

## UJIAN AKHIR SEMESTER GANJIL TAHUN AKADEMIK 2022/2023

(IF351-B) Database System – LEC

(IF351-BL) Database Systems – LAB

Ketua: Arfigo Ezra Pratama - 00000059747

Anggota:

- Wilcoustine Qhristmas Pniel Wijaya - 00000056960
- Nisa'ul Fadhilah - 00000060236
- Ayu Febriana Lingga - 00000057105

### Pembagian Tugas

1. SELECT Statement (Wilcoustine Qhristmas Pniel Wijaya)
2. Transaction Processing (Arfigo Ezra Pratama)
3. User Management, Privilege Management, & Views (Ayu Febriana Lingga)
4. Stored Routines & Triggers (Ayu Febriana Lingga)
5. Functional Dependencies & Normalization (Nisa'ul Fadhilah)

### Jawaban

LINK Google Drive :

[https://drive.google.com/drive/folders/1YFlwDuXdrGZ0fazbBq1yn6E9l1tEM\\_m1?usp=sharing](https://drive.google.com/drive/folders/1YFlwDuXdrGZ0fazbBq1yn6E9l1tEM_m1?usp=sharing)

1. SELECT Statement

```
// 1.
SELECT  CONCAT(c.Nama_Depan, ' ', c.Nama_Belakang) AS "Full Name",
o.Nama_Makanan AS "Food", CONCAT("Rp. ", m.Total_Harga) AS "Grand Total"
FROM membeli m
JOIN Customer c USING (ID_Customer)
JOIN outlet o USING (ID_Outlet);
```

```
MariaDB [gojek]> SELECT CONCAT(c.Nama_Depan, ' ', c.Nama_Belakang) AS "Full Name", o.Nama_Makanan AS "Food", CONCAT("Rp. ", m.Total_Harga) AS "Grand Total"
-> FROM membeli m
-> JOIN Customer c USING (ID_Customer)
-> JOIN outlet o USING (ID_Outlet);
+-----+-----+-----+
| Full Name | Food | Grand Total |
+-----+-----+-----+
| Ayu Komang | Kue Kering | Rp. 100000 |
| Adzka Iestari | Sushi | Rp. 50000 |
| Sari Hamidah | Nasi Rendang Mercon | Rp. 34000 |
| Vina Anastasia | Nasi Goreng Gila | Rp. 45000 |
| Ferry | Ayam Geprek | Rp. 60000 |
| Rahmat Zakaria | Cheese Thai Tea | Rp. 80000 |
| Mikhail Wijaya | Sate Kambing | Rp. 90000 |
| Ivan Maulana | Bubur Manado | Rp. 134000 |
| Rafael Pratama | Nasi Bento | Rp. 50000 |
| Indra Kurniawan | Ayam Geprek | Rp. 30000 |
+-----+-----+-----+
10 rows in set (0.001 sec)
```

Select Statement ini digunakan untuk melihat history para customer yang telah melakukan pembelian melalui aplikasi, select ini menunjukkan nama customer, makanan yang dibeli, dan total pembelanjaan.

```
// 2.
SELECT  CONCAT(d>Nama_Depan, ' ', d>Nama_Belakang) AS "Full Name",
k.Plat_Kendaraan AS "Plat Kendaraan"
FROM driver AS d
JOIN Kendaraan AS k USING (ID_Kendaraan);

MariaDB [gojek]> SELECT CONCAT(d>Nama_Depan, ' ', d>Nama_Belakang) AS "Full Name", k.Plat_Kendaraan AS "Plat Kendaraan"
-> FROM driver AS d
-> JOIN Kendaraan AS k USING (ID_Kendaraan);
+-----+-----+
| Full Name | Plat Kendaraan |
+-----+-----+
| Saiful Anwar | B 1978 RF |
| Yanti | B 9087 SXS |
| Rina Febrianti | B 5171 BXM |
| Adam Aditya | B 7177 VLP |
| Arif Ahmad | B 1971 PK |
| Rina Suryonegoro | B 9075 WBX |
| Dita Salsabila | B 6176 VXB |
| Rizky Wicaksana | B 1617 RDX |
| Sukirman | B 5015 XYM |
| Supardi | B 1971 SVU |
+-----+-----+
10 rows in set (0.002 sec)
```

Select statement ini digunakan untuk melihat para driver gojek dan plat kendaraan yang mereka gunakan saat pendaftaran gojek.

```
// 3.
SELECT  CONCAT(s>Nama_Depan, ' ', s>Nama_Belakang) AS "Full Name",
d>Nama_Divisi AS "Divisi"
FROM staff AS s
JOIN divisi AS d ON (s.ID_Divisi = d.ID_Divisi);

MariaDB [gojek]> SELECT CONCAT(s>Nama_Depan, ' ', s>Nama_Belakang) AS "Full Name", d>Nama_Divisi AS "Divisi"
-> FROM staff AS s
-> JOIN divisi AS d ON (s.ID_Divisi = d.ID_Divisi);
+-----+-----+
| Full Name | Divisi |
+-----+-----+
| Restu Saputra | Financial |
| Via Angelica | IT |
| Aldi Gunawan | IT |
| Riri Wahyuni | Visual |
| Ikbil Purwanto | Financial Manager |
| Yanto | Marketing |
| Rizky Maulana | Visual |
| Adi | Human Resource |
| Muhammad Zidan | Admin |
| Galang | Human Resource Manager |
+-----+-----+
10 rows in set (0.000 sec)
```

Select statement ini digunakan untuk melihat melihat para staff yang bekerja di gojek beserta divisi mereka.

```
// 4.
SELECT  CONCAT(d>Nama_Depan, ' ', d>Nama_Belakang) AS "Full Name",
td>Nama_Lengkap AS "Nama Tanggungan"
FROM driver AS d
LEFT JOIN Tanggungan_Driver AS td USING (ID_Driver)
ORDER BY 1
LIMIT 4;
```

```
MariaDB [gojek]> SELECT CONCAT(d>Nama_Depan, ' ', d>Nama_Belakang) AS "Full Name", td>Nama_Lengkap AS "Nama Tanggungan"
-> FROM driver AS d
-> LEFT JOIN Tanggungan_Driver AS td USING (ID_Driver)
-> ORDER BY 1
-> LIMIT 4;
+-----+-----+
| Full Name | Nama Tanggungan |
+-----+-----+
| Adam Aditya | Melinda Suryatmi |
| Arif Ahmad | Mahdi Siregar |
| Dita Salsabila | Gina Safitri |
| Rina Febrianti | Dimas Mandala |
+-----+-----+
4 rows in set (0.001 sec)
```

Select statement ini digunakan untuk melihat para staff gojek beserta tanggungan yang mereka miliki, dan tanggungan itu ditunjang oleh gojek.

```
// 5.
SELECT m.ID_Outlet AS "ID Outlet", o>Nama_Outlet AS "Outlet",
       CONCAT(c>Nama_Depan, ' ', c>Nama_Belakang) AS "Customer
Name", CONCAT("Rp. ", m.Total_Harga, ",-") AS "Total Price"
FROM membeli AS m
JOIN outlet AS o USING (ID_Outlet)
JOIN customer AS c USING (ID_Customer)
ORDER BY 1;
```

```
MariaDB [gojek]> SELECT m.ID_Outlet AS "ID Outlet", o>Nama_Outlet AS "Outlet",
->       CONCAT(c>Nama_Depan, ' ', c>Nama_Belakang) AS "Customer Name", CONCAT("Rp. ", m.Total_Harga, ",-") AS "Total Price"
-> FROM membeli AS m
-> JOIN outlet AS o USING (ID_Outlet)
-> JOIN customer AS c USING (ID_Customer)
-> ORDER BY 1;
```

ID Outlet	Outlet	Customer Name	Total Price
00001	Restauran Manado Thamrin	Ivan Maulana	Rp. 134000,-
00002	Toko Kue Nagamas	Ayu Komang	Rp. 100000,-
00003	Sushi Tokyo	Adzka Lestari	Rp. 50000,-
00004	Sate Pak Umar	Mikhail Wijaya	Rp. 90000,-
00005	Restoran Padang	Sari Hamidah	Rp. 34000,-
00006	Cheese Tokyo	Rahmat Zakaria	Rp. 80000,-
00007	Nasi Goreng Pak Ahmad	Vina Anastasia	Rp. 45000,-
00009	Bento Makusu	Rafael Pratama	Rp. 50000,-
00010	Chicken Wrecked By Tina	Ferry	Rp. 60000,-
00010	Chicken Wrecked By Tina	Indra Kurniawan	Rp. 30000,-

10 rows in set (0.001 sec)

Select statement ini digunakan untuk melihat data customer yang melakukan pembelian pada outlet-outlet kami.

## 2. Transaction Processing

Skenario:

Seorang database administrator ingin menambahkan data ke dalam tabel Partnership. Namun terjadi kesalahan penulisan pada nama perusahaan, sehingga perlu dilakukan rollback dan melakukan insert kembali.

Langkah-langkah proses TCL:

```
mysql> START TRANSACTION;
Query OK, 0 rows affected (0.01 sec)
```

## Memulai proses transaksi

```
mysql> SELECT * FROM Partnership;
```

Kode_Proposal	Nama_Perusahaan	Email	Nomor_Telepon	Surat_Kuasa	Kartu_Identitas	Dokumen_Izin_Us
aha	Nomor_Rekening	NPWP_Perusahaan	Alamat_Lengkap_Outlet			
P0001	PT Nainggolan Kuswoyo	Lidya87@gmail.com	0812-8535-5529	surat_kuasa_nk.jpg	ktp_lidya.jpg	nib_nk.jpg
	3013465277277681	npwp_nk.jpg	Jl. RE Martadinata No.10, Jakarta			
P0002	CV Mayasari	Tira.Hasanah@yahoo.com	0812-8190-5017	surat_kuasa_cv.jpg	ktp_hasanah.jpg	siup_cv.jpg
	6709527628762367	npwp_cv.jpg	Jl. Abd. Malik Pattana Endeng, Jakarta			
P0003	PT Nasyiah	Narpati.Rahmi@gmail.com	0852-5574-6705	surat_kuasa_nasyiah.jpg	ktp_narpati.jpg	tdup_nasyiah.jpg
	4844276388374767	npwp_nasyiah.jpg	Jl. HOS Cokroaminoto, Tangerang			
P0004	CV Namaga	Nababan.Martaka@gmail.com	0831-9275-3645	surat_kuasa_namaga.jpg	ktp_maria.jpg	tdy_namaga.jpg
	304665336263773	npwp_namaga.jpg	Jl. Wiru Indah No. 2, Tangerang			
P0005	PT Gemilang Bersama	Lailasari.Citra@yahoo.com	0831-7302-2168	surat_kuasa_gemilang.jpg	ktp_citra.jpg	tdup_gemilang.jpg
	374613586545965	npwp_gemilang.jpg	Jl. Cikini Raya No.117, Jakarta			
P0006	PT Tamba Najmudin	Dian85@yahoo.com	0838-1501-0826	surat_kuasa_tamba.jpg	ktp_dian.jpg	tdy_tamba.jpg
	5475462243556535	npwp_tamba.jpg	Jl. Perjuangan No.100, Tangerang			
P0007	PT Maju Sejahtera	Anggriawan@yahoo.com	0838-1575-0136	surat_kuasa_sejahtera.jpg	ktp_awan.jpg	nib_sejahtera.jpg
	6011997784546867	npwp_sejahtera.jpg	Jl. Raden Patah Kel No.39C, Tangerang			
P0008	CV Agustina Yulianti	Agustina@gmail.com	0895-1731-8260	surat_kuasa_agustina.jpg	ktp_tina.jpg	siup_agustina.jpg
	6706767433429447	npwp_agustina.jpg	Jl. Rawa Tembaga 1, Jakarta			
P0009	PT Usada	Riyanti.usada@gmail.com	0895-1797-2530	surat_kuasa_usada.jpg	ktp_riyanti.jpg	nib_usada.jpg
	5234158956226223	npwp_usada.jpg	Jl. Letjen Suprpto No.14, Jakarta			
P0010	CV Thamrin	Devi124@gmail.com	0831-1934-3150	surat_kuasa_thamrin.jpg	ktp_devi.jpg	nib_thamrin.jpg
	5641826489134814	npwp_thamrin.jpg	Jl. Kartika Raya, Tangerang			

10 rows in set (0.00 sec)

## Menampilkan tabel partnership

```
mysql> INSERT INTO Partnership (Kode_Proposal, Nama_Perusahaan, Email, Nomor_Telepon, Surat_Kuasa, Kartu_Identitas, Dokumen_Izin_Usaha, Nomor_Rekening, NPWP_Perusahaan, Alamat_Lengkap_Outlet)
-> VALUES ('P0018', 'PT Makmur Jaya', 'ilham27@yahoo.com', '0896-2901-7829', 'surat_kuasa_as.jpg', 'ktp_ilham.jpg', 'nib_ilham.jpg', '0596608412559734', 'npwp_ilham.jpg', 'Jl. Magelang No. 17, Mlati, Sleman, DI Yogyakarta');
Query OK, 1 row affected (0.02 sec)
```

Melakukan Insert ke tabel Partnership. Namun terdapat kesalahan pada nama perusahaan, yaitu PT Makmur Jaya yang seharusnya adalah PT Abadi Sejahtera

```
mysql> SELECT * FROM Partnership;
```

Kode_Proposal	Nama_Perusahaan	Email	Nomor_Telepon	Surat_Kuasa	Kartu_Identitas	Dokumen_Izin_Us
aha	Nomor_Rekening	NPWP_Perusahaan	Alamat_Lengkap_Outlet			
P0001	PT Nainggolan Kuswoyo	Lidya87@gmail.com	0812-8535-5529	surat_kuasa_nk.jpg	ktp_lidya.jpg	nib_nk.jpg
	3013465277277681	npwp_nk.jpg	Jl. RE Martadinata No.10, Jakarta			
P0002	CV Mayasari	Tira.Hasanah@yahoo.com	0812-8190-5017	surat_kuasa_cv.jpg	ktp_hasanah.jpg	siup_cv.jpg
	6709527628762367	npwp_cv.jpg	Jl. Abd. Malik Pattana Endeng, Jakarta			
P0003	PT Nasyiah	Narpati.Rahmi@gmail.com	0852-5574-6705	surat_kuasa_nasyiah.jpg	ktp_narpati.jpg	tdup_nasyiah.jpg
	4844276388374767	npwp_nasyiah.jpg	Jl. HOS Cokroaminoto, Tangerang			
P0004	CV Namaga	Nababan.Martaka@gmail.com	0831-9275-3645	surat_kuasa_namaga.jpg	ktp_maria.jpg	tdy_namaga.jpg
	304665336263773	npwp_namaga.jpg	Jl. Wiru Indah No. 2, Tangerang			
P0005	PT Gemilang Bersama	Lailasari.Citra@yahoo.com	0831-7302-2168	surat_kuasa_gemilang.jpg	ktp_citra.jpg	tdup_gemilang.jpg
	374613586545965	npwp_gemilang.jpg	Jl. Cikini Raya No.117, Jakarta			
P0006	PT Tamba Najmudin	Dian85@yahoo.com	0838-1501-0826	surat_kuasa_tamba.jpg	ktp_dian.jpg	tdy_tamba.jpg
	5475462243556535	npwp_tamba.jpg	Jl. Perjuangan No.100, Tangerang			
P0007	PT Maju Sejahtera	Anggriawan@yahoo.com	0838-1575-0136	surat_kuasa_sejahtera.jpg	ktp_awan.jpg	nib_sejahtera.jpg
	6011997784546867	npwp_sejahtera.jpg	Jl. Raden Patah Kel No.39C, Tangerang			
P0008	CV Agustina Yulianti	Agustina@gmail.com	0895-1731-8260	surat_kuasa_agustina.jpg	ktp_tina.jpg	siup_agustina.jpg
	6706767433429447	npwp_agustina.jpg	Jl. Rawa Tembaga 1, Jakarta			
P0009	PT Usada	Riyanti.usada@gmail.com	0895-1797-2530	surat_kuasa_usada.jpg	ktp_riyanti.jpg	nib_usada.jpg
	5234158956226223	npwp_usada.jpg	Jl. Letjen Suprpto No.14, Jakarta			
P0010	CV Thamrin	Devi124@gmail.com	0831-1934-3150	surat_kuasa_thamrin.jpg	ktp_devi.jpg	nib_thamrin.jpg
	5641826489134814	npwp_thamrin.jpg	Jl. Kartika Raya, Tangerang			
P0018	PT Makmur Jaya	ilham27@yahoo.com	0896-2901-7829	surat_kuasa_as.jpg	ktp_ilham.jpg	nib_ilham.jpg
	0596608412559734	npwp_ilham.jpg	Jl. Magelang No. 17, Mlati, Sleman, DI Yogyakarta			

11 rows in set (0.01 sec)

## Menampilkan tabel Partnership setelah dilakukan Insert

```
mysql> ROLLBACK;
Query OK, 0 rows affected (0.01 sec)
```

## Melakukan Rollback untuk membatalkan Insert

```
mysql> SELECT * FROM Partnership;
```

Kode_Proposal	Nama_Perusahaan	Email	Nomor_Telepon	Surat_Kuasa	Kartu_Identitas	Dokumen_Izin_Us
aha	Nomor_Rekening	NPWP_Perusahaan	Alamat_Lengkap_Outlet			
P0001	PT Nainggolan Kuswoyo	Lidya87@gmail.com	0812-8535-5529	surat_kuasa_nk.jpg	ktp_lidya.jpg	nib_nk.jpg
P0002	CV Mayasari	Tira.Hasanah@yahoo.com	0812-8190-5017	surat_kuasa_cv.jpg	ktp_hasanah.jpg	siup_cv.jpg
P0003	PT Nasyiah	Narpati.Rahmi@gmail.com	0852-5574-6705	surat_kuasa_nasyiah.jpg	ktp_narpati.jpg	tdup_nasyiah.jpg
P0004	CV Namaga	Nababan.Martaka@gmail.com	0831-9275-3645	surat_kuasa_namaga.jpg	ktp_maria.jpg	tdy_namaga.jpg
P0005	PT Gemilang Bersama	Lailasari.Citra@yahoo.com	0831-7302-2168	surat_kuasa_gemilang.jpg	ktp_citra.jpg	tdup_gemilang.jpg
P0006	PT Tamba Najmudin	Dian85@yahoo.com	0838-1501-0826	surat_kuasa_tamba.jpg	ktp_dian.jpg	tdy_tamba.jpg
P0007	PT Maju Sejahtera	Anggriawan@yahoo.com	0838-1575-0136	surat_kuasa_sejahtera.jpg	ktp_awan.jpg	nib_sejahtera.jpg
P0008	CV Agustina Yulianti	Agustina@gmail.com	0895-1731-8260	surat_kuasa_agustina.jpg	ktp_tina.jpg	siup_agustina.jpg
P0009	PT Usada	Riyanti.usada@gmail.com	0895-1797-2530	surat_kuasa_usada.jpg	ktp_riyanti.jpg	nib_usada.jpg
P0010	CV Thamrin	Devi124@gmail.com	0831-1934-3150	surat_kuasa_thamrin.jpg	ktp_devi.jpg	nib_tramrin.jpg

10 rows in set (0.00 sec)

## Menampilkan tabel Partnership setelah dilakukan Rollback

```
mysql> INSERT INTO Partnership (Kode_Proposal, Nama_Perusahaan, Email, Nomor_Telepon, Surat_Kuasa, Kartu_Identitas, Dokumen_Izin_Usaha, Nomor_Rekening, NPWP_Perusahaan, Alamat_Lengkap_Outlet)
-> VALUES ('P0018', 'PT Abadi Sejahtera', 'ilham27@yahoo.com', '0896-2901-7829', 'surat_kuasa_as.jpg', 'ktp_ilham.jpg', 'nib_ilham.jpg', '0596608412559734', 'npwp_ilham.jpg', 'Jl. Magelang No. 17, Mlati, Sleman, DI Yogyakarta');
Query OK, 1 row affected (0.02 sec)
```

## Melakukan Insert data kembali

```
mysql> SELECT * FROM Partnership;
```

Kode_Proposal	Nama_Perusahaan	Email	Nomor_Telepon	Surat_Kuasa	Kartu_Identitas	Dokumen_Izin_Us
aha	Nomor_Rekening	NPWP_Perusahaan	Alamat_Lengkap_Outlet			
P0001	PT Nainggolan Kuswoyo	Lidya87@gmail.com	0812-8535-5529	surat_kuasa_nk.jpg	ktp_lidya.jpg	nib_nk.jpg
P0002	CV Mayasari	Tira.Hasanah@yahoo.com	0812-8190-5017	surat_kuasa_cv.jpg	ktp_hasanah.jpg	siup_cv.jpg
P0003	PT Nasyiah	Narpati.Rahmi@gmail.com	0852-5574-6705	surat_kuasa_nasyiah.jpg	ktp_narpati.jpg	tdup_nasyiah.jpg
P0004	CV Namaga	Nababan.Martaka@gmail.com	0831-9275-3645	surat_kuasa_namaga.jpg	ktp_maria.jpg	tdy_namaga.jpg
P0005	PT Gemilang Bersama	Lailasari.Citra@yahoo.com	0831-7302-2168	surat_kuasa_gemilang.jpg	ktp_citra.jpg	tdup_gemilang.jpg
P0006	PT Tamba Najmudin	Dian85@yahoo.com	0838-1501-0826	surat_kuasa_tamba.jpg	ktp_dian.jpg	tdy_tamba.jpg
P0007	PT Maju Sejahtera	Anggriawan@yahoo.com	0838-1575-0136	surat_kuasa_sejahtera.jpg	ktp_awan.jpg	nib_sejahtera.jpg
P0008	CV Agustina Yulianti	Agustina@gmail.com	0895-1731-8260	surat_kuasa_agustina.jpg	ktp_tina.jpg	siup_agustina.jpg
P0009	PT Usada	Riyanti.usada@gmail.com	0895-1797-2530	surat_kuasa_usada.jpg	ktp_riyanti.jpg	nib_usada.jpg
P0010	CV Thamrin	Devi124@gmail.com	0831-1934-3150	surat_kuasa_thamrin.jpg	ktp_devi.jpg	nib_tramrin.jpg
P0018	PT Abadi Sejahtera	ilham27@yahoo.com	0896-2901-7829	surat_kuasa_as.jpg	ktp_ilham.jpg	nib_ilham.jpg

11 rows in set (0.00 sec)

## Tampilan tabel Partnership setelah dilakukan insert kembali

```
mysql> COMMIT;
Query OK, 0 rows affected (0.00 sec)
```

## Melakukan Commit untuk menyimpan transaksi

### 3. User Management, Privilege Management, & Views

#### A. User

```
// Super User
CREATE USER 'Galang'@'localhost'
IDENTIFIED BY 'galang10';

// User
CREATE USER 'Ikbal'@'localhost'
IDENTIFIED BY 'ikpurwanto';

// Restricted User
CREATE USER 'Restu'@'localhost'
IDENTIFIED BY 'resaputra';

MariaDB [gojek]> CREATE USER 'Galang'@'localhost'
-> IDENTIFIED BY 'galang10';
Query OK, 0 rows affected (0.001 sec)

MariaDB [gojek]> CREATE USER 'Ikbal'@'localhost'
-> IDENTIFIED BY 'ikpurwanto';
Query OK, 0 rows affected (0.001 sec)

MariaDB [gojek]> CREATE USER 'Restu'@'localhost'
-> IDENTIFIED BY 'resaputra';
Query OK, 0 rows affected (0.002 sec)
```

Fungsi dari user Galang adalah sebagai super user yang memiliki kendali untuk melakukan update, dan select terhadap suatu view dan memberikan akses kepada orang lain dalam perusahaan tersebut. Maka, apabila terdapat data yang ingin di update, user Galang dapat menggunakan view tersebut dan hasil dari update tersebut masuk kedalam database.

Fungsi dari user Ikbal adalah sebagai user yang memiliki kendali untuk melakukan select terhadap suatu view. Selain itu, user Ikbal dapat memberikan akses kepada orang yang berada di divisi yang sama dengan Ikbal.

Fungsi dari user Restu adalah sebagai restricted user. Maksud dari restricted user adalah user yang memiliki keterbatasan dalam menggunakan view yang diberikan sehingga user ini tidak dapat melakukan pemberian akses kepada orang lain ataupun melakukan insert dan update.

## B. View

```
CREATE OR REPLACE VIEW grand_total_sales
AS SELECT o.ID_Outlet AS `ID Outlet`, o>Nama_Outlet AS `Nama Outlet`,
CONCAT("Rp. ", SUM(m.Total_Harga), ",-") AS `Total Harga`
  FROM outlet o
  JOIN membeli m USING (ID_Outlet)
  GROUP BY o>Nama_Outlet
  ORDER BY 1;

CREATE OR REPLACE VIEW data_staff
AS SELECT s.ID_Staff AS `ID Staff`, s>Nama_Depan AS `Nama Depan`,
s>Nama_Belakang AS `Nama Belakang`, d.ID_Divisi AS `ID Divisi`
  FROM staff s
  JOIN divisi d ON(s.ID_Divisi = d.ID_Divisi);

CREATE OR REPLACE VIEW kinerja_admin
AS SELECT a.ID_Admin AS `ID Admin`, CONCAT(a>Nama_Depan, ' ',
a>Nama_Belakang) AS `Nama`, COUNT(*) AS `Jumlah Customer`
  FROM admin a
  JOIN customer c USING(ID_Admin)
  GROUP BY 1
  ORDER BY 1;
```

```
MariaDB [gojek]> CREATE OR REPLACE VIEW grand_total_sales
-> AS SELECT o.ID_Outlet AS `ID Outlet`, o>Nama_Outlet AS `Nama Outlet`, CONCAT("Rp. ", SUM(m.Total_Harga), ",-") AS `Total Harga`
-> FROM outlet o
-> JOIN membeli m USING (ID_Outlet)
-> GROUP BY o>Nama_Outlet
-> ORDER BY 1;
Query OK, 0 rows affected (0.019 sec)

MariaDB [gojek]> CREATE OR REPLACE VIEW data_staff
-> AS SELECT s.ID_Staff AS `ID Staff`, s>Nama_Depan AS `Nama Depan`, s>Nama_Belakang AS `Nama Belakang`, d.ID_Divisi AS `ID Divisi`
-> FROM staff s
-> JOIN divisi d ON(s.ID_Divisi = d.ID_Divisi);
Query OK, 0 rows affected (0.004 sec)

MariaDB [gojek]> CREATE OR REPLACE VIEW kinerja_admin
-> AS SELECT a.ID_Admin AS `ID Admin`, CONCAT(a>Nama_Depan, ' ', a>Nama_Belakang) AS `Nama`, COUNT(*) AS `Jumlah Customer`
-> FROM admin a
-> JOIN customer c USING(ID_Admin)
-> GROUP BY 1
-> ORDER BY 1;
Query OK, 0 rows affected (0.004 sec)
```

View `grand_total_sales` digunakan untuk melihat total pendapatan yang didapatkan oleh setiap outlet yang tergabung di dalam gojek. Dalam view tersebut berisikan id outlet, nama outlet dan total pendapatan dari setiap outlet.

View `data_staff` digunakan untuk melihat data-data dari staff yang bekerja di dalam gojek. Dalam view tersebut berisikan id staff, nama depan staff, nama belakang staff, dan id divisi.

View `kinerja_admin` digunakan untuk melihat berapa customer yang dapat dihandle oleh admin yang bekerja di dalam gojek. Dalam view tersebut berisikan id admin, nama panjang admin dan jumlah customer yang di handle.

### C. Privilege

```
// User
GRANT SELECT
ON grand_total_sales
TO 'Ikbal'@'localhost'
WITH GRANT OPTION;
```

```
MariaDB [gojek]> GRANT SELECT
-> ON grand_total_sales
-> TO 'Ikbal'@'localhost'
-> WITH GRANT OPTION;
Query OK, 0 rows affected (0.002 sec)
```

Hasil :

```
C:\xampp\mysql\bin>mysql -h localhost -u Ikbal -p
Enter password: *****
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 321
Server version: 10.4.24-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> USE gojek;
Database changed
MariaDB [gojek]> SELECT * FROM grand_total_sales;
+-----+-----+-----+
| ID Outlet | Nama Outlet | Total Harga |
+-----+-----+-----+
| 00001 | Restoran Manado Thamrin | Rp. 134000,- |
| 00002 | Toko Kue Nagamas | Rp. 100000,- |
| 00003 | Sushi Tokyo | Rp. 50000,- |
| 00004 | Sate Pak Umar | Rp. 90000,- |
| 00005 | Restoran Padang | Rp. 34000,- |
| 00006 | Cheese Tokyo | Rp. 80000,- |
| 00007 | Nasi Goreng Pak Ahmad | Rp. 45000,- |
| 00009 | Bento Makusu | Rp. 50000,- |
| 00010 | Chicken Wrecked By Tina | Rp. 90000,- |
+-----+-----+-----+
9 rows in set (0.004 sec)
```

Fungsi dari privilege tersebut adalah user Ikbal dapat melakukan select terhadap view grand\_total\_sales serta dapat melihat id outlet, nama outlet, serta total pendapatan dari outlet tersebut. User Ikbal tidak dapat melakukan update atau delete karena tidak diberikan privilege untuk hal tersebut. Namun, user Ikbal dapat memberikan akses kepada user Restu untuk melakukan select pada view grand\_total\_sales.

```
// Super User
GRANT SELECT
ON kinerja_admin
TO 'Galang'@'localhost'
WITH GRANT OPTION;
```



```

MariaDB [gojek]> GRANT SELECT
-> ON kinerja_admin
-> TO 'Galang'@'localhost'
-> WITH GRANT OPTION;
Query OK, 0 rows affected (0.002 sec)

```

Hasil :

```

C:\xampp\mysql\bin>mysql -h localhost -u Galang -p
Enter password: *****
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 322
Server version: 10.4.24-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> USE gojek;
Database changed
MariaDB [gojek]> SELECT * FROM kinerja_admin;
+-----+-----+-----+
| ID Admin | Nama                | Jumlah Customer |
+-----+-----+-----+
| A0001    | Tina Pramudita      | 1                |
| A0002    | Nadya Anggraeni     | 1                |
| A0003    | Rona Wantari        | 1                |
| A0004    | Adelina             | 1                |
| A0005    | Alicia Chaniago     | 1                |
| A0006    | Elsa Arina          | 1                |
| A0007    | Indah               | 1                |
| A0008    | Nabila Cindi        | 1                |
| A0009    | Amanda Khairunisa   | 1                |
| A0010    | Jennifer Laurence   | 1                |
+-----+-----+-----+
10 rows in set (0.009 sec)

```

Fungsi dari privilege ini, user Galang dapat melakukan select terhadap view kinerja\_admin serta dapat melihat id admin, nama panjang admin dan jumlah customer yang dapat dihandle. Selain itu, user Galang dapat memberikan akses view tersebut kepada orang lain.

```

// Super User
GRANT UPDATE
ON data_staff
TO 'Galang'@'localhost';

GRANT SELECT
ON data_staff
TO 'Galang'@'localhost'
WITH GRANT OPTION;

```

```

MariaDB [gojek]> GRANT UPDATE
-> ON data_staff
-> TO 'Galang'@'localhost';
Query OK, 0 rows affected (0.005 sec)

MariaDB [gojek]> GRANT SELECT
-> ON data_staff
-> TO 'Galang'@'localhost'
-> WITH GRANT OPTION;
Query OK, 0 rows affected (0.003 sec)

```

Hasil :

```
MariaDB [gojek]> SELECT * FROM data_staff;
```

ID Staff	Nama Depan	Nama Belakang	ID Divisi
S0001	Restu	Saputra	V0001
S0002	Via	Angelica	V0002
S0003	Aldi	Gunawan	V0002
S0004	Riri	Wahyuni	V0008
S0005	Ikbal	Purwanto	V0006
S0006	Yanto		V0007
S0007	Rizky	Maulana	V0008
S0008	Adi		V0003
S0009	Muhammad	Zidan	V0010
S0010	Galang		V0004

```
10 rows in set (0.003 sec)
```

```
MariaDB [gojek]> UPDATE data_staff
-> SET `Nama Depan` = 'Arya'
-> WHERE `ID Staff` = 'S0001';
Query OK, 1 row affected (0.011 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [gojek]> SELECT * FROM data_staff;
```

ID Staff	Nama Depan	Nama Belakang	ID Divisi
S0001	Arya	Saputra	V0001
S0002	Via	Angelica	V0002
S0003	Aldi	Gunawan	V0002
S0004	Riri	Wahyuni	V0008
S0005	Ikbal	Purwanto	V0006
S0006	Yanto		V0007
S0007	Rizky	Maulana	V0008
S0008	Adi		V0003
S0009	Muhammad	Zidan	V0010
S0010	Galang		V0004

```
10 rows in set (0.001 sec)
```

```
MariaDB [gojek]> SELECT * FROM staff;
```

ID_Staff	Nama_Depan	Nama_Belakang	NIK	Jabatan	Gaji	Email	ID_Divisi
S0001	Arya	Saputra	3677908745631209	Accountant	1500000	saputra.restu@yahoo.com	V0001
S0002	Via	Angelica	7496266975702119	Software Engineer	8500000	via.angelica@gmail.com	V0002
S0003	Aldi	Gunawan	4085802776564422	Back End Developer	8500000	aldi.gunawan@rocketmail.com	V0002
S0004	Riri	Wahyuni	4429922581914265	UI/UX Manager	5500000	wahyuni.riri@yahoo.com	V0008
S0005	Ikbal	Purwanto	9949337131803594	Accountant Manager	3500000	ikbal124@hotmail.com	V0006
S0006	Yanto		1021971179451670	Marketing	1500000	yanto@yahoo.com	V0007
S0007	Rizky	Maulana	4301945896763910	Front End Developer	8500000	rizkyHln@gmail.com	V0008
S0008	Adi		2078304009826382	HRD	4500000	Adi@yahoo.com	V0003
S0009	Muhammad	Zidan	2131002503627952	Marketing Manager	3500000	muhammad.zidan@hotmail.com	V0010
S0010	Galang		3020092878268848	HR Manager	5500000	galang@yahoo.com	V0004

```
10 rows in set (0.000 sec)
```

Fungsi dari privilege ini adalah, karena user Galang adalah seorang Super User, maka ia mendapatkan akses untuk melakukan update dan select terhadap view data\_staff. Kemudian, apabila kita melakukan pengecekan terhadap tabel

staff melalui root, maka data yang tadi di update akan masuk ke dalam tabel tersebut.

```
// Restricted User
GRANT SELECT
ON grand_total_sales
TO 'Restu'@'localhost';

MariaDB [gojek]> GRANT SELECT
-> ON grand_total_sales
-> TO 'Restu'@'localhost';
Query OK, 0 rows affected (0.002 sec)
```

Hasil :

```
C:\xampp\mysql\bin>mysql -h localhost -u Restu -p
Enter password: *****
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 324
Server version: 10.4.24-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> USE gojek;
Database changed
MariaDB [gojek]> SELECT * FROM grand_total_sales;
+-----+-----+-----+
| ID Outlet | Nama Outlet | Total Harga |
+-----+-----+-----+
| 00001 | Restoran Manado Thamrin | Rp. 134000,- |
| 00002 | Toko Kue Nagamas | Rp. 100000,- |
| 00003 | Sushi Tokyo | Rp. 50000,- |
| 00004 | Sate Pak Umar | Rp. 90000,- |
| 00005 | Restoran Padang | Rp. 34000,- |
| 00006 | Cheese Tokyo | Rp. 80000,- |
| 00007 | Nasi Goreng Pak Ahmad | Rp. 45000,- |
| 00009 | Bento Makusu | Rp. 50000,- |
| 00010 | Chicken Wrecked By Tina | Rp. 90000,- |
+-----+-----+-----+
9 rows in set (0.002 sec)
```

Fungsi dari privilege ini adalah untuk melakukan select terhadap view grand\_total\_sales. Kemudian, karena user Restu adalah restricted user, maka ia tidak dapat melakukan update ataupun memberikan akses dari privilege yang dimiliki kepada orang lain.

#### 4. Stored Routines & Triggers

##### A. Stored Procedure

```
// 1.
DELIMITER //
CREATE PROCEDURE search_driver(IN name VARCHAR(100))
BEGIN
    DECLARE sdFound INTEGER;
    CREATE TEMPORARY TABLE sdView (`Hasil` VARCHAR(100));
```

```

SET sdFound =
(
    SELECT COUNT(*)
    FROM driver
    WHERE Nama_Depan LIKE CONCAT('%', name, '%') OR Nama_Belakang
LIKE CONCAT('%', name, '%')
);

IF sdFound < 1 THEN
    INSERT INTO sdView VALUES ("Name Invalid! Please Try Again");
    SELECT * FROM sdView;

ELSE
    INSERT INTO sdView VALUES ("Data Found!");

    INSERT INTO sdView
    SELECT CONCAT("Driver ID      : ", ID_Driver)
    FROM driver
    WHERE Nama_Depan LIKE CONCAT('%', name, '%') OR Nama_Belakang
LIKE CONCAT('%', name, '%');

    INSERT INTO sdView
    SELECT CONCAT("Full Name      : ", Nama_Depan, ' ',
Nama_Belakang)
    FROM driver
    WHERE Nama_Depan LIKE CONCAT('%', name, '%') OR Nama_Belakang
LIKE CONCAT('%', name, '%');

    INSERT INTO sdView
    SELECT CONCAT("Email          : ", Email)
    FROM driver
    WHERE Nama_Depan LIKE CONCAT('%', name, '%') OR Nama_Belakang
LIKE CONCAT('%', name, '%');

    INSERT INTO sdView
    SELECT CONCAT("Vehicle ID    : ", ID_Kendaraan)
    FROM driver
    WHERE Nama_Depan LIKE CONCAT('%', name, '%') OR Nama_Belakang
LIKE CONCAT('%', name, '%');

    INSERT INTO sdView
    SELECT CONCAT("Vehicle Type  : ", Jenis_Kendaraan)
    FROM driver
    WHERE Nama_Depan LIKE CONCAT('%', name, '%') OR Nama_Belakang
LIKE CONCAT('%', name, '%');

    SELECT * FROM sdView;

END IF;

DROP TEMPORARY TABLE sdView;

END//

```

```

MariaDB [gojek]> DELIMITER //
MariaDB [gojek]> CREATE PROCEDURE search_driver(IN name VARCHAR(100))
-> BEGIN
->     DECLARE sdFound INTEGER;
->     CREATE TEMPORARY TABLE sdView ('Hasil' VARCHAR(100));
->     SET sdFound =
->     (
->         SELECT COUNT(*)
->         FROM driver
->         WHERE Nama_Depan LIKE CONCAT('%', name, '%') OR Nama_Belakang LIKE CONCAT('%', name, '%')
->     );
->     IF sdFound < 1 THEN
->         INSERT INTO sdView VALUES ("Name Invalid! Please Try Again");
->         SELECT * FROM sdView;
->     ELSE
->         INSERT INTO sdView VALUES ("Data Found!");
->         INSERT INTO sdView
->         SELECT CONCAT("Driver ID      : ", ID_Driver)
->         FROM driver
->         WHERE Nama_Depan LIKE CONCAT('%', name, '%') OR Nama_Belakang LIKE CONCAT('%', name, '%');
->         INSERT INTO sdView
->         SELECT CONCAT("Full Name      : ", Nama_Depan, ' ', Nama_Belakang)
->         FROM driver
->         WHERE Nama_Depan LIKE CONCAT('%', name, '%') OR Nama_Belakang LIKE CONCAT('%', name, '%');
->         INSERT INTO sdView
->         SELECT CONCAT("Email          : ", Email)
->         FROM driver
->         WHERE Nama_Depan LIKE CONCAT('%', name, '%') OR Nama_Belakang LIKE CONCAT('%', name, '%');
->         INSERT INTO sdView
->         SELECT CONCAT("Vehicle ID    : ", ID_Kendaraan)
->         FROM driver
->         WHERE Nama_Depan LIKE CONCAT('%', name, '%') OR Nama_Belakang LIKE CONCAT('%', name, '%');
->         INSERT INTO sdView
->         SELECT CONCAT("Vehicle Type  : ", Jenis_Kendaraan)
->         FROM driver
->         WHERE Nama_Depan LIKE CONCAT('%', name, '%') OR Nama_Belakang LIKE CONCAT('%', name, '%');
->         SELECT * FROM sdView;
->     END IF;
->     DROP TEMPORARY TABLE sdView;
-> END//
Query OK, 0 rows affected (0.005 sec)

```

Hasil :

```

MariaDB [gojek]> DELIMITER ;
MariaDB [gojek]> CALL search_driver('yanti');
+-----+
| Hasil |
+-----+
| Data Found! |
| Driver ID      : D0002 |
| Full Name      : Yanti |
| Email          : yanti@yahoo.com |
| Vehicle ID     : K0003 |
| Vehicle Type   : Motor |
+-----+
6 rows in set (0.005 sec)

Query OK, 6 rows affected (0.028 sec)

MariaDB [gojek]> CALL search_driver('ayu');
+-----+
| Hasil |
+-----+
| Name Invalid! Please Try Again |
+-----+
1 row in set (0.002 sec)

Query OK, 1 row affected (0.010 sec)

MariaDB [gojek]> CALL search_driver('febrianti');
+-----+
| Hasil |
+-----+
| Data Found! |
| Driver ID      : D0003 |
| Full Name      : Rina Febrianti |
| Email          : rina.febrianti@gmail.com |
| Vehicle ID     : K0007 |
| Vehicle Type   : Motor |
+-----+
6 rows in set (0.004 sec)

Query OK, 6 rows affected (0.022 sec)

```

Stored procedure diatas digunakan untuk mencari data driver berdasarkan pada nama depan ataupun nama belakang. Output yang ditampilkan adalah id driver, nama panjang driver, email driver, id kendaraan dan tipe kendaraan driver.

```
// 2.
DELIMITER //
CREATE PROCEDURE search_customer (IN cid VARCHAR(100))
BEGIN
    DECLARE scCount INTEGER;
    CREATE TEMPORARY TABLE scView (`Hasil Pencarian` VARCHAR(100));
    SET scCount =
    (
        SELECT COUNT(*)
        FROM Customer
        WHERE ID_Customer LIKE cid
    );

    IF scCount < 1 THEN
        INSERT INTO scView VALUES ("ID Not Found! Please Try Again");
        SELECT * FROM scView;

    ELSE
        INSERT INTO scView VALUES ("Data Found!");

        INSERT INTO scView
        SELECT CONCAT("Customer ID : ", ID_Customer)
        FROM customer
        WHERE ID_Customer LIKE cid;

        INSERT INTO scView
        SELECT CONCAT("Full Name : ", Nama_Depan, ' ',
Nama_Belakang)
        FROM customer
        WHERE ID_Customer LIKE cid;

        INSERT INTO scView
        SELECT CONCAT("Phone Number : ", Nomor_Telepon)
        FROM customer
        WHERE ID_Customer LIKE cid;

        INSERT INTO scView
        SELECT CONCAT("Service Type : ", Jenis_Layanan)
        FROM layanan
        WHERE ID_Customer LIKE cid;

        INSERT INTO scView
        SELECT CONCAT("Total Price : Rp. ", Total_Harga)
```

```

        FROM membeli
        WHERE ID_Customer LIKE cid;

        SELECT * FROM scView;
    END IF;
    DROP TEMPORARY TABLE scView;
END//

```

```

MariaDB [gojek]> DELIMITER //
MariaDB [gojek]> CREATE PROCEDURE search_customer (IN cid VARCHAR(100))
-> BEGIN
->     DECLARE scCount INTEGER;
->     CREATE TEMPORARY TABLE scView ('Hasil Pencarian' VARCHAR(100));
->     SET scCount =
->     (
->         SELECT COUNT(*)
->         FROM Customer
->         WHERE ID_Customer LIKE cid
->     );
->     IF scCount < 1 THEN
->         INSERT INTO scView VALUES ("ID Not Found! Please Try Again");
->         SELECT * FROM scView;
->     ELSE
->         INSERT INTO scView VALUES ("Data Found!");
->         INSERT INTO scView
->         SELECT CONCAT("Customer ID : ", ID_Customer)
->         FROM customer
->         WHERE ID_Customer LIKE cid;
->         INSERT INTO scView
->         SELECT CONCAT("Full Name : ", Nama_Depan, ' ', Nama_Belakang)
->         FROM customer
->         WHERE ID_Customer LIKE cid;
->         INSERT INTO scView
->         SELECT CONCAT("Phone Number : ", Nomor_Telepon)
->         FROM customer
->         WHERE ID_Customer LIKE cid;
->         INSERT INTO scView
->         SELECT CONCAT("Service Type : ", Jenis_Layanan)
->         FROM layanan
->         WHERE ID_Customer LIKE cid;
->         INSERT INTO scView
->         SELECT CONCAT("Total Price : Rp. ", Total_Harga)
->         FROM membeli
->         WHERE ID_Customer LIKE cid;
->         SELECT * FROM scView;
->     END IF;
->     DROP TEMPORARY TABLE scView;
-> END//
Query OK, 0 rows affected (0.005 sec)

```

Hasil :

```

MariaDB [gojek]> DELIMITER ;
MariaDB [gojek]> CALL search_customer('C0001');
+-----+
| Hasil Pencarian |
+-----+
| Data Found!     |
| Customer ID   : C0001 |
| Full Name    : Ayu Komang |
| Phone Number  : 0831-5127-6945 |
| Service Type  : GoFood |
| Total Price   : Rp. 100000 |
+-----+
6 rows in set (0.003 sec)

Query OK, 6 rows affected (0.025 sec)

MariaDB [gojek]> CALL search_customer('C0050');
+-----+
| Hasil Pencarian |
+-----+
| ID Not Found! Please Try Again |
+-----+
1 row in set (0.003 sec)

```

Stored procedure diatas digunakan untuk mencari data customer berdasarkan pada id customer. Output yang ditampilkan adalah id customer, nama panjang customer, nomor telepon customer, tipe layanan gojek yang digunakan customer, dan total harga yang dibayar oleh customer.

## B. Stored Function

```
// 1.
DELIMITER //
CREATE FUNCTION email_admin(fname VARCHAR(100), lname VARCHAR(100))
RETURNS VARCHAR(100)
BEGIN
    RETURN CONCAT(fname, '.', lname, '@gojekmail.com');
END//
```

```
MariaDB [gojek]> DELIMITER //
MariaDB [gojek]> CREATE FUNCTION email_admin(fname VARCHAR(100), lname VARCHAR(100))
-> RETURNS VARCHAR(100)
-> BEGIN
->     RETURN CONCAT(fname, '.', lname, '@gojekmail.com');
-> END//
Query OK, 0 rows affected (0.004 sec)
```

Hasil :

```
MariaDB [gojek]> DELIMITER ;
MariaDB [gojek]> SELECT CONCAT>Nama_Depan, ' ', Nama_Belakang) AS 'Full Name', email_admin>Nama_Depan, Nama_Belakang) AS 'Email'
-> FROM admin
-> WHERE Nama_Depan LIKE 'A%'
-> ORDER BY 1;
+-----+-----+
| Full Name | Email |
+-----+-----+
| Adelina   | Adelina.@gojekmail.com |
| Alicia Chaniago | Alicia.Chaniago@gojekmail.com |
| Amanda Khairunisa | Amanda.Khairunisa@gojekmail.com |
+-----+-----+
3 rows in set (0.001 sec)

MariaDB [gojek]> SELECT CONCAT>Nama_Depan, ' ', Nama_Belakang) AS 'Full Name', email_admin>Nama_Depan, Nama_Belakang) AS 'Email'
-> FROM admin
-> WHERE Nama_Depan LIKE 'N%'
-> LIMIT 1;
+-----+-----+
| Full Name | Email |
+-----+-----+
| Nadya Anggraeni | Nadya.Anggraeni@gojekmail.com |
+-----+-----+
1 row in set (0.000 sec)
```

Stored function diatas digunakan untuk memunculkan email dari admin gojek apabila email mereka diganti menjadi [nama\\_depan.nama\\_belakang@gojekmail.com](mailto:nama_depan.nama_belakang@gojekmail.com). Output yang muncul apabila stored function tersebut dijalankan adalah nama panjang admin, dan email baru dengan tambahan @gojekmail.com.

```
// 2.
DELIMITER //
CREATE FUNCTION Customer_Voucher(Total_Harga INTEGER)
RETURNS VARCHAR(100)
BEGIN
    DECLARE cust_voucher VARCHAR(100);
```



```

IF Total_Harga > 70000 THEN
    SET cust_voucher = 'GOSIXTY';
ELSEIF
    (Total_Harga <= 70000 AND Total_Harga >= 50000) THEN
    SET cust_voucher = 'GOFOURTY';
ELSEIF
    Total_Harga < 50000 THEN
    SET cust_voucher = 'GOTWENTY';
END IF;

RETURN (cust_voucher);
END//

```

```

MariaDB [gojek]> DELIMITER //
MariaDB [gojek]> CREATE FUNCTION Customer_Voucher(Total_Harga INTEGER)
-> RETURNS VARCHAR(100)
-> BEGIN
->     DECLARE cust_voucher VARCHAR(100);
->
->     IF Total_Harga > 70000 THEN
->         SET cust_voucher = 'GOSIXTY';
->     ELSEIF
->         (Total_Harga <= 70000 AND Total_Harga >= 50000) THEN
->         SET cust_voucher = 'GOFOURTY';
->     ELSEIF
->         Total_Harga < 50000 THEN
->         SET cust_voucher = 'GOTWENTY';
->     END IF;
->
->     RETURN (cust_voucher);
-> END//
Query OK, 0 rows affected (0.005 sec)

```

Hasil :

```

MariaDB [gojek]> SELECT m.ID_Customer AS "Customer ID", Customer_Voucher(m.Total_Harga) AS "Voucher"
-> FROM membeli m
-> JOIN customer c USING(ID_Customer)
-> ORDER BY 1;

```

Customer ID	Voucher
C0001	GOSIXTY
C0002	GOFOURTY
C0003	GOTWENTY
C0004	GOTWENTY
C0005	GOFOURTY
C0006	GOSIXTY
C0007	GOSIXTY
C0008	GOSIXTY
C0009	GOFOURTY
C0010	GOTWENTY

10 rows in set (0.130 sec)

Stored function diatas digunakan untuk memunculkan kode voucher yang dapat digunakan oleh customer di pemesanan berikutnya. Voucher tersebut didapatkan dari banyaknya uang yang dikeluarkan oleh customer, misalkan customer mengeluarkan uang sebesar lebih dari 70 ribu, maka voucher yang didapatkan adalah diskon 60% (GOSIXTY). Apabila customer mengeluarkan

uang sebesar 50 ribu hingga 70 ribu, maka voucher yang didapatkan adalah diskon 40% (GOFORTY). Apabila customer mengeluarkan uang sebesar kurang dari 50 ribu, maka voucher yang didapatkan adalah diskon 20% (GOTWENTY). Output yang muncul apabila stored function tersebut dijalankan adalah id customer, dan kode voucher yang dapat digunakan untuk pemesanan selanjutnya.

### C. Trigger

```
// 1.
CREATE TABLE log_kendaraan(
    id_log          INTEGER(11) AUTO_INCREMENT,
    id_kendaraan    VARCHAR(5),
    plat            VARCHAR(5),
    plat_baru       VARCHAR(11),
    update_date     DATE,
    PRIMARY KEY(id_log)
);

DELIMITER //
CREATE TRIGGER update_nopol_driver
BEFORE UPDATE ON kendaraan FOR EACH ROW
BEGIN
    INSERT INTO log_kendaraan
    SET id_kendaraan = OLD.ID_Kendaraan,
    plat = OLD.Plat_Kendaraan,
    plat_baru = NEW.Plat_Kendaraan,
    update_date = NOW();
END//
```

```
MariaDB [gojek]> CREATE TABLE log_kendaraan(
-> id_log          INTEGER(11) AUTO_INCREMENT,
-> id_kendaraan    VARCHAR(5),
-> plat            VARCHAR(5),
-> plat_baru       VARCHAR(11),
-> update_date     DATE,
-> PRIMARY KEY(id_log)
-> );
Query OK, 0 rows affected (0.033 sec)

MariaDB [gojek]> DELIMITER //
MariaDB [gojek]> CREATE TRIGGER update_nopol_driver
-> BEFORE UPDATE ON kendaraan FOR EACH ROW
-> BEGIN
-> INSERT INTO log_kendaraan
-> SET id_kendaraan = OLD.ID_Kendaraan,
-> plat = OLD.Plat_Kendaraan,
-> plat_baru = NEW.Plat_Kendaraan,
-> update_date = NOW();
-> END//
Query OK, 0 rows affected (0.008 sec)
```

Hasil :

```
MariaDB [gojek]> DELIMITER ;
MariaDB [gojek]> UPDATE kendaraan SET Plat_Kendaraan = 'B 1978 RF' WHERE ID_Kendaraan = 'K0002';
Query OK, 1 row affected (0.023 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [gojek]> SELECT * FROM log_kendaraan;
+-----+-----+-----+-----+-----+
| id_log | id_kendaraan | plat | plat_baru | update_date |
+-----+-----+-----+-----+-----+
| 1 | K0002 | B 662 | B 1978 RF | 2022-12-16 |
+-----+-----+-----+-----+-----+
1 row in set (0.000 sec)

MariaDB [gojek]> SELECT * FROM kendaraan;
+-----+-----+-----+
| ID_Kendaraan | Tipe_Kendaraan | Plat_Kendaraan |
+-----+-----+-----+
| K0001 | Sportbike | B 1971 PK |
| K0002 | Sedan | B 1978 RF |
| K0003 | Sportbike | B 9087 SXS |
| K0004 | Hatchback | B 9075 WBX |
| K0005 | Sedan | B 5015 XYM |
| K0006 | SUV | B 1971 SVU |
| K0007 | Skuter | B 5171 BXM |
| K0008 | Bebek | B 7177 VLP |
| K0009 | Skuter | B 6176 VXB |
| K0010 | Sportbike | B 1617 RDX |
+-----+-----+-----+
10 rows in set (0.000 sec)
```

Trigger tersebut berfungsi sebagai pengubah plat kendaraan driver. Misalkan, driver mengubah plat menjadi 'B 1978 RF' pada id kendaraan 'K0002', maka kita memasukkannya ke dalam table dan kemudian apabila kita melakukan cek ke log kendaraan, maka akan muncul history dari perubahan tersebut. Isi dari tabel log kendaraan adalah id log, id kendaraan driver, plat driver yang sebelumnya, plat driver yang baru dan tanggal dilakukannya update tersebut.

```
// 2.
CREATE TABLE outlet_history(
    id_log          INTEGER(11) AUTO_INCREMENT,
    id_outlet       VARCHAR(5),
    nama_outlet     VARCHAR(30),
    jumlah         INTEGER(11),
    jumlah_update   INTEGER(11),
    update_date     DATE,
    PRIMARY KEY(id_log)
);

DELIMITER //
CREATE TRIGGER update_jumlah_outlet
AFTER UPDATE ON outlet FOR EACH ROW
BEGIN
    INSERT INTO outlet_history
    SET id_outlet = OLD.ID_Outlet,
        nama_outlet = OLD>Nama_Outlet,
        jumlah = OLD.Jumlah,
        jumlah_update = NEW.Jumlah,
        update_date = NOW();
END//
```

```

MariaDB [gojek]> CREATE TABLE outlet_history(
->   id_log          INTEGER(11) AUTO_INCREMENT,
->   id_outlet       VARCHAR(5),
->   nama_outlet     VARCHAR(30),
->   jumlah          INTEGER(11),
->   jumlah_update   INTEGER(11),
->   update_date     DATE,
->   PRIMARY KEY(id_log)
-> );
Query OK, 0 rows affected (0.023 sec)

MariaDB [gojek]> DELIMITER //
MariaDB [gojek]> CREATE TRIGGER update_jumlah_outlet
-> AFTER UPDATE ON outlet FOR EACH ROW
-> BEGIN
->   INSERT INTO outlet_history
->   SET id_outlet = OLD.ID_Outlet,
->   nama_outlet = OLD>Nama_Outlet,
->   jumlah = OLD.Jumlah,
->   jumlah_update = NEW.Jumlah,
->   update_date = NOW();
-> END//
Query OK, 0 rows affected (0.007 sec)

```

Hasil :

```

MariaDB [gojek]> DELIMITER ;
MariaDB [gojek]> UPDATE outlet SET Jumlah = 5 WHERE ID_Outlet = '00006';
Query OK, 1 row affected (0.014 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [gojek]> SELECT * FROM outlet_history;
+-----+-----+-----+-----+-----+-----+
| id_log | id_outlet | nama_outlet | jumlah | jumlah_update | update_date |
+-----+-----+-----+-----+-----+-----+
| 1 | 00006 | Cheese Tokyo | 1 | 5 | 2022-12-16 |
+-----+-----+-----+-----+-----+-----+
1 row in set (0.000 sec)

MariaDB [gojek]> SELECT * FROM outlet;
+-----+-----+-----+-----+-----+
| ID_Outlet | Nama_Outlet | Nama_Makanan | Jumlah | Kode_Proposal |
+-----+-----+-----+-----+-----+
| 00001 | Restoran Manado Thamrin | Bubur Manado | 3 | P0010 |
| 00002 | Toko Kue Nagamas | Kue Kering | 1 | P0004 |
| 00003 | Sushi Tokyo | Sushi | 5 | P0001 |
| 00004 | Sate Pak Umar | Sate Kambing | 2 | P0002 |
| 00005 | Restoran Padang | Nasi Rendang Mercon | 9 | P0009 |
| 00006 | Cheese Tokyo | Cheese Thai Tea | 5 | P0003 |
| 00007 | Nasi Goreng Pak Ahmad | Nasi Goreng Gila | 1 | P0007 |
| 00008 | Ayam Kremes Dian | Ayam Goreng Kremes | 3 | P0006 |
| 00009 | Bento Makusu | Nasi Bento | 2 | P0005 |
| 00010 | Chicken Wrecked By Tina | Ayam Geprek | 1 | P0008 |
+-----+-----+-----+-----+-----+
10 rows in set (0.000 sec)

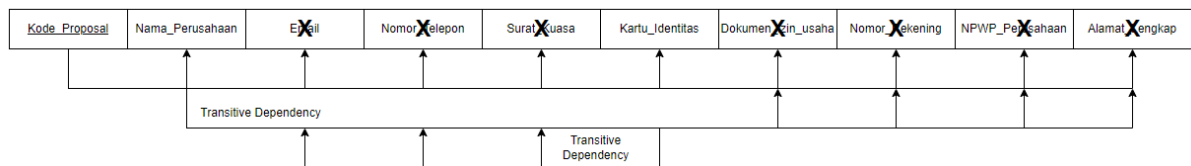
```

Trigger tersebut berfungsi sebagai pengubah jumlah store dari sebuah outlet. Misalkan, pemilik outlet mengubah jumlah outlet menjadi 5 pada id outlet '00006', maka kita memasukkannya ke dalam table dan kemudian apabila kita melakukan cek ke outlet history, maka akan muncul history dari perubahan tersebut. Isi dari tabel outlet history adalah id log, id outlet, nama outlet, jumlah store sebelumnya, jumlah store sesudahnya, dan tanggal dilakukannya update tersebut.

## 5. Functional Dependencies & Normalization

<https://drive.google.com/drive/u/0/folders/1qth8M-12rh-1vOIkfz54VLBIK2ZX8DQR>

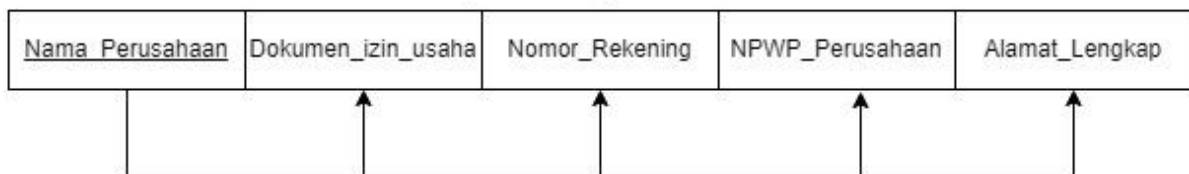
- Partnership



### 2 NF Remove Functional Dependency

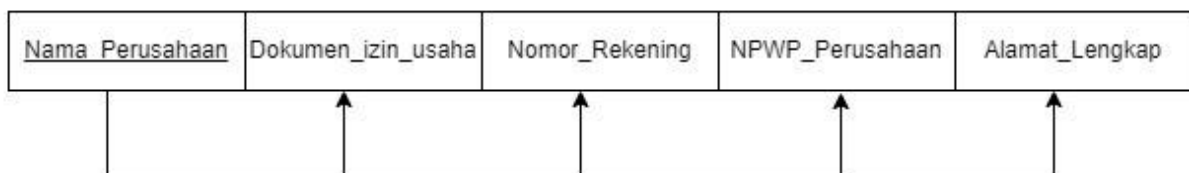
Tabel tidak memiliki functional dependency

### 3 NF Remove Transitive Dependency

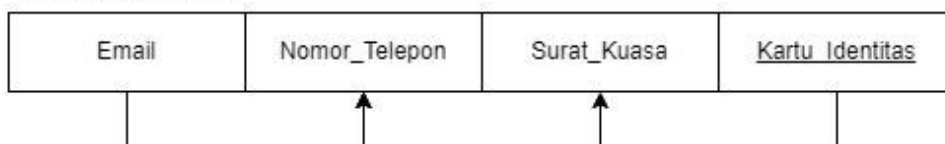


### Result

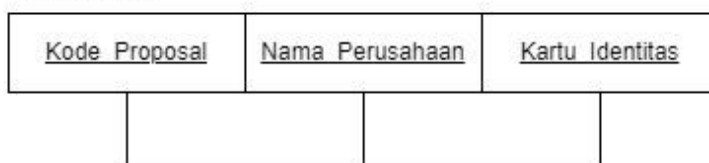
Perusahaan



Penanggung Jawab

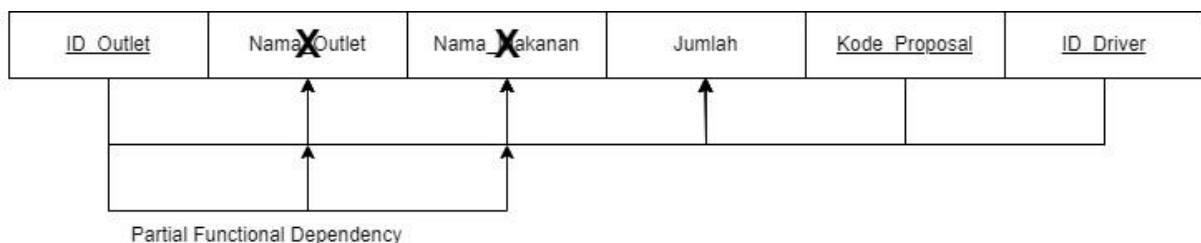


Partnership

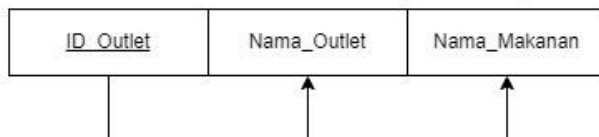


Yang pertama dilakukan adalah mencari tabel non-key yang bisa diketahui dengan primary key kode\_proposal. Dalam kasus ini saya mendefinisikan tidak ada, karena nama perusahaan bisa diperkecil lagi menjadi satu tabel tersendiri yang berisi data perusahaan yang diputuskan menjadi transitive dependency karena alamat perusahaan, dokumen izin usaha, nomor rekening tentunya hanya bisa diketahui jika sudah diketahui nama suatu perusahaan tersebut. Begitu pula dengan identitas penanggung jawab perusahaan seperti surat kuasa, nomor telepon dan email baru akan diketahui jika ada data dari kartu identitasnya sehingga seluruh tabel yang bukan primary key akan dicoret di tabel utama dan seluruh primary key akan menjadi satu entity partnership yang menghubungkan antar entity lainnya

- Outlet



## 2 NF Remove Functional Dependency

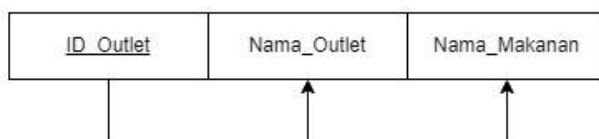


## 3 NF Remove Transitive Dependency

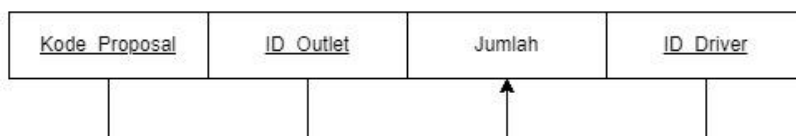
Tabel tidak memiliki transitive dependency

## Result

Outlet



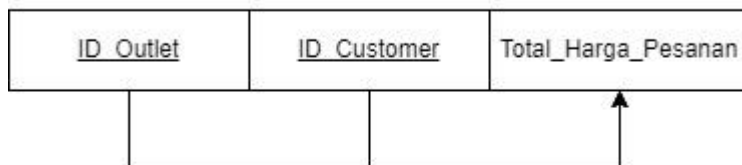
Pesanan



Pada tabel outlet nama outlet dan nama makanan yang dijual dari outlet tersebut bisa diketahui melalui ID\_Outlet. Sehingga tersisa primary key kode\_proposal, ID\_outlet, dan ID\_Driver yang menjadi komponen wajib untuk terjadinya keterangan pesanan dan jumlah dari pesanan driver dari customer.

- Membeli

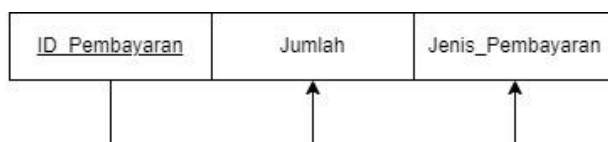
Membeli



Sudah 3 NF

Tabel membeli sudah menjadi tabel 3NF karena merupakan penghubung yang menjadi pendukung untuk terjadinya pembelian Customer kepada Outlet dilengkapi dengan jumlah yang harus dibayar. Tabel ini akan bekerja sama melengkapi tabel pesanan.

- Pembayaran



2 NF Remove Functional Dependency

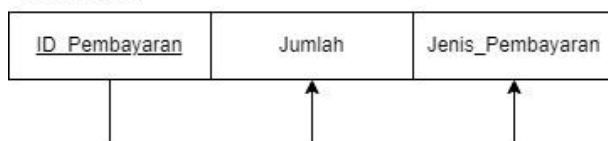
Tabel tidak memiliki functional dependency

3 NF Remove Transitive Dependency

Tabel tidak memiliki transitive dependency

Result

Pembayaran



Tabel ini sudah menjadi 3NF karena seluruh kolom non key bisa diketahui dengan primary key nya yaitu ID\_Pembayaran

- Customer

<u>ID_Customer</u>	Nama <del>X</del> Depan	Nama <del>X</del> Belakang	Nomor <del>X</del> Telepon	<del>X</del> Email	Alamat <del>X</del> Utama	Tujuan <del>X</del>	<del>X</del>	ID_Pembayaran	ID_Admin	Titik Koordinat	ID_Driver
	↑	↑	↑	↑	↑	↑	↑				
	↑	↑	↑	↑	↑	↑	↑				

## 2 NF Remove Functional Dependency

<u>ID_Customer</u>	Nama_Depan	Nama_Belakang	Nomor_Telepon	Email	Alamat_Utama	Tujuan	NIK
	↑	↑	↑	↑	↑	↑	↑

## 3 NF Remove Transitive Dependency

Tabel tidak memiliki transitive dependency

## Result

Customer

<u>ID_Customer</u>	Nama_Depan	Nama_Belakang	Nomor_Telepon	Email	Alamat_Utama	Tujuan	NIK
	↑	↑	↑	↑	↑	↑	↑

Layani\_Customer

<u>ID_Customer</u>	<u>ID_Pembayaran</u>	<u>ID_Admin</u>	<u>Titik_Koordinat</u>	<u>ID_Driver</u>

Seluruh kolom non-key pada tabel ini bisa dilihat atau diwakilkan dengan tabel primary key nya yaitu ID\_Customer, sehingga pada tabel aslinya kolom tersebut dicoret dan seluruh sisa kolom primary key pada tabel asli membuat tabel baru dan nantinya akan menjadi penghubung antar tabel.



- Admin

<u>ID_Admin</u>	Nama_Depan	Nama_Belakang	Username	Password	Email
	↑	↑	↑	↑	↑

## 2 NF Remove Functional Dependency

Tabel tidak memiliki functional dependency

## 3 NF Remove Transitive Dependency

Tabel tidak memiliki transitive dependency

## Result

Admin

<u>ID_Admin</u>	Nama_Depan	Nama_Belakang	Username	Password	Email
	↑	↑	↑	↑	↑

Tabel Admin sudah 3NF karena seluruh kolomnya yang bukan primary key bisa diwakilkan dengan primary key nya yaitu ID\_Admin

- Navigasi

<u>Titik Koordinat</u>
------------------------

## 2 NF Remove Functional Dependency

Tabel tidak memiliki functional dependency

## 3 NF Remove Transitive Dependency

Tabel tidak memiliki transitive dependency

## Result

Navigasi

<u>Titik Koordinat</u>
------------------------

Titik koordinat merupakan kolom tunggal yang ada pada tabel navigasi sehingga dia tentunya bisa mewakili dirinya sendiri dan sudah berbentuk 3NF.

- Nomor\_Telepon\_Driver

Nomor\_Telepon\_driver

<u>ID Driver</u>	<u>Nomor Telepon</u>
------------------	----------------------

## Sudah 3 NF

Tabel ini sudah berbentuk 3NF karena hanya merupakan penghubung antara tabel Driver dan nomor telepon dari drivernya.

- Driver

<u>ID Driver</u>	<del>NIK</del>	<del>Nama Depan</del>	<del>Nama Belakang</del>	<del>Jenis Kelamin</del>	<del>Email</del>	<del>Jenis Kendaraan</del>	<u>Titik Koordinat</u>	<u>ID Kendaraan</u>
	↑	↑	↑	↑	↑	↑		

2 NF Remove Functional Dependency

<u>ID Driver</u>	NIK	Nama Depan	Nama Belakang	Jenis Kelamin	Email	Jenis Kendaraan
	↑	↑	↑	↑	↑	↑

3 NF Remove Transitive Dependency

Tabel tidak memiliki Transitive dependency

Result

Driver

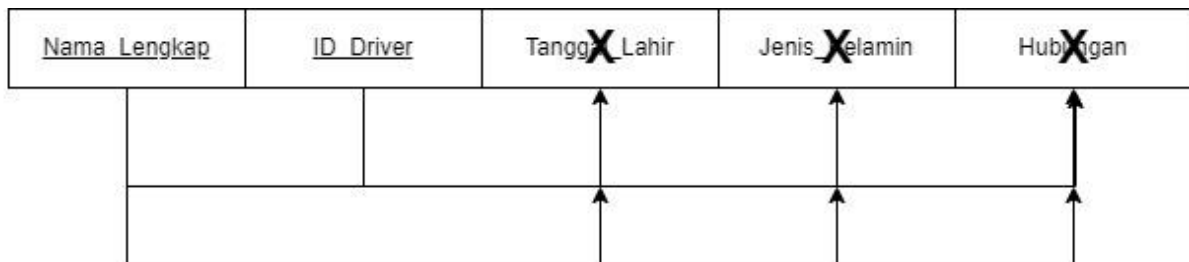
<u>ID Driver</u>	NIK	Nama Depan	Nama Belakang	Jenis Kelamin	Email	Jenis Kendaraan
	↑	↑	↑	↑	↑	↑

Jemputan Customer

<u>ID Driver</u>	<u>ID Kendaraan</u>	<u>Titik Koordinat</u>

Seluruh kolom non key pada tabel ini bisa direpresentasikan dengan ID\_Driver sehingga perlu dihilangkan functional dependency nya dan tabel non key tersebut dihapus dari tabel aslinya sehingga siswa seluruh tabel primary key pada tabel tersebut menjadi tabel baru dan nantinya akan menjadi penghubung.

- Tanggungan\_Driver



## 2 NF Remove Functional Dependency



## 3 NF Remove Transitive Dependency

Tabel tidak memiliki Transitive dependency

### Result

Data\_Tanggungan\_Driver

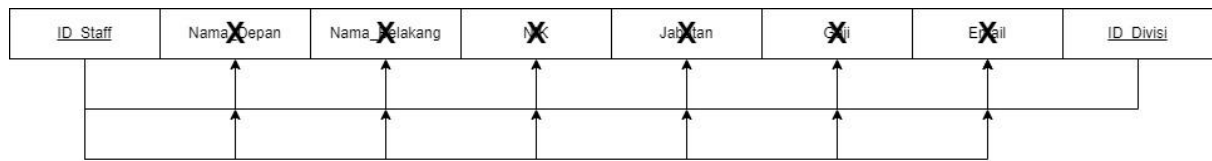


Tanggungan\_Driver

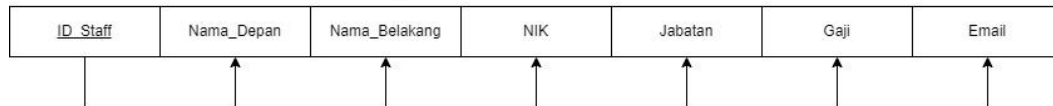


Seluruh kolom non-key pada tabel ini bisa direpresentasikan dengan kolom Nama\_Lengkap sehingga akan tersisa 2 tabel primary pada tabel utama sehingga menjadikan tabel data\_tanggungan\_driver akan terhubung pada Driver yang memiliki tanggungan tersebut.

- Staff



## 2 NF Remove Functional Dependency

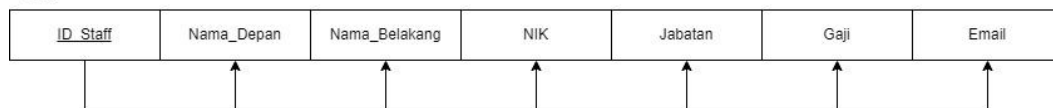


## 3 NF Remove Transitive Dependency

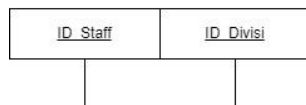
Tabel tidak memiliki Transitive dependency

## Result

Staff

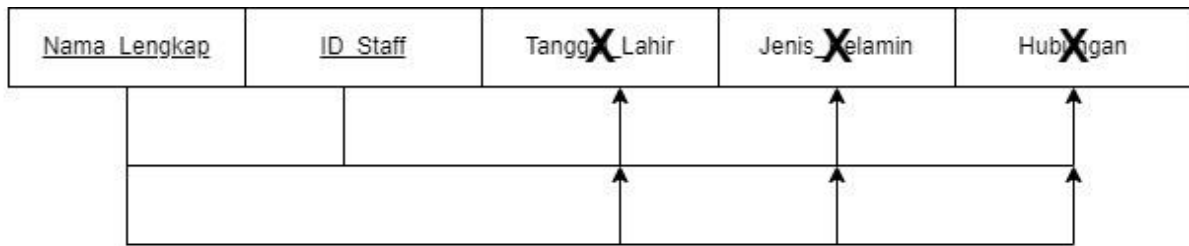


Departmet\_Staff



Seluruh kolom non-key pada tabel staff dapat diwakilkan dengan kolom\_ID\_Staff, sehingga pada tabel aslinya tersisa 2 primary key, dengan fungsi ID\_Divisi nantinya akan menghubungkan Staff dengan Divisinya.

- Tanggungan\_Staff



## 2 NF Remove Functional Dependency



## 3 NF Remove Transitive Dependency

Tabel tidak memiliki Transitive dependency

## Result

Data\_Tanggungan\_Staff



Tanggungan\_Staff



Seluruh kolom non-key pada tabel ini bisa direpresentasikan dengan kolom Nama\_Lengkap sehingga akan tersisa 2 tabel primary pada tabel utama sehingga menjadikan tabel data\_tanggungan\_staff akan terhubung pada Staff yang memiliki tanggungan tersebut.

- Divisi



## 2 NF Remove Functional Dependency

Tabel tidak memiliki functional dependency

## 3 NF Remove Transitive Dependency

Tabel tidak memiliki Transitive dependency

## Result



Tabel ini sudah berbentuk 3NF karena nama divisi bisa dilihat dari ID\_Divisi itu sendiri.

- Bekerja

Bekerja



## Sudah 3 NF

Sudah 3 NF karena ini merupakan tabel penghubung antara staff dan layanan

- Nomor\_Telepon\_Staff

Nomor\_Telepon\_Staff

<u>ID_Staff</u>	<u>Nomor_Telepon</u>
-----------------	----------------------

Sudah 3 NF

Sudah 3 NF karena ini merupakan tabel penghubung antara staff dan nomor teleponnya

- Layanan

<u>ID_Layanan</u>	<del>Harga</del>	<del>Jenis_Layanan</del>	<u>ID_Driver</u>	<u>ID_Customer</u>	Estimasi_waktu_tiba	Waktu_Tiba	Tanggal_Transaksi

2 NF Functional Dependency

<u>ID_Layanan</u>	Harga	Jenis_Layanan

3 NF Remove Transitive Dependency

Tabel tidak memiliki Transitive dependency

Result

Layanan

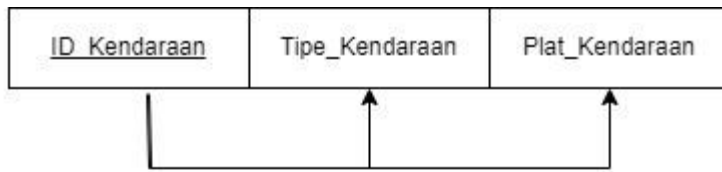
<u>ID_Layanan</u>	Harga	Jenis_Layanan

Pelayanan\_Driver\_untuk\_customer

<u>ID_Layanan</u>	<u>ID_Driver</u>	<u>ID_Customer</u>	Estimasi_waktu_tiba	Waktu_Tiba	Tanggal_Transaksi

Yang bisa diwakilkan dengan kolom ID\_Layanan hanya harga dan jenis layanannya, selain itu tidak ada lagi primary key yang dapat mewakili kolom lainnya sehingga kolom yang tersisa pada tabel asli akan membentuk tabel baru menjadi pelayanan\_driver untuk customer.

- Kendaraan



## 2 NF Remove Functional Dependency

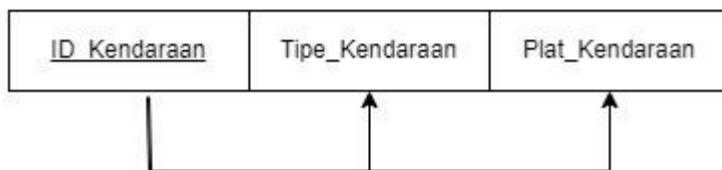
Tabel tidak memiliki functional dependency

## 3 NF Remove Transitive Dependency

Tabel tidak memiliki Transitive dependency

## Result

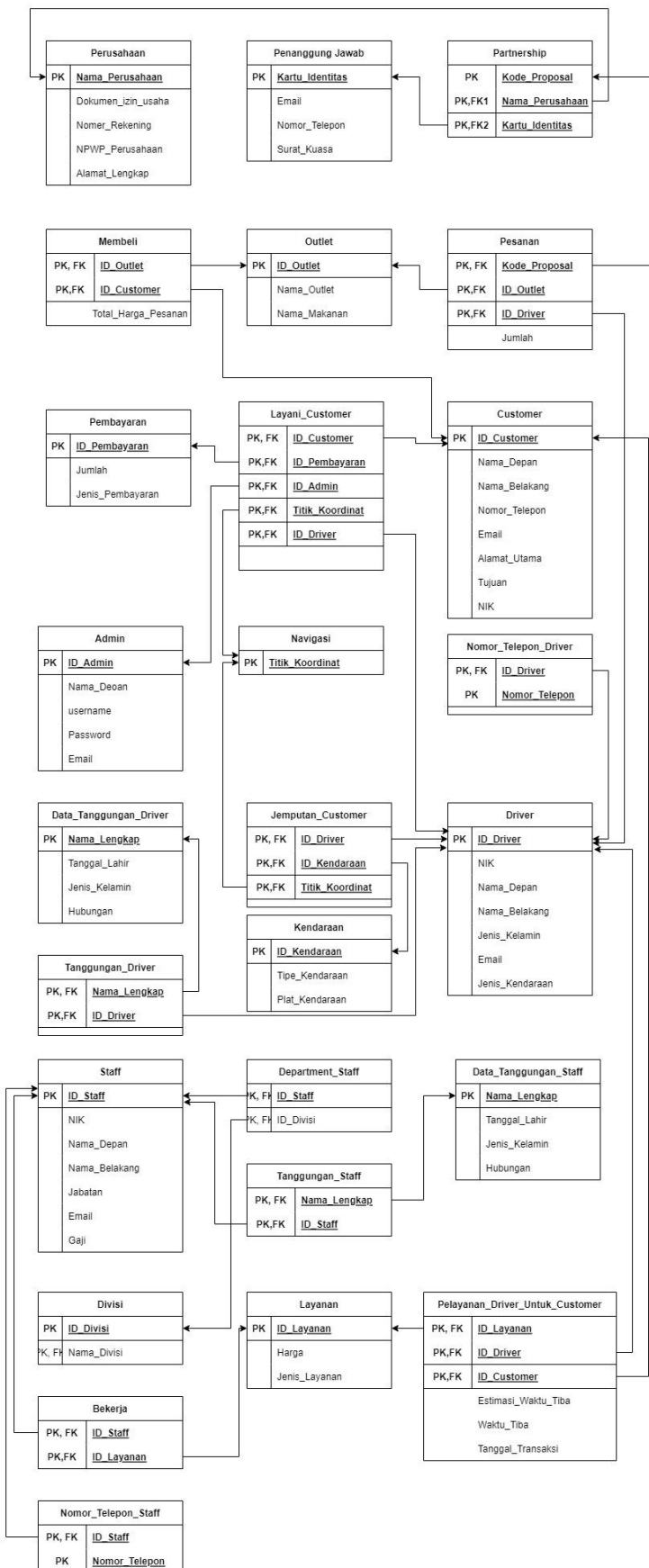
Kendaraan



Sudah berbentuk 3NF karena seluruh kolom non key pada tabel ini bisa direpresentasikan menggunakan satu-satunya primary key pada tabel ini.



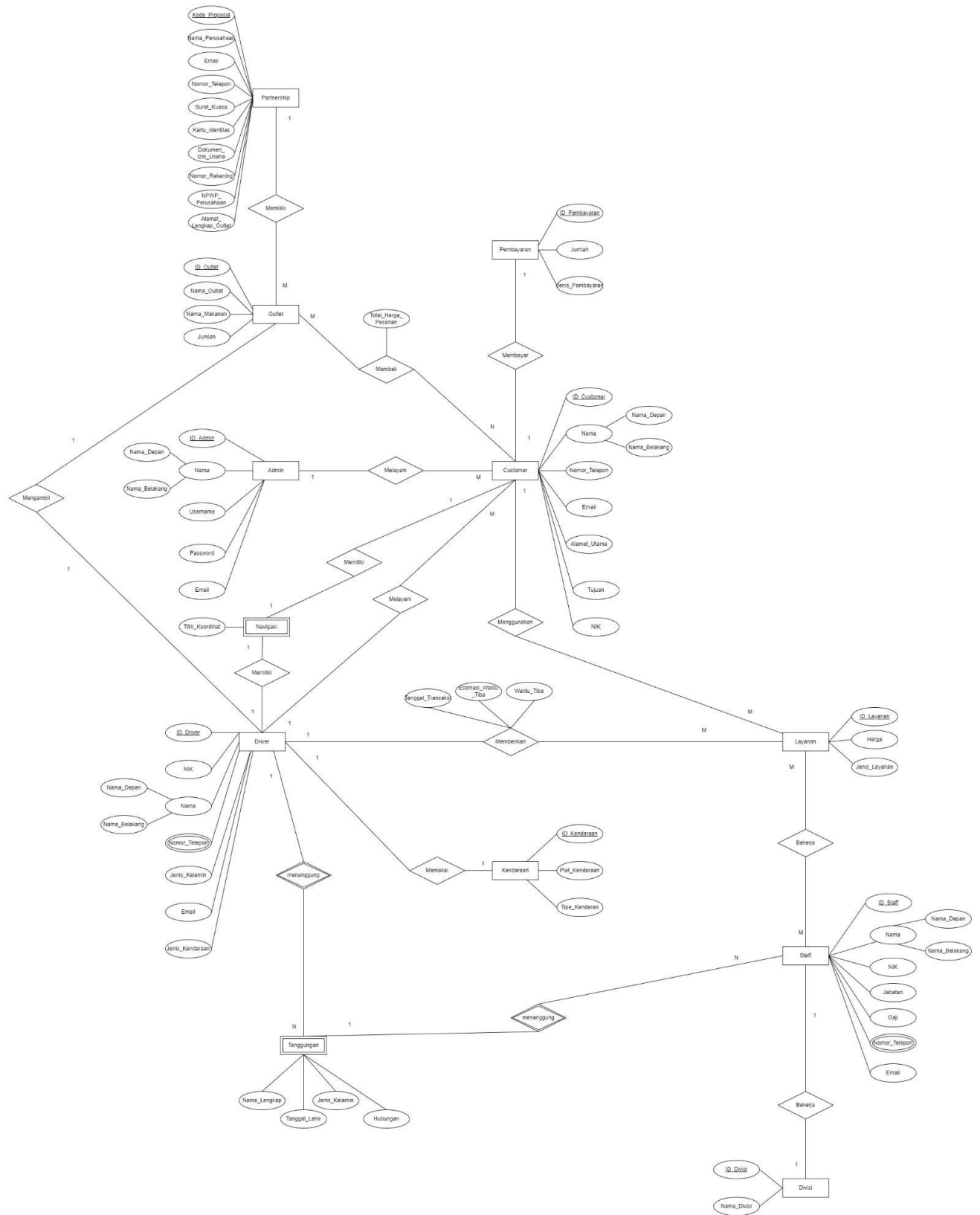
## ● Schema Result



Perbedaan yang terjadi dari database schema baru ini tidak terlalu signifikan. Hal ini bisa terjadi karena dengan menggunakan metode normalisasi tabel terdahulu menjadi lebih kecil (terpecah menjadi beberapa bagian) yang tiap-tiap bagiannya hanya tersambung dengan tabel aslinya sehingga tidak terjadi perubahan proses dalam sistem. Tabel yang tadinya berjumlah 17 menjadi 26.

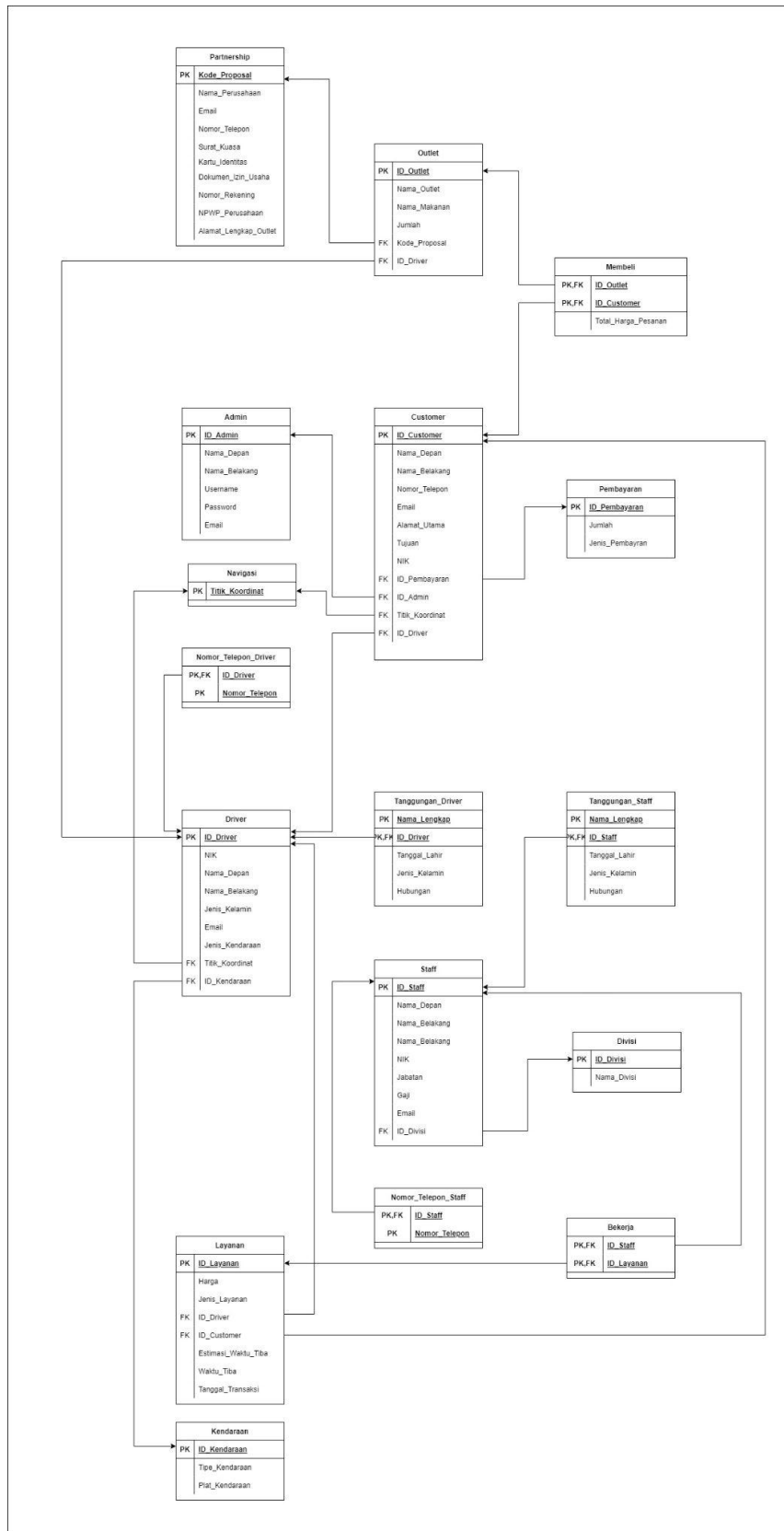
Jadi, perubahan yang terjadi hanya pada tabel menjadi bertambah banyak sedangkan proses, sistem dan relasi tetap sama.

<https://drive.google.com/drive/u/0/folders/1VLS76MexPUEZCFhXZkOSlc27Owv9xu88>



## Schema Diagram

<https://drive.google.com/drive/u/0/folders/1VLS76MexPUeZCFhXZkOSlc27Owv9xu88>



## 6. Evaluasi Diri

### - Arfigo Ezra Pratama :

Pada pengerjaan uas ini, kami dengan cepat membagi-bagi tugas antar anggota agar pengerjaan dapat dicicil dari jauh-jauh hari bersama-sama. Menurut saya kinerja antar anggota kelompok dalam mengerjakan UAS ini sangat baik. Semua mengerjakan bagiannya masing-masing dengan baik dan tepat waktu. Namun satu kekurangan yang saya rasakan adalah kurangnya komunikasi antar kelompok. Karena bisa dibilang kami tidak ada melakukan zoom atau team meeting dan hanya berkomunikasi saat pembagian tugas dan H-1 pengumpulan untuk mengecek dan mengevaluasi hasil akhir laporan kami. Namun terlepas masalah tersebut, saya merasa bahwa kerjasama antar anggota kami cukup baik, karena ketika saya bertanya mengenai suatu hal yang saya tidak mengerti, saya dibantu oleh anggota kelompok untuk memahami dan menyelesaikan masalah tersebut dengan baik.

### - Wilcoustine Qhristmas Pniel Wijaya :

Pada pengerjaan UAS kali ini, kami memulainya dengan melihat soal mana yang kemungkinan berkesinambungan dan dapat digabung dan dikerjakan oleh 1 orang saja. Lalu kami membuat pembagian tugas siapa-siapa saja yang mengerjakan nomor berapa. Dalam pengerjaan UAS ini kami memang terkesan individual dan fokus pada pengerjaan tugas masing-masing dan bisa terbilang cukup jarang melakukan komunikasi satu-sama lain. Tetapi kami tetap, melakukan update dan mengabari satu-sama lain tentang progress kami. Selain itu bisa dibilang pengerjaan UAS ini cukup lama, dikarenakan ada kendala dalam pengerjaan tugas masing-masing. Tetapi bisa dibilang kinerja kita bersama sudah cukup baik, kami bisa mengerjakan tugas masing-masing sebelum deadline pengumpulan, dan juga kerja sama kami bisa terbilang cukup baik, dikarenakan adanya kepedulian satu sama lain, saling mengingatkan, dan membantu.

### - Nisa'ul Fadhilah :

Pembagian tugas dilakukan dengan adil karena masing-masing memilih. Awalnya bagian normalisasi dinilai paling ribet jadi saya sendiri memilih normalisasi. Tidak ada kendala dalam pengerjaan, proses meeting tidak pernah

terjadi tetapi menurut saya hal itu juga tidak mempengaruhi kinerja asalkan komunikasi tetap terjadi dan selalu ingat tanggungjawab. Hasil cukup baik setiap kesulitan selalu menemukan jalan keluar tetapi ada sedikit perubahan yang awalnya Transaction Processing dan User Management, Privilege Management, & Views dikerjakan oleh arfigo diambil alih oleh ayu febriana. Hal ini menurut saya juga menjadi bukti bahwa tidak ada saling mengandalkan di kelompok kami, semua sadar akan kewajibannya. Yang mampu “membantu” yang tidak mampu tidak malu untuk “dibantu”.

- Ayu Febriana Lingga :

Dalam pembagian tugas ini, kami mengawali dengan memilih nomor secara individu. Setelah itu, kami mulai mengerjakan tugas tersebut masing-masing. Untuk kinerja dari masing-masing perorangan cukup baik, namun terdapat kurangnya komunikasi antar anggota baik secara lisan maupun melalui chat line. Walaupun kurangnya komunikasi, kami tetap saling membantu apabila terdapat anggota yang tidak paham ataupun terkendala dengan device. Walaupun begitu, kami tetap ingat dengan tugas utama kami seperti pada pembagian tugas diatas. Untuk hasil kerja untuk UAS ini, sudah cukup baik namun jauh lebih baik apabila pengerjaan UAS mata kuliah ini dipercepat dan tidak mendekati deadline.