

**IF2240 - Basis Data**

# **Tugas Besar**

**Milestone 2: Implementasi Basis Data**



Disusun Oleh:

The Garuda Pancasila Spirit of Brr Brr Patapim

**Program Studi Teknik Informatika**

**Sekolah Teknik Elektro dan Informatika**

**Institut Teknologi Bandung**

**2025**

## IDENTITAS KELOMPOK

Nomor Kelompok : 3

Kode Kelompok : BNI

Nama Kelompok : The Garuda Pancasila Spirit of Brr Brr Patapim

Anggota Kelompok : 1. Albertus Christian Poandy (13523077)  
2. Zulfaqqar Nayaka Athadiansyah (13523094)  
3. Heleni Gratia M Tampubolon (13523107)  
4. Ferdin Arsenarendra Purtadi (13523117)  
5. Ziyen Agil Nur Ramadhan (13622076)

Kelas : K2

Dosen Pengampu : 1. Dr. Fazat Nur Azizah, S.T., M.Sc.  
2. Dr. Maya Nabila, S.Si., M.Si.

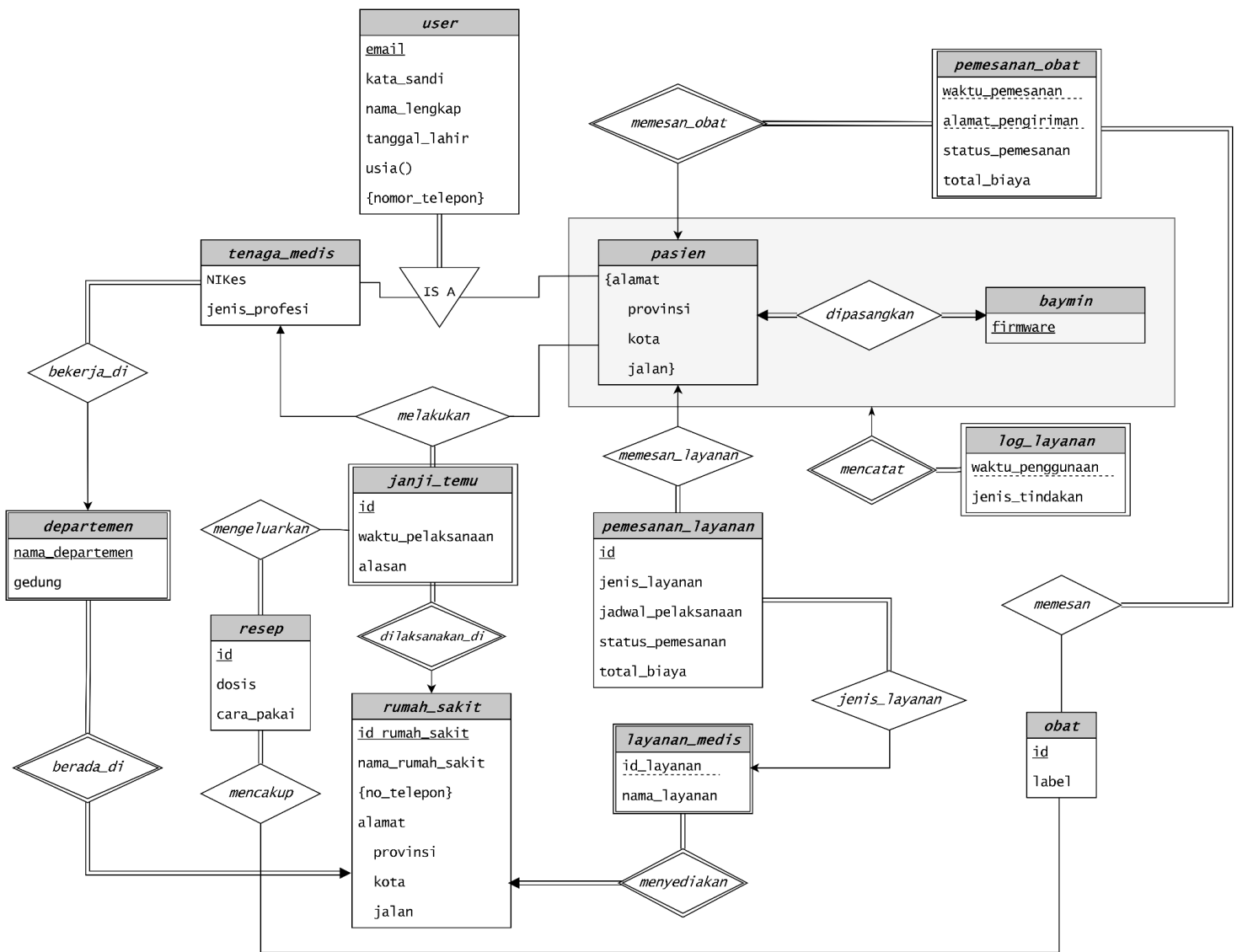
Problem Set : HaloBot

Nama Asisten : Shabrina Maharani

Tanggal Pengumpulan : 16 Mei 2025

# 1. MODEL ENTITY-RELATIONSHIP

## a. Entity-Relationship Diagram



## b. Deskripsi

Ada sejumlah revisi yang diterapkan pada versi sebelumnya pada Milestone 1:

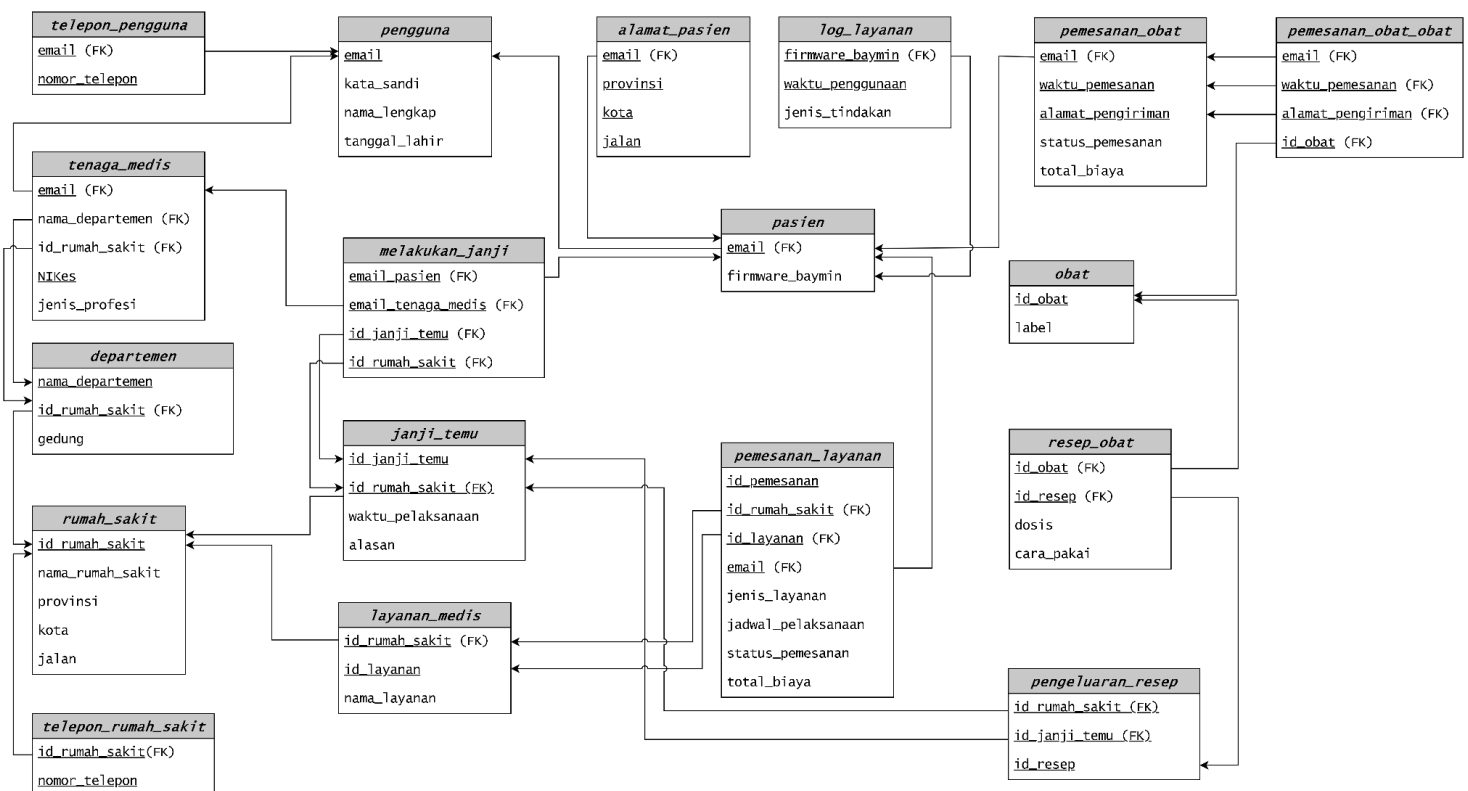
- **Partisipasi janji temu:** Seorang tenaga medis dapat menangani lebih dari satu pasien dalam sebuah janji temu, dan setiap janji temu harus melibatkan satu atau lebih pasien secara keseluruhan (partisipasi bersifat total dari sisi pasien).
- **Partisipasi departemen dan rumah sakit:** Setiap departemen harus berada di dalam satu rumah sakit, dan setiap rumah sakit

harus memiliki setidaknya satu departemen (partisipasi total di kedua arah).

- **Perubahan tipe atribut jenis\_profesi pada tabel tenaga\_medis:** Berdasarkan klarifikasi dari kakak asisten, maksud dari penjelasan tentang jenis profesi dari tenaga medis di spesifikasi bukan berarti atribut jenis profesi ditentukan oleh departemennya. Jadi, atribut tersebut tidak lagi dinyatakan sebagai *derived attribute*.

## 2. MODEL RELASIONAL

### a. Diagram Relasional



### b. Deskripsi

Model relasional ini dibangun dari pemetaan *entity relationship diagram* (ERD) sistem Halobot ke dalam bentuk tabel-tabel relasional yang saling terhubung melalui hubungan *foreign key*. Setiap entitas utama dalam sistem seperti pengguna, pasien, tenaga medis, rumah sakit, dan layanan medis direpresentasikan sebagai tabel tersendiri, sementara hubungan

antar entitas direpresentasikan melalui tabel relasi atau penambahan atribut *foreign key*.

Tabel pengguna menjadi pusat dari entitas individu, dan direlasikan ke entitas pasien dan tenaga\_medis melalui atribut email. Pasien yang merupakan pengguna dapat memesan layanan medis melalui tabel pemesanan\_layanan yang menghubungkan pasien (email) dengan layanan (id\_layanan) dan rumah sakit (id\_rumah\_sakit). Sementara itu, tenaga medis ditempatkan dalam sebuah departemen (nama\_departemen, id\_rumah\_sakit), dan memiliki hubungan dengan pasien dalam konteks konsultasi melalui entitas janji\_temu.

Relasi antara pasien dan tenaga medis dalam hal konsultasi direpresentasikan melalui tabel melakukan\_janji yang menghubungkan pasien, tenaga medis, dan janji temu secara eksplisit. Setiap janji temu juga bisa berujung pada pengeluaran resep yang dicatat dalam tabel pengeluaran\_resep dengan referensi ke id\_resep, id\_janji\_temu, dan id\_rumah\_sakit.

Sistem resep dan obat dimodelkan secara *many-to-many* menggunakan tabel antara resep\_obat yang menghubungkan entitas resep dan obat. Pasien juga dapat memesan obat secara langsung melalui tabel pemesanan\_obat, dengan daftar obat yang dipesan dicatat dalam tabel pemesanan\_obat\_obat

Hubungan antara Baymin dan pasien disatukan dalam tabel pasien dan aktivitas layanan dicatat melalui tabel log\_layanan. Tabel log layanan ini mengacu pada atribut firmware\_baymin sebagai identitas Baymin yang terhubung ke pasien.

Asumsi yang digunakan :

- Pasien dapat memesan lebih dari 1 obat dalam sekali pemesanan.
- Jenis profesi dari tenaga medis adalah antara dokter atau medis.
- Dalam janji temu, seorang tenaga medis dapat melayani banyak pasien.

- Layanan Baymin disatukan dengan pasien karena setiap pasien memiliki Baymin yang unik.
- Pemesanan layanan medis rumah sakit oleh pasien hanya bisa satu layanan untuk tiap pemesanan.
- Pengguna dan rumah sakit dianggap dapat memiliki lebih dari satu nomor telepon sehingga telepon menjadi relasi tersendiri.

Berikut adalah daftar tabel (relasi) dalam basis data Halobot beserta domain dari atribut-atributnya dan batasannya. Atribut yang digarisbawahi berperan sebagai *primary key*.

pengguna		
Data dari pengguna, baik pasien maupun tenaga medis.		
<i>Atribut</i>	<i>Tipe</i>	<i>Batasan</i>
<u>email</u>	VARCHAR(255)	PRIMARY KEY, LIKE '%_@_%._%'
kata_sandi	VARCHAR(255)	NOT NULL
nama_lengkap	VARCHAR(255)	NOT NULL
tanggal_lahir	DATE	NOT NULL
usia	INT	>= 0

telepon_pengguna		
Nomor telepon dari pengguna dengan email tertentu. Bisa <i>multiple</i> .		
<i>Atribut</i>	<i>Tipe</i>	<i>Batasan</i>
<u>email</u>	VARCHAR(255)	PRIMARY KEY, NOT NULL, FOREIGN KEY
<u>no_telepon</u>	VARCHAR(255)	NOT NULL, UNIQUE

pasien		
Data dari pasien.		
<i>Atribut</i>	<i>Tipe</i>	<i>Batasan</i>
<u>email</u>	VARCHAR(255)	FOREIGN KEY, NOT NULL
firmware_baymin	INT	NOT NULL, UNIQUE

tenaga_medis		
Data dari tenaga medis.		
<i>Atribut</i>	<i>Tipe</i>	<i>Batasan</i>
<u>email</u>	VARCHAR(255)	PRIMARY KEY, FOREIGN KEY
nama_departemen	VARCHAR(255)	FOREIGN KEY, NOT NULL
id_rumah_sakit	INT	FOREIGN KEY, NOT NULL
<u>NIKes</u>	VARCHAR(255)	PRIMARY KEY, NOT NULL, UNIQUE
jenis_profesi	VARCHAR(255)	NOT NULL

rumah_sakit		
Data dari rumah sakit .		
<i>Atribut</i>	<i>Tipe</i>	<i>Batasan</i>
<u>id_rumah_sakit</u>	INT	PRIMARY KEY, NOT NULL
nama_rumah_sakit	VARCHAR(255)	NOT NULL
provinsi	VARCHAR(255)	NOT NULL
kota	VARCHAR(255)	NOT NULL
jalan	VARCHAR(255)	NOT NULL

departemen		
Data dari departemen dalam rumah sakit.		
<i>Atribut</i>	<i>Tipe</i>	<i>Batasan</i>
<u>nama_departemen</u>	VARCHAR(255)	PRIMARY KEY
<u>id_rumah_sakit</u>	INT	FOREIGN KEY, PRIMARY KEY, NOT NULL
gedung	VARCHAR(255)	NOT NULL

telepon_rumah_sakit		
Data dari telepon untuk rumah sakit. Bisa lebih dari satu untuk tiap rumah sakit.		
<i>Atribut</i>	<i>Tipe</i>	<i>Batasan</i>
<u>id_rumah_sakit</u>	VARCHAR(255)	PRIMARY KEY, FOREIGN KEY, NOT NULL
<u>no_telepon</u>	VARCHAR(255)	PRIMARY KEY, NOT NULL, UNIQUE

layanan_medis		
Data dari layanan medis yang disediakan oleh rumah sakit.		
<i>Atribut</i>	<i>Tipe</i>	<i>Batasan</i>
<u>id_rumah_sakit</u>	INT	PRIMARY KEY, FOREIGN KEY, NOT NULL
<u>id_layanan</u>	INT	PRIMARY KEY, NOT NULL
nama_layanan	ENUM (vaksinasi, fisioterapi)	NOT NULL



janji_temu		
Data rincian dari janji temu yang dilaksanakan di rumah sakit.		
<i>Atribut</i>	<i>Tipe</i>	<i>Batasan</i>
<u>id_janji_temu</u>	INT	PRIMARY KEY, NOT NULL, AUTO INCREMENT
<u>id_rumah_sakit</u>	INT	PRIMARY KEY, FOREIGN KEY, NOT NULL
waktu_pelaksanaan	DATETIME	NOT NULL
alasan	VARCHAR(255)	NOT NULL

melakukan_janji		
Relasi yang menghubungkan pasien, tenaga medis, dan rumah sakit dalam suatu janji temu.		
<i>Atribut</i>	<i>Tipe</i>	<i>Batasan</i>
<u>email_pasien</u>	VARCHAR(255)	PRIMARY KEY, FOREIGN KEY, NOT NULL
<u>email_tenaga_medis</u>	VARCHAR(255)	PRIMARY KEY, FOREIGN KEY, NOT NULL
<u>id_janji_temu</u>	INT	PRIMARY KEY, FOREIGN KEY, NOT NULL
<u>id_rumah_sakit</u>	INT	PRIMARY KEY, FOREIGN KEY, NOT NULL

alamat_pasien		
Alamat dari pasien. Bisa lebih dari satu untuk tiap pasien.		
<i>Atribut</i>	<i>Tipe</i>	<i>Batasan</i>

<u>email</u>	VARCHAR(255)	PRIMARY KEY, FOREIGN KEY, NOT NULL
<u>provinsi</u>	VARCHAR(255)	PRIMARY KEY, NOT NULL
<u>kota</u>	VARCHAR(255)	PRIMARY KEY, NOT NULL
<u>jalan</u>	VARCHAR(255)	PRIMARY KEY, NOT NULL

log_layanan		
Log layanan dari Baymin milik pengguna.		
<i>Atribut</i>	<i>Tipe</i>	<i>Batasan</i>
<u>firmware_baymin</u>	INT	PRIMARY KEY, FOREIGN KEY, NOT NULL
waktu_penggunaa	DATETIME	NOT NULL
jenis_tindakan	VARCHAR(255)	NOT NULL

pemesanan_layanan		
Data pemesanan layanan oleh pasien pada rumah sakit.		
<i>Atribut</i>	<i>Tipe</i>	<i>Batasan</i>
<u>id_pemesanan</u>	INT	PRIMARY KEY, NOT NULL, AUTO INCREMENT
<u>id_rumah_sakit</u>	INT	PRIMARY KEY, FOREIGN KEY, NOT NULL
<u>id_layanan</u>	INT	PRIMARY KEY, FOREIGN KEY, NOT NULL
<u>email</u>	VARCHAR(255)	PRIMARY KEY,

		FOREIGN KEY, NOT NULL
jenis_layanan	VARCHAR(255)	NOT NULL
jadwal_pelaksanaan	DATE	NOT NULL
status_pemesanan	ENUM (belum dibayar, dijadwalkan, sedang berlangsung, selesai, dibatalkan)	NOT NULL
total_biaya	int	NOT NULL, >=0

pengeluaran_resep		
Relasi yang menghubungkan resep yang dikeluarkan pada janji temu yang dilaksanakan di rumah sakit yang bersesuaian.		
<i>Atribut</i>	<i>Tipe</i>	<i>Batasan</i>
<u>id_rumah_sakit</u>	INT	PRIMARY KEY, NOT NULL
<u>id_janji_temu</u>	INT	PRIMARY KEY, FOREIGN KEY, NOT NULL
<u>id_resep</u>	INT	PRIMARY KEY, NOT NULL, AUTO INCREMENT, UNIQUE

resep_obat		
Data rincian penggunaan obat dalam suatu resep.		
<i>Atribut</i>	<i>Tipe</i>	<i>Batasan</i>
<u>id_obat</u>	INT	PRIMARY KEY, NOT NULL
<u>id_resep</u>	INT	PRIMARY KEY, FOREIGN KEY, NOT

		NULL
dosis	VARCHAR(255)	NOT NULL
cara_pakai	VARCHAR(255)	NOT NULL

obat		
Data dari obat yang tersedia.		
<i>Atribut</i>	<i>Tipe</i>	<i>Batasan</i>
<u>id_obat</u>	INT	PRIMARY KEY, NOT NULL, UNIQUE
label	ENUM (analgesik, antibiotik, obat herbal)	NOT NULL

pemesanan_obat		
Data dari tiap pemesanan obat oleh pengguna dengan alamat tertentu.		
<i>Atribut</i>	<i>Tipe</i>	<i>Batasan</i>
<u>email</u>	VARCHAR(255)	PRIMARY KEY, FOREIGN KEY, NOT NULL
<u>waktu_pemesanan</u>	DATETIME	PRIMARY KEY, FOREIGN KEY, NOT NULL
<u>alamat_pengiriman</u>	VARCHAR(255)	PRIMARY KEY, FOREIGN KEY, NOT NULL
status_pemesanan	ENUM (belum dibayar, dijadwalkan, sedang berlangsung, selesai, dibatalkan)	NOT NULL
total_biaya	INT	>=0

pemesanan_obat_obat		
Relasi yang menghubungkan pemesanan obat dengan obat yang dipesan.		
<i>Atribut</i>	<i>Tipe</i>	<i>Batasan</i>
<u>email</u>	VARCHAR(255)	PRIMARY KEY, FOREIGN KEY, NOT NULL
waktu_pemesanan	DATETIME	PRIMARY KEY, FOREIGN KEY, NOT NULL
<u>alamat_pengiriman</u>	VARCHAR(255)	PRIMARY KEY, FOREIGN KEY, NOT NULL
<u>id_obat</u>	INT	PRIMARY KEY, FOREIGN KEY, NOT NULL

Selanjutnya, hubungan antara tiap tabel dideskripsikan melalui tabel berikut.

<i>Atribut Tabel</i>	<i>Atribut Tabel yang Direferensikan</i>
telepon_pengguna(email)	pengguna(email)
resep_obat(id_obat)	obat(id_obat)
resep_obat(id_resep)	pengeluaran_resep(id_resep)
pemesanan_layanan(id_rumah_sakit)	layanan_medis(id_rumah_sakit)
pemesanan_layanan(id_layanan)	layanan_medis(id_layanan)
pemesanan_layanan(email)	pasien(email)
pengeluaran_resep(id_janji_temu)	janji_temu(id_janji_temu)
pengeluaran_resep(id_rumah_sakit)	janji_temu(id_rumah_sakit)
telepon_rumah_sakit(id_rumah_sakit)	rumah_sakit(id_rumah_sakit)

departemen(id_rumah_sakit)	rumah_sakit(id_rumah_sakit)
janji_temu(id_rumah_sakit)	rumah_sakit(id_rumah_sakit)
log_layanan(firmware_baymin)	pasien(firmware_baymin)
pasien(email)	pengguna(email)
alamat_pasien(email)	pasien(email)
pemesanan_obat(email)	pasien(email)
melakukan_janji(id_janji_temu)	janji_temu(id_janji_temu)
melakukan_janji(id_rumah_sakit)	janji_temu(id_rumah_sakit)
melakukan_janji(email_tenaga_medis)	tenaga_medis(email)
melakukan_janji(email_pasien)	pasien(email)
tenaga_medis(email)	pengguna(email)
tenaga_medis(nama_departemen)	departemen(nama_departemen)
tenaga_medis(id_rumah_sakit)	departemen(id_rumah_sakit)
pemesanan_obat_obat(email)	pemesanan_obat(email)
pemesanan_obat_obat(waktu_pemesanan)	pemesanan_obat(waktu_pemesanan)
pemesanan_obat_obat(alamat_pengiriman )	pemesanan_obat(alamat_pengiriman)
pemesanan_obat_obat(id_obat)	obat(id_obat)
layanan_medis(id_rumah_sakit)	rumah_sakit(id_rumah_sakit)

### 3. IMPLEMENTASI BASIS DATA

#### a. Proses

Implementasi basis data dilakukan menggunakan sistem manajemen basis data (DBMS) MariaDB. Langkah awal pembuatan unit basis data baru Halobot, dilakukan melalui eksekusi perintah SQL standar pada

server MariaDB. Setelah unit basis data dibuat, langkah berikutnya adalah pendefinisian skema tabel. Skema ini diwujudkan melalui serangkaian perintah Data Definition Language (DDL), yang mengacu pada diagram relasional yang telah dirancang.

Setelah struktur basis data berhasil diimplementasikan dan semua tabel telah dibuat di MariaDB, dilakukan pengisian data awal, atau lebih sering disebut proses *seeding*. Metode yang dipilih adalah *programmatic seeding* dengan menggunakan *script* Python. Pustaka yang dimanfaatkan dalam skrip *seeding* ini adalah Faker. Pustaka ini dikonfigurasi untuk menggunakan lokal Indonesia, sehingga data acak yang dihasilkannya, seperti nama individu, alamat, nomor telepon, dan berbagai atribut lainnya, akan sesuai dengan konteks dan format yang lazim di Indonesia. Setelah semua data berhasil disisipkan ke dalam tabel-tabel yang relevan, keseluruhan transaksi akan di-*commit* ke basis data, sehingga basis data memiliki kumpulan data awal yang memadai dan siap untuk digunakan.

b. Hasil Implementasi

Berikut adalah query yang menampilkan hasil implementasi seluruh tabel.

Tujuan	Menampilkan daftar tabel dalam basis data Halobot
Query	SHOW tables;
Tangkapan Layar Query	
<pre>MariaDB [Halobot]&gt; SHOW tables; +-----+ +-----+   Tables_in_Halobot   +-----+   alamat_pasien          departemen             janji_temu             layanan_medis          log_layanan            melakukan_janji        obat                  pasien                 pemesanan_layanan      pemesanan_obat         pemesanan_obat_obat     pengeluaran_resep      pengguna               resep_obat             rumah_sakit            telepon_pengguna       telepon_rumah_sakit     tenaga_medis         +-----+ 18 rows in set (0.000 sec)</pre>	



Tujuan	Menampilkan seluruh <i>constraints foreign key</i> dalam basis data Halobot
Query	<b>SELECT</b> TABLE_NAME, COLUMN_NAME, CONSTRAINT_NAME, REFERENCED_TABLE_NAME, REFERENCED_COLUMN_NAME <b>FROM</b> INFORMATION_SCHEMA.KEY_COLUMN_USAGE <b>WHERE</b> TABLE_SCHEMA = 'Halobot' <b>AND</b> REFERENCED_TABLE_NAME IS NOT NULL;

### Tangkapan Layar Query

MariaDB [Halobot]> SELECT TABLE_NAME, COLUMN_NAME, CONSTRAINT_NAME, REFERENCED_TABLE_NAME, REFERENCED_COLUMN_NAME FROM INFORMATION_SCHEMA.KEY_COLUMN_USAGE WHERE TABLE_SCHEMA = 'Halobot' AND REFERENCED_TABLE_NAME IS NOT NULL;					
TABLE_NAME	COLUMN_NAME	CONSTRAINT_NAME	REFERENCED_TABLE_NAME	REFERENCED_COLUMN_NAME	
l_telepon_pengguna	l_email	l_telepon_pengguna_ibfk_1	l_pengguna	l_email	
l_resep_obat	l_id_obat	l_resep_obat_ibfk_1	l_obat	l_id_obat	
l_resep_obat	l_id_resep	l_resep_obat_ibfk_2	l_pengeluaran_resep	l_id_resep	
l_pemesanan_layanan	l_id_rumah_sakit	l_pemesanan_layanan_ibfk_1	l_layanan_medis	l_id_rumah_sakit	
l_pemesanan_layanan	l_id_layanan	l_pemesanan_layanan_ibfk_1	l_layanan_medis	l_id_layanan	
l_pemesanan_layanan	l_email	l_pemesanan_layanan_ibfk_2	l_pasien	l_email	
l_pengeluaran_resep	l_id_janji_temu	l_pengeluaran_resep_ibfk_1	l_janji_temu	l_id_janji_temu	
l_pengeluaran_resep	l_id_rumah_sakit	l_pengeluaran_resep_ibfk_1	l_janji_temu	l_id_rumah_sakit	
l_telepon_rumah_sakit	l_id_rumah_sakit	l_telepon_rumah_sakit_ibfk_1	l_rumah_sakit	l_id_rumah_sakit	
l_departemen	l_id_rumah_sakit	l_departemen_ibfk_1	l_rumah_sakit	l_id_rumah_sakit	
l_janji_temu	l_id_rumah_sakit	l_janji_temu_ibfk_1	l_rumah_sakit	l_id_rumah_sakit	
l_log_layanan	l_firmware_baymin	l_log_layanan_ibfk_1	l_pasien	l_firmware_baymin	
l_pasien	l_email	l_pasien_ibfk_1	l_pengguna	l_email	
lAlamat_pasien	l_email	lAlamat_pasien_ibfk_1	l_pasien	l_email	
l_pemesanan_obat	l_email	l_pemesanan_obat_ibfk_1	l_pasien	l_email	
l_melakukan_janji	l_id_janji_temu	l_melakukan_janji_ibfk_1	l_janji_temu	l_id_janji_temu	
l_melakukan_janji	l_id_rumah_sakit	l_melakukan_janji_ibfk_1	l_janji_temu	l_id_rumah_sakit	
l_melakukan_janji	l_email_tenaga_medis	l_melakukan_janji_ibfk_2	l_tenaga_medis	l_email	
l_melakukan_janji	l_email_pasien	l_melakukan_janji_ibfk_3	l_pasien	l_email	
l_tenaga_medis	l_email	l_tenaga_medis_ibfk_1	l_pengguna	l_email	
l_tenaga_medis	l_nama_departemen	l_tenaga_medis_ibfk_2	l_departemen	l_nama_departemen	
l_tenaga_medis	l_id_rumah_sakit	l_tenaga_medis_ibfk_2	l_departemen	l_id_rumah_sakit	
l_pemesanan_obat_obat	l_email	l_pemesanan_obat_obat_ibfk_1	l_pemesanan_obat	l_email	
l_pemesanan_obat_obat	l_waktu_pemesanan	l_pemesanan_obat_obat_ibfk_1	l_pemesanan_obat	l_waktu_pemesanan	
l_pemesanan_obat_obat	lAlamat_pengiriman	l_pemesanan_obat_obat_ibfk_1	l_pemesanan_obat	lAlamat_pengiriman	
l_pemesanan_obat_obat	l_id_obat	l_pemesanan_obat_obat_ibfk_2	l_obat	l_id_obat	
l_layanan_medis	l_id_rumah_sakit	l_layanan_medis_ibfk_1	l_rumah_sakit	l_id_rumah_sakit	
27 rows in set (0.002 sec)					

Tujuan	Menampilkan jumlah baris pada tiap tabel dalam Halobot
Query	SELECT table_name, table_rows FROM information_schema.tables WHERE table_schema = 'Halobot';

#### Tangkapan Layar Query

```
MariaDB [Halobot]> SELECT table_name, table_rows
-> FROM information_schema.tables
-> WHERE table_schema = 'Halobot';
```

```
+-----+-----+
| table_name          | table_rows |
+-----+-----+
| telepon_pengguna    |          300 |
| rumah_sakit         |           20 |
| resep_obat          |          100 |
| obat               |           20 |
| pemesanan_layanan  |          100 |
| pengeluaran_resep   |           50 |
| telepon_rumah_sakit |           50 |
| departemen          |           50 |
| janji_temu          |           50 |
| log_layanan         |          100 |
| pasien              |           50 |
| alamat_pasien       |          100 |
| pemesanan_obat      |           50 |
| melakukan_janji     |          150 |
| pengguna            |          150 |
| tenaga_medis        |          100 |
| pemesanan_obat_obat |          100 |
| layanan_medis       |           50 |
+-----+-----+
```

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

Tujuan	Menampilkan deskripsi dari tabel pengguna
Query	describe pengguna;
Tangkapan Layar Query	
<pre> MariaDB [Halobot]&gt; DESCRIBE pengguna; +-----+-----+-----+-----+-----+-----+   Field            Type            Null   Key   Default   Extra   +-----+-----+-----+-----+-----+-----+   email            varchar(255)    NO     PRI   NULL                kata_sandi       varchar(255)    NO           NULL                nama_lengkap     varchar(255)    NO           NULL                tanggal_lahir   date            NO           NULL              +-----+-----+-----+-----+-----+-----+ 4 rows in set (0.003 sec) </pre>	

Tujuan	Menampilkan deskripsi dari tabel tenaga_medis
Query	describe tenaga_medis;
Tangkapan Layar Query	
<pre> MariaDB [halobot]&gt; describe tenaga_medis; +-----+-----+-----+-----+-----+-----+   Field            Type            Null   Key   Default   Extra   +-----+-----+-----+-----+-----+-----+   email            varchar(255)    NO     PRI   NULL                nama_departemen   varchar(255)    NO     MUL   NULL                id_rumah_sakit   int(11)          NO           NULL                NIKes            varchar(255)    NO     PRI   NULL                jenis_profesi     varchar(255)    NO           NULL              +-----+-----+-----+-----+-----+-----+ 5 rows in set (0.034 sec) </pre>	

Tujuan	Menampilkan deskripsi dari tabel pasien
Query	describe pasien;
Tangkapan Layar Query	
<pre> MariaDB [halobot]&gt; describe pasien; +-----+-----+-----+-----+-----+-----+   Field            Type            Null   Key   Default   Extra   +-----+-----+-----+-----+-----+-----+   email            varchar(255)    NO     MUL   NULL                 firmware_baymin   int(11)         NO     PRI   NULL               +-----+-----+-----+-----+-----+-----+ 2 rows in set (0.036 sec) </pre>	

Tujuan	Menampilkan deskripsi dari tabel alamat_pasien
Query	describe alamat_pasien;
Tangkapan Layar Query	
<pre> MariaDB [halobot]&gt; describe alamat_pasien; +-----+-----+-----+-----+-----+-----+   Field            Type            Null   Key   Default   Extra   +-----+-----+-----+-----+-----+-----+   email            varchar(255)    NO     PRI   NULL                 provinsi         varchar(100)    NO     PRI   NULL                 kota             varchar(100)    NO     PRI   NULL                 jalan            varchar(100)    NO     PRI   NULL               +-----+-----+-----+-----+-----+-----+ 4 rows in set (0.029 sec) </pre>	

Tujuan	Menampilkan deskripsi dari tabel departemen
Query	describe departemen;
Tangkapan Layar Query	

```

MariaDB [halobot]> describe departemen;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| nama_departemen | varchar(255)  | NO   | PRI | NULL    |       |
| id_rumah_sakit  | int(11)       | NO   | PRI | NULL    |       |
| gedung          | varchar(255)  | NO   |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.025 sec)

```

Tujuan	Menampilkan deskripsi dari tabel janji_temu
Query	describe janji_temu;
Tangkapan Layar Query	
<pre> MariaDB [halobot]&gt; describe janji_temu; +-----+-----+-----+-----+-----+-----+   Field            Type            Null   Key   Default   Extra   +-----+-----+-----+-----+-----+-----+   id_janji_temu     int(11)         NO     PRI   NULL      auto_increment     id_rumah_sakit    int(11)         NO     PRI   NULL                        waktu_pelaksanaan   datetime        NO           NULL                        alasan            varchar(255)    NO           NULL                      +-----+-----+-----+-----+-----+-----+ 4 rows in set (0.022 sec) </pre>	

Tujuan	Menampilkan deskripsi dari tabel layanan_medis
Query	describe layanan_medis;
Tangkapan Layar Query	
<pre> MariaDB [halobot]&gt; describe layanan_medis; +-----+-----+-----+-----+-----+-----+   Field            Type            Null   Key   Default   Extra   +-----+-----+-----+-----+-----+-----+   id_rumah_sakit   int(11)         NO     PRI   NULL                id_layanan       int(11)         NO     PRI   NULL                nama_layanan     enum('vaksinasi','fisioterapi')   NO           NULL              +-----+-----+-----+-----+-----+-----+ 3 rows in set (0.024 sec) </pre>	

Tujuan	Menampilkan deskripsi dari tabel log_layanan
Query	describe log_layanan;
Tangkapan Layar Query	

MariaDB [halobot]> describe log_layanan;						
Field	Type	Null	Key	Default	Extra	
firmware_baymin	int(11)	NO	PRI	NULL		
waktu_pengguna	datetime	NO	PRI	NULL		
jenis_tindakan	varchar(255)	NO		NULL		
3 rows in set (0.027 sec)						

Tujuan	Menampilkan deskripsi dari tabel melakukan_janji
Query	describe melakukan_janji;

#### Tangkapan Layar Query

MariaDB [halobot]> describe melakukan_janji;						
Field	Type	Null	Key	Default	Extra	
email_pasien	varchar(255)	NO	PRI	NULL		
id_janji_temu	int(11)	NO	PRI	NULL		
email_tenaga_medis	varchar(255)	NO	PRI	NULL		
id_rumah_sakit	int(11)	NO	PRI	NULL		
4 rows in set (0.036 sec)						

Tujuan	Menampilkan deskripsi dari tabel obat
Query	describe obat;

#### Tangkapan Layar Query

MariaDB [halobot]> describe obat;						
Field	Type	Null	Key	Default	Extra	
id_obat	int(11)	NO	PRI	NULL		
label	enum('analgesik','antibiotik','obat herbal')	NO		NULL		
2 rows in set (0.035 sec)						

Tujuan	Menampilkan deskripsi dari tabel pemesanan_layanan
Query	describe pemesanan_layanan;

#### Tangkapan Layar Query

MariaDB [halobot]> describe pemesanan_layanan;						
Field	Type	Null	Key	Default	Extra	
id_pemesanan	int(11)	NO	PRI	NULL	auto_increment	
id_rumah_sakit	int(11)	NO	PRI	NULL		
id_layanan	int(11)	NO	PRI	NULL		
email	varchar(255)	NO	PRI	NULL		
jenis_layanan	varchar(255)	NO		NULL		
jadwal_pelaksanaan	date	NO		NULL		
status_pemesanan	enum('belum dibayar','dijadwalkan','sedang berlangsung','selesai','dibatalkan')	NO		NULL		
total_biaya	int(11)	NO		NULL		
8 rows in set (0.052 sec)						

Tujuan	Menampilkan deskripsi dari tabel pemesanan_obat
Query	describe pemesanan_obat;
Tangkapan Layar Query	
<pre> MariaDB [halobot]&gt; describe pemesanan_obat; +-----+-----+-----+-----+-----+-----+   Field   Type   Null   Key   Default   Extra   +-----+-----+-----+-----+-----+-----+   email   varchar(255)   NO   PRI   NULL       waktu_pemesanan   datetime   NO   PRI   NULL       alamat_pengiriman   varchar(255)   NO   PRI   NULL       status_pemesanan   enum('belum dibayar','dijadwalkan','sedang berlangsung','selesai','dibatalkan')   NO     NULL       total_biaya   int(11)   NO     NULL     +-----+-----+-----+-----+-----+-----+ 5 rows in set (0.035 sec) </pre>	

Tujuan	Menampilkan deskripsi dari tabel pemesanan_obat_obat
Query	describe pemesanan_obat_obat;
Tangkapan Layar Query	
<pre> MariaDB [halobot]&gt; describe pemesanan_obat_obat; +-----+-----+-----+-----+-----+-----+   Field   Type   Null   Key   Default   Extra   +-----+-----+-----+-----+-----+-----+   email   varchar(255)   NO   PRI   NULL       waktu_pemesanan   datetime   NO   PRI   NULL       alamat_pengiriman   varchar(255)   NO   PRI   NULL       id_obat   int(11)   NO   PRI   NULL     +-----+-----+-----+-----+-----+-----+ 4 rows in set (0.037 sec) </pre>	

Tujuan	Menampilkan deskripsi dari tabel pengeluaran_resep
Query	describe pengeluaran_resep;
Tangkapan Layar Query	

```
MariaDB [halobot]> describe pengeluaran_resep;
```

Field	Type	Null	Key	Default	Extra
id_janji_temu	int(11)	NO	PRI	NULL	
id_rumah_sakit	int(11)	NO	PRI	NULL	
id_resep	int(11)	NO	PRI	NULL	

```
3 rows in set (0.031 sec)
```

Tujuan	Menampilkan deskripsi dari tabel resep_obat
Query	Describe resep_obat;

#### Tangkapan Layar Query

```
MariaDB [halobot]> describe resep_obat;
```

Field	Type	Null	Key	Default	Extra
id_obat	int(11)	NO	PRI	NULL	
id_resep	int(11)	NO	PRI	NULL	
dosis	varchar(255)	YES		NULL	
cara_pakai	varchar(255)	NO		NULL	

```
4 rows in set (0.033 sec)
```

Tujuan	Menampilkan deskripsi dari tabel rumah_sakit
Query	Describe rumah_sakit;

#### Tangkapan Layar Query

```
MariaDB [halobot]> describe rumah_sakit;
```

Field	Type	Null	Key	Default	Extra
id_rumah_sakit	int(11)	NO	PRI	NULL	
nama_rumah_sakit	varchar(255)	NO		NULL	
provinsi	varchar(255)	NO		NULL	
kota	varchar(255)	NO		NULL	
jalan	varchar(255)	NO		NULL	

```
5 rows in set (0.045 sec)
```



Tujuan	Menampilkan deskripsi dari tabel telepon_pengguna
Query	Describe telepon_pengguna;
Tangkapan Layar Query	
<pre> MariaDB [halobot]&gt; describe telepon_pengguna; +-----+-----+-----+-----+-----+-----+   Field        Type            Null   Key   Default   Extra   +-----+-----+-----+-----+-----+-----+   email        varchar(255)    NO     PRI   NULL                no_telepon   varchar(255)    NO     PRI   NULL              +-----+-----+-----+-----+-----+-----+ 2 rows in set (0.036 sec) </pre>	

Tujuan	Menampilkan deskripsi dari tabel telepon_rumah_sakit
Query	Describe telepon_rumah_sakit;
Tangkapan Layar Query	
<pre> MariaDB [halobot]&gt; describe telepon_rumah_sakit; +-----+-----+-----+-----+-----+-----+   Field            Type            Null   Key   Default   Extra   +-----+-----+-----+-----+-----+-----+   id_rumah_sakit   int(11)         NO     PRI   NULL                no_telepon       varchar(255)    NO     PRI   NULL              +-----+-----+-----+-----+-----+-----+ 2 rows in set (0.030 sec) </pre>	

#### 4. PENGAMBILAN INSIGHT BASIS DATA

##### a. Query sederhana

Tujuan	<p>Wak Rusdi terpental ke Universe-135 karena bencana kosmik. Konon katanya, di Universe-135 ini Gubernur Jawa Barat-nya terlalu sibuk membuat konten pencitraan di YouTube (tidak seperti di Universe kita) sehingga lupa memperhatikan kualitas fasilitas kesehatan publik sampai-sampai masih ada warga Jawa Barat yang lebih memilih untuk melakukan janji temu pada rumah sakit di provinsi lain.</p> <p>Untuk membuktikan klaim ini, Wak Rusdi meretas database Halobot. Bantulah ia menemukan email dari pasien yang mempunyai alamat dari Jawa Barat dan memenuhi kriteria tadi. Sertakan juga provinsi dari rumah sakitnya.</p>
Query	<pre>SELECT DISTINCT ap.email, rs.provinsi FROM alamat_pasien ap JOIN melakukan_janji mj ON ap.email = mj.email_pasien JOIN janji_temu jt ON mj.id_janji_temu = jt.id_janji_temu JOIN rumah_sakit rs ON jt.id_rumah_sakit = rs.id_rumah_sakit WHERE ap.provinsi = 'Jawa Barat' AND ap.provinsi &lt;&gt; rs.provinsi;</pre>

##### Tangkapan Layar Query

```
MariaDB [Halobot]> SELECT DISTINCT ap.email, rs.provinsi
-> FROM alamat_pasien ap
-> JOIN melakukan_janji mj ON ap.email = mj.email_pasien
-> JOIN janji_temu jt ON mj.id_janji_temu = jt.id_janji_temu
-> JOIN rumah_sakit rs ON jt.id_rumah_sakit = rs.id_rumah_sakit
-> WHERE ap.provinsi = 'Jawa Barat'
-> AND ap.provinsi <> rs.provinsi;

+-----+-----+
| email                                | provinsi                                |
+-----+-----+
| sihombingqueen@example.org          | Kepulauan Bangka Belitung            |
| sihombingqueen@example.org          | Nusa Tenggara Barat                  |
| omahendra@example.com               | Kalimantan Timur                    |
| sihombingqueen@example.org          | Kalimantan Timur                    |
| hmayasari@example.net               | DI Yogyakarta                       |
| hmayasari@example.net               | Kalimantan Barat                    |
| hmayasari@example.net               | Bengkulu                             |
| omahendra@example.com               | Aceh                                 |
| sihombingqueen@example.org          | DI Yogyakarta                       |
+-----+-----+

9 rows in set (0.001 sec)
```

b. Query dengan *set operation*

Tujuan	<p>Tahun 2025, bumi tidak seperti dulu lagi. Sistem layanan kesehatan global telah terintegrasi dengan teknologi bernama Halobot, yang mengatur segala hal mulai dari jadwal pelayanan, pembelian obat, penjadwalan layanan, dan lain-lain.</p> <p>Semuanya tampak sempurna hingga Unit Audit Kesehatan Dunia (UAKD) menemukan celah dalam sistem: pasien yang melakukan pemesanan obat atau layanan dengan waktu pemesanan atau jadwal pelaksanaan setelah tanggal 10 Mei 2025, namun tidak pernah menyelesaikan prosesnya. Mereka tercatat dalam data, tetapi jejak akhirnya menghilang (tidak dibatalkan maupun diselesaikan). Pasien-pasien ini kemudian dijuluki sebagai 'Pasien Bayangan'.</p> <p>Untuk mengungkap siapa mereka, bantulah UAKD mengidentifikasi nama-nama pasien dan emailnya untuk dapat dihubungi dengan aktivitas menggantung ini, agar sistem bisa dievaluasi apakah masalah tersebut disebabkan oleh kesalahan proses, kelalaian pasien, atau gangguan pada sistem Halobot itu sendiri.</p>
Query	<pre> SELECT DISTINCT p.nama_lengkap, p.email FROM pengguna AS p INNER JOIN pasien AS pa ON p.email = pa.email INNER JOIN pemesanan_obat AS po ON po.email = pa.email WHERE DATE(po.waktu_pemesanan) &gt; '2025-05-10'  UNION  SELECT DISTINCT p.nama_lengkap, p.email FROM pengguna AS p INNER JOIN pasien AS pa ON p.email = pa.email INNER JOIN pemesanan_layanan AS pl ON pl.email = pa.email WHERE DATE(pl.jadwal_pelaksanaan) &gt; '2025-05-10'  EXCEPT  SELECT DISTINCT p.nama_lengkap, p.email FROM pengguna AS p INNER JOIN pasien AS pa ON p.email = pa.email INNER JOIN pemesanan_layanan AS pl ON pl.email = pa.email WHERE pl.status_pemesanan LIKE 'dibatalkan'  EXCEPT </pre>


	SELECT DISTINCT p.nama_lengkap, p.email FROM pengguna AS p INNER JOIN pasien AS pa ON p.email = pa.email INNER JOIN pemesanan_obat AS po ON po.email = pa.email WHERE po.status_pemesanan LIKE 'selesai';
Tangkapan Layar Query	
<pre> +-----+-----+   nama_lengkap   email   +-----+-----+   dr. Wulan Halimah   maheswarabaktianto@example.net     Hj. Nova Nugroho   opudjiastuti@example.com     Imam Nababan   vinopuspita@example.com     Siska Sitorus   ynainggolan@example.net     Yulia Haryanto   bagus17@example.org     Dr. Hana Laksita, S.IP   digdaya32@example.net     Maras Sihotang   dsaragih@example.com     R.A. Ika Gunarto, S.Sos   dsinaga@example.org     Ir. Damar Palastri   ganda49@example.org     Hj. Gawati Wibisono, S.T.   ganjaranhasanah@example.com     Yessi Mangunsong   jmustofa@example.org     Drs. Gatot Usamah   kpurwanti@example.org     Vanya Widiastuti   lantaranwaluyo@example.org     Sutan Wahyu Purwanti   limar98@example.org     Cut Olivia Dabukke   nnasyiah@example.com     Ajimin Yulianti, S.Kom   omahendra@example.com     Prabowo Utami   pradanayessi@example.net     R. Kurnia Gunawan   prasetyabella@example.net     Mila Waluyo, M.TI.   rahmanlaksita@example.net     R.M. Latif Lailasari   samosirrosmann@example.com     Chandra Damanik   sihombingqueen@example.org     Ophelia Dongoran   tania31@example.net     Syahrini Mandasari   vanyafujiati@example.org     Mahesa Prasasta   wgunarto@example.com   +-----+-----+ 24 rows in set (0.009 sec) </pre>	

c. Query dengan agregasi

Tujuan	Mengidentifikasi pasien yang menunjukkan keterlibatan aktif dalam sistem pelayanan kesehatan, khususnya mereka yang telah memesan setidaknya dua jenis layanan medis yang berbeda dan juga menerima resep obat. Analisis ini akan berfokus pada pasien yang menjalani proses layanan kesehatan secara berkelanjutan dari tahap awal konsultasi hingga tindak lanjut pengobatan. Hasil dari analisis nantinya dimanfaatkan untuk mengevaluasi efektivitas jalur layanan yang ada, sekaligus mengidentifikasi pola penggunaan layanan yang
--------	--

	konsisten untuk pengembangan strategi pelayanan lanjutan.
Query	<pre> SELECT p.email AS pasien_email, COUNT(DISTINCT pl.id_layanan) AS jumlah_layanan, COUNT(DISTINCT ro.id_obat) AS jumlah_obat FROM pasien p JOIN pemesanan_layanan pl ON p.email = pl.email JOIN melakukan_janji mj ON p.email = mj.email_pasien JOIN janji_temu jt ON mj.id_janji_temu = jt.id_janji_temu JOIN pengeluaran_resep pr ON jt.id_janji_temu = pr.id_janji_temu JOIN resep_obat ro ON pr.id_resep = ro.id_resep JOIN obat o ON ro.id_obat = o.id_obat GROUP BY p.email HAVING COUNT(DISTINCT pl.id_layanan) &gt;= 2 AND COUNT(DISTINCT ro.id_obat) &gt;= 1 ORDER BY jumlah_layanan DESC; </pre>
Tangkapan Layar Query	
<pre> +-----+-----+-----+   pasien_email   jumlah_layanan   jumlah_obat   +-----+-----+-----+   rahayu87@example.org   5   5     silviariyanti@example.org   5   4     wartaprasetya@example.org   4   11     limar53@example.net   3   4     rahmanlaksita@example.net   3   12     gunawanrama@example.com   3   4     cager91@example.org   3   2     hmayasari@example.net   3   9     sihombingqueen@example.org   3   6     xananauwais@example.org   3   4     jagaraga55@example.com   3   6     uaryani@example.com   3   7     lantarwaluyo@example.org   2   2     vinopuspita@example.com   2   6     baktionopalastri@example.net   2   6     maheswarabaktianto@example.net   2   12     samosirrosman@example.com   2   10     wastutikemal@example.net   2   2     hadiyolanda@example.com   2   5     sarah91@example.org   2   14     capanurdiyanti@example.net   2   6     omahendra@example.com   2   2     yolandakani@example.net   2   4     jmustofa@example.org   2   7   +-----+-----+-----+ 24 rows in set (0.014 sec) </pre>	

d. Query dengan *subquery*

Tujuan	Dilaporkan bahwa telah terjadi malapraktik oleh seorang tenaga medis yang tidak bertanggung jawab. Malapraktik ini terjadi di sebuah rumah sakit RS Tbk Bandung dengan layanan medis fisioterapi. Untuk memberikan kompensasi kepada korban, diperlukan nama serta email pasien. Bantulah untuk menampilkan nama dan email pasien yang setidaknya pernah memesan layanan 'fisioterapi' di rumah sakit 'RS Tbk Bandung'.
Query	<pre> SELECT p.nama_lengkap,p.email FROM pengguna p JOIN pasien ps ON p.email = ps.email WHERE EXISTS (   SELECT 1   FROM pemesanan_layanan pl   JOIN layanan_medis lm ON pl.id_layanan = lm.id_layanan   JOIN rumah_sakit rs ON pl.id_rumah_sakit = rs.id_rumah_sakit   WHERE pl.email = ps.email   AND lm.nama_layanan = 'fisioterapi'   AND rs.nama_rumah_sakit = 'RS Tbk Bandung' ); </pre>
Tangkapan Layar Query	
 <pre> +-----+-----+   nama_lengkap   email   +-----+-----+   Maras Sihotang   dsaragih@example.com     R.A. Ika Gunarto, S.Sos   dsinaga@example.org     Vanya Widiastuti   lantaranwaluyo@example.org     Jati Saefullah   maida92@example.com     Samiah Nasyiah, S.I.Kom   rahayu87@example.org     Ir. Mitra Nurdiyanti, S.T.   wartaprasetya@example.org   +-----+-----+ 6 rows in set (0.001 sec) </pre>	

e. Query kompleks

Tujuan	Kasus malpraktik dipecahkan. Ternyata, terdapat dua oknum di dua tempat berbeda yang melakukan malapraktik. Tenaga medis yang terlibat adalah Laras Hartati dan Rahayu Waluyo. Untungnya, Laras Hartati baru melakukan praktik di RS Tbk Bandung dan Rahayu Waluyo baru melakukan praktik di RS Tbk Malang. Untuk mengantisipasi kesalahan pemberian resep, diperlukan data-data seperti nama pasien, alamat pengiriman obat, id_resep, dan cara pakainya. Pasien yang menjadi prioritas pengamatan adalah anak-anak di bawah 15 tahun serta lansia di atas 50 tahun karena
--------	---

	<p>kerentanan mereka terhadap obat.</p> <p>Tugas:</p> <p>Tampilkan nama pasien, alamat pengiriman sebuah pemesanan obat, id_resep, dan cara pakai resep yang pernah dipesan oleh pasien yang pernah melakukan janji temu di rumah sakit “RS Tbk Bandung” dengan tenaga medis bernama Laras Hartati atau pernah melakukan janji temu di rumah sakit RS Tbk Malang dengan tenaga medis bernama Rahayu Waluyo dengan usia pasien kurang dari 15 tahun dan lebih dari 50 tahun.</p>
Query	<pre> SELECT   pu.nama_lengkap AS nama_pasien,   po.alamat_pengiriman,   ro.id_resep,   ro.cara_pakai FROM pemesanan_obat_obat po JOIN pasien ps          ON po.email          = ps.email JOIN pengguna pu        ON ps.email          = pu.email JOIN resep_obat ro      ON po.id_obat        = ro.id_obat JOIN pengeluaran_resep pr ON pr.id_resep      = ro.id_resep WHERE   pr.id_janji_temu IN (     -- Pasien di RS Tbk Bandung oleh Laras Hartati     SELECT mj.id_janji_temu     FROM melakukan_janji mj     JOIN janji_temu jt   ON mj.id_janji_temu   = jt.id_janji_temu AND mj.id_rumah_sakit = jt.id_rumah_sakit     JOIN rumah_sakit rs  ON jt.id_rumah_sakit   = rs.id_rumah_sakit     JOIN tenaga_medis tm ON mj.email_tenaga_medis = tm.email     JOIN pengguna pm    ON tm.email            = pm.email     WHERE rs.nama_rumah_sakit = 'RS Tbk Bandung'     AND pm.nama_lengkap      = 'Laras Hartati'      UNION      -- Pasien di RS Tbk Malang oleh Rahayu Waluyo     SELECT mj.id_janji_temu     FROM melakukan_janji mj     JOIN janji_temu jt   ON mj.id_janji_temu   = jt.id_janji_temu AND mj.id_rumah_sakit = jt.id_rumah_sakit     JOIN rumah_sakit rs  ON jt.id_rumah_sakit   = rs.id_rumah_sakit     JOIN tenaga_medis tm ON mj.email_tenaga_medis = tm.email     JOIN pengguna pm    ON tm.email            = pm.email     WHERE rs.nama_rumah_sakit = 'RS Tbk Malang'     AND pm.nama_lengkap      = 'Rahayu Waluyo'   )   AND (     TIMESTAMPDIFF(YEAR, pu.tanggal_lahir, CURDATE()) &lt; 15     OR     TIMESTAMPDIFF(YEAR, pu.tanggal_lahir, CURDATE()) &gt; 50   ); </pre>
Tangkapan Layar Query	

```
+-----+-----+-----+-----+
| nama_pasien      | alamat_pengiriman      | id_resep | cara_pakai |
+-----+-----+-----+-----+
| Yulia Haryanto   | Gg. Dipatiukur No. 117, Pariaman, Jawa Timur 06545 | 38414 | Sebelum makan |
| Hana Puspasari   | Jalan Ahmad Yani No. 093, Pasuruan, KU 01004 | 38414 | Sebelum makan |
| Nadia Nababan    | Gg. Yos Sudarso No. 301, Solok, Sulawesi Tengah 21368 | 16191 | Saat makan |
| Nadia Nababan    | Gg. Yos Sudarso No. 301, Solok, Sulawesi Tengah 21368 | 16191 | Sesudah makan |
| Vanya Widiastuti | Jalan Moch. Toha No. 1, Salatiga, DKI Jakarta 74954 | 38414 | Sebelum makan |
| Rsmadi Hassanah  | Gg. Moch. Ramdan No. 752, Bukittinggi, JK 75268 | 16191 | Sesudah makan |
| dr. Wulan Halimah | Jalan KH Amin Jasuta No. 55, Tasikmalaya, Jawa Barat 32819 | 38414 | Sebelum makan |
| Bakti Saptono, M.Pd | Gang Kutai No. 9, Kupang, DKI Jakarta 04204 | 38414 | Sebelum makan |
| Imam Nababan    | Gg. H.J Maemunah No. 39, Binjai, Papua 71340 | 38414 | Sebelum makan |
| Ir. Mitra Nurdiganti, S.T. | Jl. Cempaka No. 727, Metro, Sumatera Barat 55676 | 38414 | Sebelum makan |
+-----+-----+-----+-----+
10 rows in set (0.003 sec)
```



## LAMPIRAN

### ASISTENSI MODEL ER

#### a. Deskripsi Asistensi

Nomor Kelompok	3
Anggota yang hadir saat wawancara	<ol style="list-style-type: none"> <li>1. Albertus Christian Poandy (13523077)</li> <li>2. Zulfaqqar Nayaka Athadiansyah (13523094)</li> <li>3. Heleni Gratia M Tampubolon (13523107)</li> <li>4. Ferdin Arsenarendra Purtadi (13523117)</li> <li>5. Ziyen Agil Nur Ramadhan (13622076)</li> </ol>
Tempat, tanggal	Kamis, 8 Mei 2025
Waktu	15.00 -15.30 WIB
Nama Asisten	Shabrina Maharani

#### b. Notulensi

No.	Daftar Revisi
1.	Untuk janji temu, satu tenaga medis dapat tergabung dalam janji temu untuk banyak pasien dan partisipasi janji temu bersifat total.
2	Partisipasi antara Departemen dan Rumah Sakit seharusnya total
3	Penjelasan tentang jenis profesi dari tenaga medis di spek maksudnya bukan berarti jenis profesi ditentukan berdasarkan departemen (karena tiap departemen bisa mempunyai baik dokter maupun perawat), melainkan data tenaga medis harus menyimpan jenis profesi dan departemennya.

#### c. Dokumentasi

The screenshot displays a presentation slide with an ER diagram and a video call interface. The ER diagram shows the following entities and relationships:

- tenaga\_medis** (Entity): Attributes include `tanggal_lahir`, `usia()`, and `{nomor_telepon}`. It has a relationship `IS A` with `pasien`.
- pasien** (Entity): Attributes include `alamat`, `provinsi`, `kota`, and `jalan`. It has a relationship `dipasarkan` with `departemen`.
- departemen** (Entity): Attributes include `nama` and `alamat`. It has a relationship `dipasarkan` with `pasien`.
- janji\_temu** (Entity): Attributes include `waktu_pelaksanaan` and `nama`. It has a relationship `melakukan` with `tenaga_medis`.
- pesanan\_layanan** (Entity): Attributes include `nama` and `alamat`. It has a relationship `mencatat` with `departemen`.

The video call interface shows a presentation slide with a large 'S' and a video feed of the presenter, Albertus Christian Poandy. The status bar at the bottom indicates the time is 3:20 PM and the presentation is titled 'IF2240\_TB1\_M1\_BNI\_PresentasiERD'.