkar. **Gambar 18** menunjukkan *query* untuk menampilkan perbandingan antara mobil datang dan barang dibongkar. **Gambar 19** menunjukkan query rata-rata pending. **Gambar 20** menunjukkan *query* untuk menampilkan rata-rata lama bongkar. **Gambar 21** menunjukkan *query* untuk menampilkan rata-rata lama antri. **Gambar 22** menunjukkan *query* untuk menampilkan rata-rata kuantitas barang per mobil.

```
select avg(a.persen_bongkar) as Persentase_Bongkar, $__time(b.bulan)
from datamart.fact_trucking a join datamart.dim_time b on (a.sk_waktu =
b.sk_waktu)
join datamart.dim_plant c on (a.sk_plant = c.sk_plant)
join datamart.dim_material d on (a.sk_material = d.sk_material)
join datamart.dim_vendor e on (a.sk_vendor = e.sk_vendor)
where b.year in (${Year:sqlstring})
and b.bulan in (${Month:sqlstring})
and c.plant_name in (${Plant:sqlstring})
and d.material_type_name in (${Material:sqlstring})
and e.jenis_mobil in (${Jenis_Mobil:sqlstring})
group by b.bulan
order by b.bulan;
```

**Gambar 15** *Query Service Level Bongkar per Bulan*

```
select avg(a.avg_lama_proses)/60 as Bulan, $__time(b.bulan)
from datamart.fact_trucking a join datamart.dim_time b on (a.sk_waktu =
b.sk_waktu)
join datamart.dim_plant c on (a.sk_plant = c.sk_plant)
join datamart.dim_material d on (a.sk_material = d.sk_material)
join datamart.dim_vendor e on (a.sk_vendor = e.sk_vendor)
where b.year in (${Year:sqlstring})
and b.bulan in (${Month:sqlstring})
and c.plant_name in (${Plant:sqlstring})
and d.material_type_name in (${Material:sqlstring})
and e.jenis_mobil in (${Jenis_Mobil:sqlstring})
group by b.bulan
order by b.bulan;
```

**Gambar 16** *Query Rata-Rata Lama Proses*

```

select $__time(b.bulan),b.bulan+1 as bulan, avg(a.avg_lama_bongkar) as
std_bongkar
from datamart.fact_trucking a join datamart.dim_time b on (a.sk_waktu =
b.sk_waktu)
join datamart.dim_plant c on (a.sk_plant = c.sk_plant)
join datamart.dim_material d on (a.sk_material = d.sk_material)
join datamart.dim_vendor e on (a.sk_vendor = e.sk_vendor)
where b.year in (${Year:sqlstring})
and b.bulan+1 in (${Month:sqlstring})
and c.plant_name in (${Plant:sqlstring})
and d.material_type_name in (${Material:sqlstring})
and e.jenis_mobil in (${Jenis_Mobil:sqlstring})
and bulan <12
group by b.bulan
order by b.bulan;

```

**Gambar 17** *Query* Standard Bongkar

```

select avg(a.Mobil_Datang) as Mobil_Datang, avg(a.Barang_Dibongkar) as
Dibongkar, $__time(b.tanggal)
from datamart.fact_trucking a join datamart.dim_time b on (a.sk_waktu =
b.sk_waktu)
join datamart.dim_plant c on (a.sk_plant = c.sk_plant)
join datamart.dim_material d on (a.sk_material = d.sk_material)
join datamart.dim_vendor e on (a.sk_vendor = e.sk_vendor)
where b.year in (${Year:sqlstring})
and b.bulan in (${Month:sqlstring})
and c.plant_name in (${Plant:sqlstring})
and d.material_type_name in (${Material:sqlstring})
and e.jenis_mobil in (${Jenis_Mobil:sqlstring})
group by b.tanggal
order by b.tanggal;

```

**Gambar 18** *Query* Mobil Datang Dibanding Barang Dibongkar

```

select avg(a.avg_lama_pending) as Pending, $__time(b.tanggal)
from datamart.fact_trucking a join datamart.dim_time b on (a.sk_waktu =
b.sk_waktu)
join datamart.dim_plant c on (a.sk_plant = c.sk_plant)
join datamart.dim_material d on (a.sk_material = d.sk_material)
join datamart.dim_vendor e on (a.sk_vendor = e.sk_vendor)
where b.year in (${Year:sqlstring})
and b.bulan in (${Month:sqlstring})
and c.plant_name in (${Plant:sqlstring})
and d.material_type_name in (${Material:sqlstring})
and e.jenis_mobil in (${Jenis_Mobil:sqlstring})
group by b.tanggal
order by b.tanggal;

```

**Gambar 19 Query Rata-Rata Lama Pending**

```

select avg(a.avg_lama_bongkar)/60 as avg_bongkar, $__time(b.tanggal)
from datamart.fact_trucking a join datamart.dim_time b on (a.sk_waktu =
b.sk_waktu)
join datamart.dim_plant c on (a.sk_plant = c.sk_plant)
join datamart.dim_material d on (a.sk_material = d.sk_material)
join datamart.dim_vendor e on (a.sk_vendor = e.sk_vendor)
where b.year in (${Year:sqlstring})
and b.bulan in (${Month:sqlstring})
and c.plant_name in (${Plant:sqlstring})
and d.material_type_name in (${Material:sqlstring})
and e.jenis_mobil in (${Jenis_Mobil:sqlstring})
group by b.tanggal
order by b.tanggal;

```

**Gambar 20 Query Rata-Rata Lama Bongkar**

```

select avg(a.avg_lama_antri)/60 as avg_antri, $__time(b.tanggal)
from datamart.fact_trucking a join datamart.dim_time b on (a.sk_waktu =
b.sk_waktu)
join datamart.dim_plant c on (a.sk_plant = c.sk_plant)
join datamart.dim_material d on (a.sk_material = d.sk_material)
join datamart.dim_vendor e on (a.sk_vendor = e.sk_vendor)
where b.year in (${Year:sqlstring})
and b.bulan in (${Month:sqlstring})
and c.plant_name in (${Plant:sqlstring})
and d.material_type_name in (${Material:sqlstring})
and e.jenis_mobil in (${Jenis_Mobil:sqlstring})
group by b.tanggal
order by b.tanggal;

```

**Gambar 21 Query Rata-Rata Lama Antri**

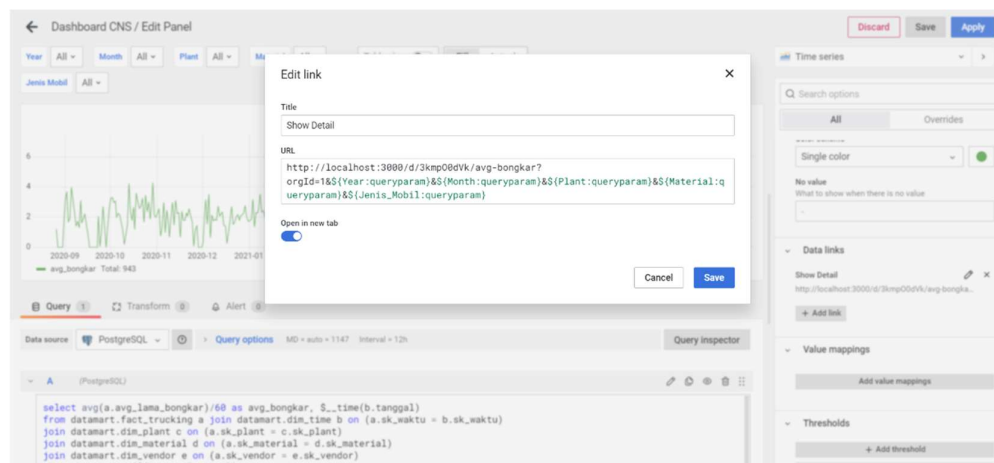
```

select avg(a.avg_qty_per_mobil) as QTY, $__time(b.tanggal)
from datamart.fact_trucking a join datamart.dim_time b on (a.sk_waktu =
b.sk_waktu)
join datamart.dim_plant c on (a.sk_plant = c.sk_plant)
join datamart.dim_material d on (a.sk_material = d.sk_material)
join datamart.dim_vendor e on (a.sk_vendor = e.sk_vendor)
where b.year in (${Year:sqlstring})
and b.bulan in (${Month:sqlstring})
and c.plant_name in (${Plant:sqlstring})
and d.material_type_name in (${Material:sqlstring})
and e.jenis_mobil in (${Jenis_Mobil:sqlstring})
group by b.tanggal
order by b.tanggal;

```

**Gambar 22** Query Rata-Rata *Quantity* Per Mobil

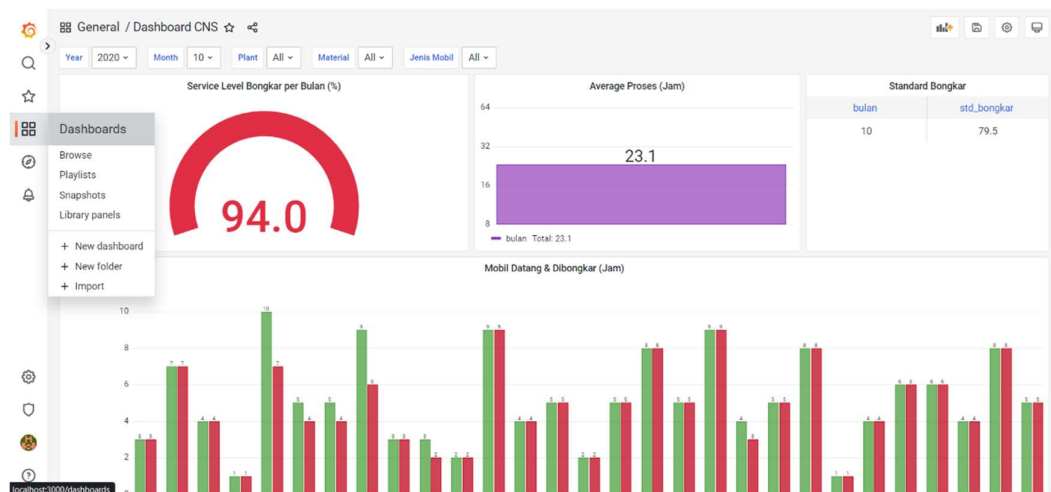
20. Kemudian untuk membuat fitur “Show Bongkar”, buat database baru dan membuat panel dalam bentuk tabel dengan query yang dapat dilihat pada **Gambar 20**.
21. **Gambar 23** menunjukkan penggunaan *data link* untuk menghubungkan database *show* bongkar dengan database CNS



**Gambar 23** Data Link Panel Rata-Rata Lama Bongkar

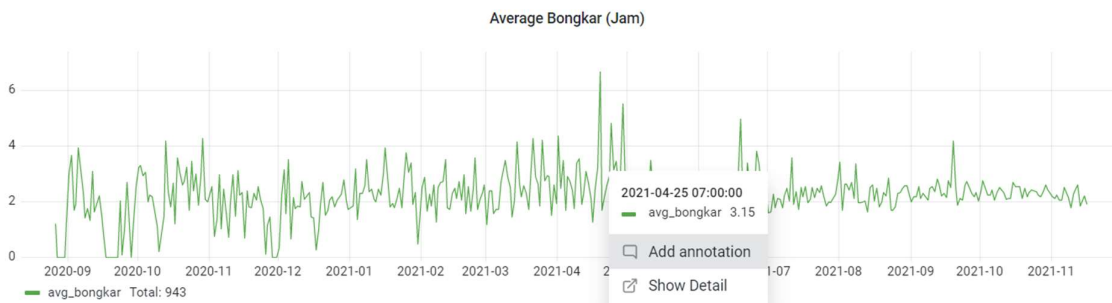
## PENGUNAAN DASHBOARD

User dapat mengakses *dashboard* pengawasan aktivitas truk pada PT. Cipta Niaga Semesta dengan membuka Grafana dalam web browser. User memasukkan <http://localhost:3000/> kemudian melakukan sign in, kemudian akan membuka halaman utama Grafana. Dari halaman utama, user dapat membuka *dashboards* dan memilih *dashboard* CNS. Halaman *dashboard* CNS dapat dilihat pada **Gambar 24**.



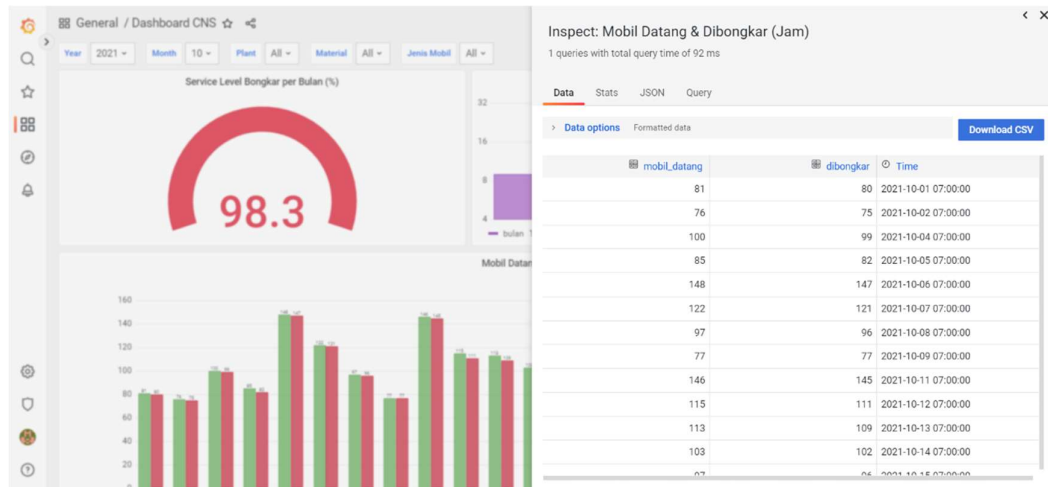
**Gambar 24** Halaman *Dashboard* CNS

Dalam halaman dashboard CNS dapat menggunakan *filter year, month, plant, material* dan jenis mobil untuk menampilkan data tertentu. **Gambar 25** menunjukkan cara membuka dashboard *show bongkar*.



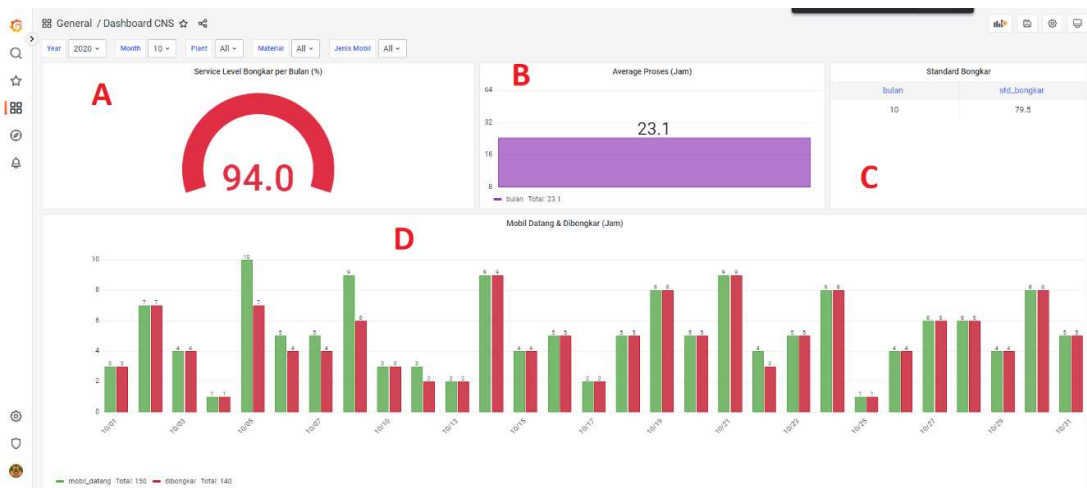
**Gambar 25** *Dashboard* CNS *Show Bongkar*

Jika *user* ingin mengunduh data, klik judul dengan data yang diinginkan dan klik *inspect*. Kemudian akan menampilkan halaman *inspect* data yang memiliki tombol “Download CSV” untuk mengunduh data dalam Microsoft Excel. Halaman *inspect* data dapat dilihat pada **Gambar 26**.



**Gambar 26** Halaman *Inspect* Data Dashboard

Berikut ini merupakan penjelasan gambar dari setiap *panel* dalam *dashboard* CNS:



**Gambar 27** Penjelasan *Panel* Atas



**Gambar 27** menunjukkan *dashboard* CNS bagian atas. Bagian A menunjukkan perhitungan persentase bongkar per bulan. Bagian B menunjukkan rata-rata lama proses dari truk masuk sampai keluar. Bagian C menunjukkan standard bongkar yang didapatkan dengan menghitung lama proses bulan lalu menjadi standard untuk bulan sekarang. Bagian D menunjukkan perbandingan mobil datang dibanding dengan barang yang dibongkar di hari yang sama.



**Gambar 28** Penjelasan *Panel* Bawah

**Gambar 28** menunjukkan *dashboard* CNS bagian bawah. Bagian E menunjukkan lama pending yang menghitung berapa banyak kendaraan yang masuk tetapi tidak melakukan pembongkaran. Bagian F menunjukkan rata-rata lama bongkar yaitu waktu antara mulai bongkar sampai selesai bongkar. Bagian G menunjukkan rata-rata lama atri yaitu waktu antara kendaraan masuk sampai mulai bongkar. Bagian H menunjukkan rata-rata kuantitas per mobil yang diangkut setiap hari.