

# **LAPORAN FINAL PROJECT**

## **“Transportasi Umum”**

### **Kelompok 6**

#### **I. DATABASE REQUIREMENTS**

Database ini ialah miniworld untuk one stop service information untuk layanan transportasi, tetapi dibatasi untuk beberapa layanan transportasi umum saja. Layanan transportasi umum sendiri dibagi menjadi 2 jenis yakni transportasi antar kota dan transportasi lokal.

Transportasi antar kota yakni layanan transportasi umum yang bisa digunakan untuk perjalanan lintas kota atau provinsi. Diantaranya yakni penerbangan, pelayaran, kereta api, dan bus. Adapun untuk transportasi lokal ialah layanan yang wilayah operasinya masih dalam satu kota. Contohnya yakni bus lokal dan taxi.

Semua layanan transportasi disediakan oleh sebuah perusahaan transportasi. Setiap perusahaan transportasi bisa memiliki cabang kantor di setiap kota. Khusus untuk perusahaan penerbangan, juga bisa memiliki cabang di bandara.

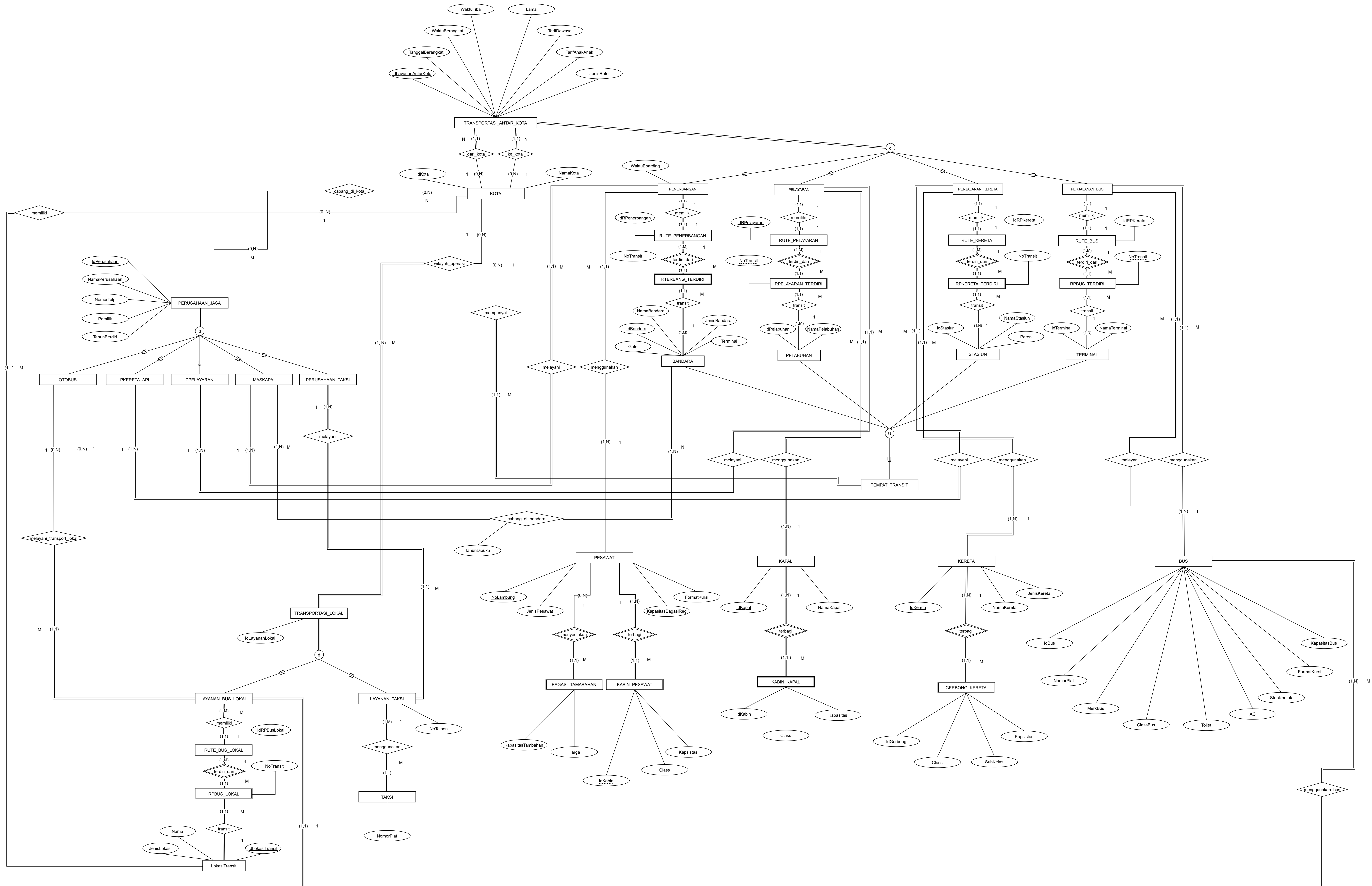
Setiap perusahaan transportasi memiliki layanan transportasi dengan detail harga, asal perjalanan, tujuan perjalanan, rutenya, dan dimana saja akan transit. Layanan transportasi terbagi berdasarkan alat transportasinya dan perusahaan mana yang menjalankannya.

Secara umum informasi yang bisa didapatkan pengguna ialah:

1. Informasi perusahaan penyedia layanan transportasi
2. Layanan perjalanan transportasi dari dan ke kota mana saja
3. Detail dari setiap perjalanan seperti harga, waktu berangkat, waktu tiba dan alat transportasi yang mana yang dipakai.
4. Detail setiap alat transportasi seperti kelas, fasilitas dan maksimum penumpang
5. Setiap layanan transportasi memiliki rute dan transit dimana saja
6. Tempat-tempat transit setiap jenis layanan
7. Cabang setiap perusahaan

#### **II. MODEL DATA ERD**

Data model ER/EER dari database requirement



### III. MAPPING DAN RELATIONAL MODEL

#### A. Step

##### Strong Entity

- KOTA(IdKota, NamaKota)
- PERUSAHAAN\_JASA(IdPerusahaan, NamaPerusahaan, NomorTelp, Pemilik, TahunBerdiri, JenisPerusahaan)
- PESAWAT(NoLambung, JenisPesawat, KapasitasBagasiReg)
- KAPAL(NoLambung, NamaKapal)
- KERETA(IdKereta, NamaKereta)
- BUS(NomorPlat, MerkBus, Kelas, Toilet, AC, StopKontak, FormatKursi, KapasitasBus)
- TAKSI(NomorPlat)
- LOKASI\_TRANSIT\_BUS(IdLokasi, NamaLokasi, JenisLokasi)
- BANDARA(IdBandara, NamaBandara, JenisBandara, Gate, Terminal)
- PELABUHAN(IdPelabuhan, NamaPelabuhan)
- STASIUN(IdStasiun, NamaStasiun, Peron)
- TERMINAL(IdTerminal, NamaTerminal)
- Kota(KOTA(IdKota)))
- RUTE\_PENERBANGAN(IdRutePenerbangan)
- RUTE\_PELAYARAN(IdRutePelayaran)
- RUTE\_KERETA(IdRuteKereta)
- RUTE\_BUS(IdRuteBus)
- RUTE\_BUS\_LOKAL(IdRuteBusLokal)
- LAYANAN\_BUS\_LOKAL(IdLayananBusLokal)
- LAYANAN\_TAKSI(IdLayananTaksi, NomorTelepon)
- (Specializaton)
- PENERBANGAN(IdPenerbangan, TglBerangkat, WaktuBoarding, TglTiba, WaktuTiba, TarifDewasa, TarifAnak, JenisRute)
- PELAYARAN(IdPelayaran, TglBerangkat, WaktuBerangkat, TglTiba, WaktuTiba, TarifDewasa, TarifAnak, JenisRute)
- PERJALANAN\_KERETA(IdPerjalananKereta, TglBerangkat, WaktuBerangkat, TglTiba, WaktuTiba, TarifDewasa, TarifAnak, JenisRute)
- PERJALANAN\_BUS(IdPerjalananBus, TglBerangkat, WaktuBerangkat, TglTiba, WaktuTiba, TarifDewasa, TarifAnak, JenisRute)

##### Weak Entity

- BAGASI\_TAMBAHAN(NomorLambung(PESAWAT(NoLambung)), BagasiTambahan, Harga)
- KABIN\_PESAWAT(NoLambung(PESAWAT(NoLambung)), IdKabin, Kelas, FormatKursi, Kapasitas)
- KABIN\_KAPAL(NoLambung(KAPAL(NoLambung)), IdKabin, Kelas, Kapasitas)
- GERBONG\_KERETA(IdKereta(KERETA(IdKereta)), IdGerbong, Kelas, SubKelas, Kapasitas)
- RUTE\_PENERBANGAN\_TERDIRI(IdRutePenerbangan(RUTE\_PENERBANGAN(IdRutePenerbangan)), NoTransit)
- RUTE\_PELAYARAN\_TERDIRI(IdRutePelayaran(RUTE\_PELAYARAN(IdRutePelayaran)), NoTransit)
- RUTE\_KERETA\_TERDIRI(IdRuteKereta(RUTE\_KERETA(IdRuteKereta)), NoTransit)
- RUTE\_BUS\_TERDIRI(IdRuteBus(RUTE\_BUS(IdRuteBus)), NoTransit)
- RUTE\_BUS\_LOKAL\_TERDIRI(IdRuteBusLokal(RUTE\_BUS\_LOKAL(IdRuteBusLokal)), NoTransit)

### 1:1 Relationship

- RUTE\_PENERBANGAN(IdRutePenerbangan, IdPenerbangan(PENERBANGAN(IdPenerbangan)))
- RUTE\_PELAYARAN(IdRutePelayaran, IdPelayaran(PELAYARAN(IdPelayaran)))
- RUTE\_KERETA(IdRuteKereta, IdPerjalananKereta(PERJALANAN\_KERETA(IdPerjalananKereta)))
- RUTE\_BUS(IdRuteBus, IdPerjalananBus(PERJALANAN\_BUS(IdPerjalananBus)))
- RUTE\_BUS\_LOKAL(IdRuteBusLokal, IdLayananBusLokal(LAYANAN\_BUS\_LOKAL(IdLayananBusLokal)))
- LOKASI\_TRANSIT\_BUS(IdLokasi, NamaLokasi, JenisLokasi, Kota(KOTA(IdKota)))

### 1:1 Union Relationship

- BANDARA(IdBandara, NamaBandara, JenisBandara, Gate, Terminal, Kota(KOTA(IdKota)))
- PELABUHAN(IdPelabuhan, NamaPelabuhan, Kota(KOTA(IdKota)))
- STASIUN(IdStasiun, NamaStasiun, Peron, Kota(KOTA(IdKota)))
- TERMINAL(IdTerminal, NamaTerminal, Kota(KOTA(IdKota)))

### 1:M Relationship

- TAKSI(NomorPlat, IdLayananTaksi(LAYANAN\_TAKSI(IdLayananTaksi)))
- RUTE\_PENERBANGAN\_TERDIRI(IdRutePenerbangan(RUTE\_PENERBANGAN(IdRutePenerbangan)), NoTransit, IdBandara(BANDARA(IdBandara)))
- RUTE\_PELAYARAN\_TERDIRI(IdRutePelayaran(RUTE\_PELAYARAN(IdRutePelayaran)), NoTransit, IdPelabuhan(PELABUHAN(IdPelabuhan)))
- RUTE\_KERETA\_TERDIRI(IdRuteKereta(RUTE\_KERETA(IdRuteKereta)), NoTransit, IdStasiun(STASIUN(IdStasiun)))
- RUTE\_BUS\_TERDIRI(IdRuteBus(RUTE\_BUS(IdRuteBus)), NoTransit, IdTerminal(TERMINAL(IdTerminal)))
- RUTE\_BUS\_LOKAL\_TERDIRI(IdRuteBusLokal(RUTE\_BUS\_LOKAL(IdRuteBusLokal)), NoTransit, IdLokasiTransit(LOKASI\_TRANSIT\_BUS(IdLokasi)))

### 1:M dengan Specialization

- PENERBANGAN(IdPenerbangan, TglBerangkat, WaktuBoarding, TglTiba, WaktuTiba, TarifDewasa, TarifAnak, JenisRute, DariKota(KOTA(IdKota)), KeKota(KOTA(IdKota)), Maskapai(PERUSAHAAN\_JASA(IdPerusahaan)), Pesawat(PESAWAT(NoLambung)))
- PELAYARAN(IdPelayaran, TglBerangkat, WaktuBerangkat, TglTiba, WaktuTiba, TarifDewasa, TarifAnak, JenisRute, DariKota(KOTA(IdKota)), KeKota(KOTA(IdKota)), Penyedia(PERUSAHAAN\_JASA(IdPerusahaan)), Kapal(KAPAL(NoLambung)))
- PERJALANAN\_KERETA(IdPerjalananKereta, TglBerangkat, WaktuBerangkat, TglTiba, WaktuTiba, TarifDewasa, TarifAnak, JenisRute, DariKota(KOTA(IdKota)), KeKota(KOTA(IdKota)), Penyedia(PERUSAHAAN\_JASA(IdPerusahaan)), Kereta(KERETA(IdKereta)))
- PERJALANAN\_BUS(IdPerjalananBus, TglBerangkat, WaktuBerangkat, TglTiba, WaktuTiba, TarifDewasa, TarifAnak, JenisRute, DariKota(KOTA(IdKota)), KeKota(KOTA(IdKota)), Penyedia(PERUSAHAAN\_JASA(IdPerusahaan)), Kereta(BUS(NomorPlat)))

### M:N Relationship

- CABANG\_KOTA(IdCabangKota, NomorTelepon, TahunDibuka, IdKota(KOTA(IdKota)), IdPerusahaan(PERUSAHAAN\_JASA(IdPerusahaan)))

- CABANG\_BANDARA(IdCabangBandara, TahunDibuka, IdBandara(BANDARA(Id-Bandara)), IdMaskapai(PERUSAHAAN\_JASA(IdPerusahaan)))

## B. Formal Model

1. KOTA(IdKota, NamaKota)
2. PERUSAHAAN\_JASA(IdPerusahaan, NamaPerusahaan, NomorTelp, Pemilik, Tahun-Berdiri, JenisPerusahaan)
3. BANDARA(IdBandara, NamaBandara, JenisBandara, Gate, Terminal, Kota(KOTA(IdKota)))
4. PELABUHAN(IdPelabuhan, NamaPelabuhan, Kota(KOTA(IdKota)))
5. STASIUN(IdStasiun, NamaStasiun, Peron, Kota(KOTA(IdKota)))
6. TERMINAL(IdTerminal, NamaTerminal, Kota(KOTA(IdKota)))
7. PESAWAT(NoLambung, JenisPesawat, KapasitasBagasiReg)
8. KAPAL(NoLambung, NamaKapal)
9. KERETA(IdKereta, NamaKereta)
10. BUS(NomorPlat, MerkBus, Kelas, Toilet, AC, StopKontak, FormatKursi, KapasitasBus)
11. BAGASI\_TAMBAHAN(NomorLambung(PESAWAT(NoLambung)), BagasiTambahan, Harga)
12. KABIN\_PESAWAT(NoLambung(PESAWAT(NoLambung)), IdKabin, Kelas, Format-Kursi, Kapasitas)
13. KABIN\_KAPAL(NoLambung(KAPAL(NoLambung)), IdKabin, Kelas, Kapasitas)
14. GERBONG\_KERETA(IdKereta(KERETA(IdKereta)), IdGerbong, Kelas, SubKelas, Kapasitas)
15. PENERBANGAN(IdPenerbangan, TglBerangkat, WaktuBoarding, TglTiba, WaktuTiba, TarifDewasa, TarifAnak, JenisRute, DariKota(KOTA(IdKota)), KeKota(KOTA(IdKota)), Maskapai(PERUSAHAAN\_JASA(IdPerusahaan)), Pesawat(PESAWAT(NoLambung)))
16. PELAYARAN(IdPelayaran, TglBerangkat, WaktuBerangkat, TglTiba, WaktuTiba, TarifDewasa, TarifAnak, JenisRute, DariKota(KOTA(IdKota)), KeKota(KOTA(IdKota)), Penyedia(PERUSAHAAN\_JASA(IdPerusahaan)), Kapal(KAPAL(NoLambung)))
17. PERJALANAN\_KERETA(IdPerjalananKereta, TglBerangkat, WaktuBerangkat, TglTiba, WaktuTiba, TarifDewasa, TarifAnak, JenisRute, DariKota(KOTA(IdKota)), KeKota(KOTA(IdKota)), Penyedia(PERUSAHAAN\_JASA(IdPerusahaan)), Kereta(KERETA(IdKereta)))
18. PERJALANAN\_BUS(IdPerjalananBus, TglBerangkat, WaktuBerangkat, TglTiba, WaktuTiba, TarifDewasa, TarifAnak, JenisRute, DariKota(KOTA(IdKota)), KeKota(KOTA(IdKota)), Penyedia(PERUSAHAAN\_JASA(IdPerusahaan)), Kereta(BUS(NomorPlat)))
19. LAYANAN\_BUS\_LOKAL(IdLayananBusLokal, WilayahOperasi(KOTA(IdKota)), Penyedia(PERUSAHAAN\_JASA(IdPerusahaan)), Kereta(BUS(NomorPlat)))
20. LAYANAN\_TAKSI(IdLayananTaksi, WilayahOperasi(KOTA(IdKota)), NomorTelepon, Penyedia(PERUSAHAAN\_JASA(IdPerusahaan)))
21. TAKSI(NomorPlat, IdLayananTaksi(LAYANAN\_TAKSI(IdLayananTaksi)))
22. LOKASI\_TRANSIT\_BUS(IdLokasi, NamaLokasi, JenisLokasi, Kota(KOTA(IdKota)))
23. RUTE\_PENERBANGAN(IdRutePenerbangan, IdPenerbangan(PENERBANGAN(IdPenerbangan)))
24. RUTE\_PENERBANGAN\_TERDIRI(IdRutePenerbangan(RUTE\_PENERBANGAN(IdRutePenerbangan)), NoTransit, IdBandara(BANDARA(IdBandara)))
25. RUTE\_PELAYARAN(IdRutePelayaran, IdPelayaran(PELAYARAN(IdPelayaran)))
26. RUTE\_PELAYARAN\_TERDIRI(IdRutePelayaran(RUTE\_PELAYARAN(IdRutePelayaran)), NoTransit, IdPelabuhan(PELABUHAN(IdPelabuhan)))
27. RUTE\_KERETA(IdRuteKereta, IdPerjalananKereta(PERJALANAN\_KERETA(IdPerjalananKereta)))

28. RUTE\_KERETA\_TERDIRI(IdRuteKereta(RUTE\_KERETA(IdRuteKereta)), NoTransit, IdStasiun(STASIUN(IdStasiun)))
29. RUTE\_BUS(IdRuteBus, IdPerjalananBus(PERJALANAN\_BUS(IdPerjalananBus)))
30. RUTE\_BUS\_TERDIRI(IdRuteBus(RUTE\_BUS(IdRuteBus)), NoTransit, IdTerminal(TERMINAL(IdTerminal)))
31. RUTE\_BUS\_LOKAL(IdRuteBusLokal, IdLayananBusLokal(LAYANAN\_BUS\_LOKAL(IdLayananBusLokal)))
32. RUTE\_BUS\_LOKAL\_TERDIRI(IdRuteBusLokal(RUTE\_BUS\_LOKAL(IdRuteBusLokal)), NoTransit, IdLokasiTransit(LOKASI\_TRANSIT\_BUS(IdLokasi)))
33. CABANG\_KOTA(IdCabangKota, NomorTelepon, TahunDibuka, IdKota(KOTA(IdKota)), IdPerusahaan(PERUSAHAAN\_JASA(IdPerusahaan)))
34. CABANG\_BANDARA(IdCabangBandara, TahunDibuka, IdBandara(BANDARA(IdBandara)), IdMaskapai(PERUSAHAAN\_JASA(IdPerusahaan)))

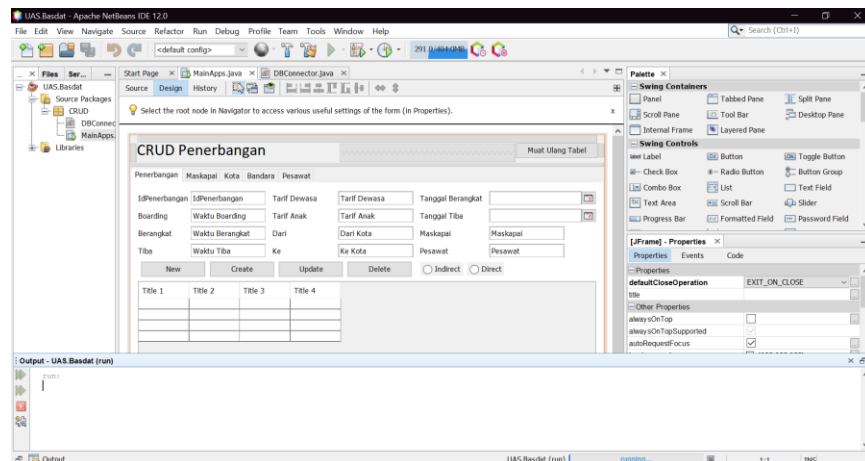
## II. IMPLEMENTASI

### A. SQL

File SQL terlampir.

### B. Implementasi

Database diimplementasikan pada aplikasi Java GUI yang dibuat menggunakan Java Swing. Aplikasi fokus fungsi Create, Read, Update, dan Delete terhadap data penerbangan. Untuk implementasinya menggunakan view yang didapat dari tabel PerusahaanJasa, Kota, Bandara, Pesawat dan tabel Penerbangan itu Sendiri. Aplikasi dibuat menggunakan IDE Netbeans.



Berikut ini tampilan awal ketika program dijalankan:

Id Penerba...	Tanggal B...	Boarding	Berangkat	Tanggal Tiba	Tiba	Tarif Dewa...	Tarif Anak	Jenis Rute
BTK-112	2021-01-01	12:25:00.00...	12:35:00.00...	2021-01-01	13:55:00.00...	230000	190000	Direct
GIA-208	2020-12-31	06:30:00.00...	06:45:00.00...	2020-12-31	07:45:00.00...	250000	200000	Direct
JT-303	2020-12-31	08:30:00.00...	08:45:00.00...	2020-12-31	09:45:00.00...	230000	190000	Direct

## 1. Read

Berikut salah satu kode untuk membaca database pada tabel PerusahaanJasa sebagai yang dipilih hanya Maskapai:

```
public void tampilTabelMaskapai() {
    conn=DBConnector.ConnectToDB("TransportasiUmum");
    try {
        String[]title = {"Kode Maskapai","Nama Maskapai","Nomor Telpon","Kepemilikan","Tanggal Berdiri"};

        String sql="SELECT * FROM ViewPerusahaan";
        model=new DefaultTableModel(null,title);
        pst=conn.prepareStatement(sql);
        rs=pst.executeQuery();
        String[]data = new String[5];

        while(rs.next()){
            data[0] = rs.getString("IdPerusahaan");
            data[1] = rs.getString("NamaPerusahaan");
            data[2] = rs.getString("NomorTelp");
            data[3] = rs.getString("Pemilik");
            data[4] = rs.getString("TahunBerdiri");

            model.addRow(data);
        }
        tableMaskapai.setModel(model);
    }
    catch (SQLException e) {
        JOptionPane.showMessageDialog(null, e.getMessage());
    }
}
```

## 2. Fungsi Create

Kode

```
try {
    String txtDirect = null;
    if(optDirect.isSelected() == false && optIndirect.isSelected() == true ) txtDirect = "Indirect";
    if(optDirect.isSelected() == true && optIndirect.isSelected() == false ) txtDirect = "Direct";
    conn=DBConnector.ConnectToDB("TransportasiUmum");
    String sql = "INSERT INTO ViewPenerbangan(IdPenerbangan, TglBerangkat, WaktuBoarding, WaktuBerangkat, TglTiba, WaktuTiba, TarifDewasa, TarifAnak, JenisRute) VALUES(?, ?, ?, ?, ?, ?, ?, ?, ?)";
    pst=conn.prepareStatement(sql);
    pst.setString(1,txtId.getText().toUpperCase());
    pst.setString(2,(((JTextField)datBerangkat.getDateEditor().getUIComponent()).getText()).toString());
    pst.setString(3,txtBoarding.getText());
    pst.setString(4,txtBerangkat.getText());
    pst.setString(5,(((JTextField)datTiba.getDateEditor().getUIComponent()).getText()).toString());
    pst.setString(6,txtTiba.getText());
    pst.setString(7,txtDewasa.getText());
    pst.setString(8,txtAnak.getText());
    pst.setString(9,txtDirect.toString());
    pst.setString(10,txtDari.getText().toUpperCase());
    pst.setString(11,txtKe.getText().toUpperCase());
    pst.setString(12,txtMaskapai.getText().toUpperCase());
    pst.setString(13,txtPesawat.getText().toUpperCase());
    pst.executeUpdate();
    tampilTabelPenerbangan();
} catch (Exception e) {
    JOptionPane.showMessageDialog(rootPane, e);
}
```

Percobaan





```

try {
    String txtDirect = null;
    if(optDirect.isSelected() == false && optIndirect.isSelected() == true ) txtDirect = "Indirect";
    if(optDirect.isSelected() == true && optIndirect.isSelected() == false ) txtDirect = "Direct";
    conn=DBConnector.ConnectToDB("TransportasiUmum");
    String sql = "UPDATE ViewPenerbangan SET "
        + "TglBerangkat = ?, "
        + "WaktuBoarding = ?, "
        + "WaktuBerangkat = ?, "
        + "TglTiba = ?, "
        + "WaktuTiba = ?, "
        + "TarifDewasa = ?, "
        + "TarifAnak = ?, "
        + "JenisRute = ?, "
        + "DariKota = ?, "
        + "KeKota = ?, "
        + "Maskapai = ?, "
        + "Pesawat = ? "
        + "WHERE IdPenerbangan = ?;";

    pst=conn.prepareStatement(sql);
    pst.setString(1,(((JTextField)datBerangkat.getDateEditor().getUiComponent()).getText()).toString());
    pst.setString(2,txtBoarding.getText());
    pst.setString(3,txtBerangkat.getText());
    pst.setString(4,(((JTextField)datTiba.getDateEditor().getUiComponent()).getText()).toString());
    pst.setString(5,txtTiba.getText());
    pst.setString(6,txtDewasa.getText());
    pst.setString(7,txtAnak.getText().toString());
    pst.setString(8,txtDirect.toString());
    pst.setString(9,txtDari.getText().toUpperCase());
    pst.setString(10,txtKe.getText().toUpperCase());
    pst.setString(11,txtMaskapai.getText().toUpperCase());
    pst.setString(12,txtPesawat.getText().toUpperCase());
    pst.setString(13,txtId.getText().toUpperCase());
    pst.execute();
    tampilTabelPenerbangan();
}

```

## Percobaan

CRUD Penerbangan

Muat Ulang Tabel

Penerbangan

Maskapai Kota Bandara Pesawat

IdPenerbangan

JT-313

Tarif Dewasa

300000

Tanggal Berangkat

2021-01-19

Boarding

19:10:00.0000000

Tarif Anak

280000

Tanggal Tiba

2021-01-19

Berangkat

19:30:00.0000000

Dari

jog

Maskapai

M-JT

Tiba

23:10:10.0000000

Ke

jkt

Pesawat

PK-101

New

Create

Update

Delete

Indirect

Direct

Id Penerba...	Tanggal B...	Boarding	Berangkat	Tanggal Tiba	Tiba	Tarif Dewa...	Tarif Anak	Jenis Rute
BTK-112	2021-01-01	12:25:00.00...	12:35:00.00...	2021-01-01	13:55:00.00...	230000	190000	Direct
GIA-208	2020-12-31	06:30:00.00...	06:45:00.00...	2020-12-31	07:45:00.00...	250000	200000	Direct
JT-303	2020-12-31	08:30:00.00...	08:45:00.00...	2020-12-31	09:45:00.00...	230000	190000	Direct
JT-313	2020-12-30	19:10:00.00...	19:30:00.00...	2020-12-30	23:10:10.00...	190000	180000	Direct

## Hasil

**CRUD Penerbangan** Muat Ulang Tabel

Penerbangan Maskapai Kota Bandara Pesawat

IdPenerbangan: JT-313 Tarif Dewasa: 300000 Tanggal Berangkat: 2021-01-19

Boarding: 19:10:00.0000000 Tarif Anak: 280000 Tanggal Tiba: 2021-01-19

Berangkat: 19:30:00.0000000 Dari: JOG Maskapai: M-JT

Tiba: 23:10:10.0000000 Ke: JKT Pesawat: PK-101

☐ Indirect ☒ Direct

Id Penerba...	Tanggal B...	Boarding	Berangkat	Tanggal Tiba	Tiba	Tarif Dewa...	Tarif Anak	Jenis Rute
BTK-112	2021-01-01	12:25:00.00...	12:35:00.00...	2021-01-01	13:55:00.00...	230000	190000	Direct
GIA-208	2020-12-31	06:30:00.00...	06:45:00.00...	2020-12-31	07:45:00.00...	250000	200000	Direct
JT-303	2020-12-31	08:30:00.00...	08:45:00.00...	2020-12-31	09:45:00.00...	230000	190000	Direct
JT-313	2021-01-19	19:10:00.00...	19:30:00.00...	2021-01-19	23:10:10.00...	300000	280000	Direct

#### 4. Delete

##### Kode

```
try {
    String txtDirect = null;
    if(optDirect.isSelected() == false && optIndirect.isSelected() == true ) txtDirect = "Indirect";
    if(optDirect.isSelected() == true && optIndirect.isSelected() == false ) txtDirect = "Direct";
    conn=DBCConnector.ConnectToDB("TransportasiUmum");
    String sql = "DELETE FROM ViewPenerbangan "
        + "WHERE IdPenerbangan = ?;"
        ;
    pst=conn.prepareStatement(sql);
    pst.setString(1,txtId.getText());
    pst.execute();
    tampilTabelPenerbangan();
} catch (Exception e) {
    JOptionPane.showMessageDialog(rootPane, e);
}
```

##### Hasil

**CRUD Penerbangan** Muat Ulang Tabel

Penerbangan Maskapai Kota Bandara Pesawat

IdPenerbangan: JT-313 Tarif Dewasa: 300000 Tanggal Berangkat: 2021-01-19

Boarding: 19:10:00.0000000 Tarif Anak: 280000 Tanggal Tiba: 2021-01-19

Berangkat: 19:30:00.0000000 Dari: JOG Maskapai: M-JT

Tiba: 23:10:10.0000000 Ke: JKT Pesawat: PK-101

☐ Indirect ☒ Direct

Id Penerba...	Tanggal B...	Boarding	Berangkat	Tanggal Tiba	Tiba	Tarif Dewa...	Tarif Anak	Jenis Rute
BTK-112	2021-01-01	12:25:00.00...	12:35:00.00...	2021-01-01	13:55:00.00...	230000	190000	Direct
GIA-208	2020-12-31	06:30:00.00...	06:45:00.00...	2020-12-31	07:45:00.00...	250000	200000	Direct
JT-303	2020-12-31	08:30:00.00...	08:45:00.00...	2020-12-31	09:45:00.00...	230000	190000	Direct

### III. REFERENSI

Tuliskan singkat sumber untuk menulis laporan ini, baik primary source ataupun secondary sources



#### A. Component Java Swing Calendar Picker

<https://toedter.com/jcalendar/>

#### B. Retrieving SQL Query Using JDBC on Java Swing Table

<https://stackoverflow.com/questions/27815400/retrieving-data-from-jdbc-database-into-jtable>

#### IV. KELOMPOK

NIM	Nama/email	Role	Tanggungjawab	TTD
M0519017	Annas Abdurrahman/ annasabdurrahman 354@student.uns.a c.id	Ketua	ER Diagram, Mapping, Implementasi Aplikasi Java Swing, Laporan	 Annas Abdurrahman
M0519004	Adianta Putratama	Anggota	ER Diagram	
M0519026	Bagus Yulfan Ardiyanto/ bagus.yulfan@stud ent.uns.ac.id	Anggota	ER Diagram, Mapping, Laporan	 Bagus Yulfan A