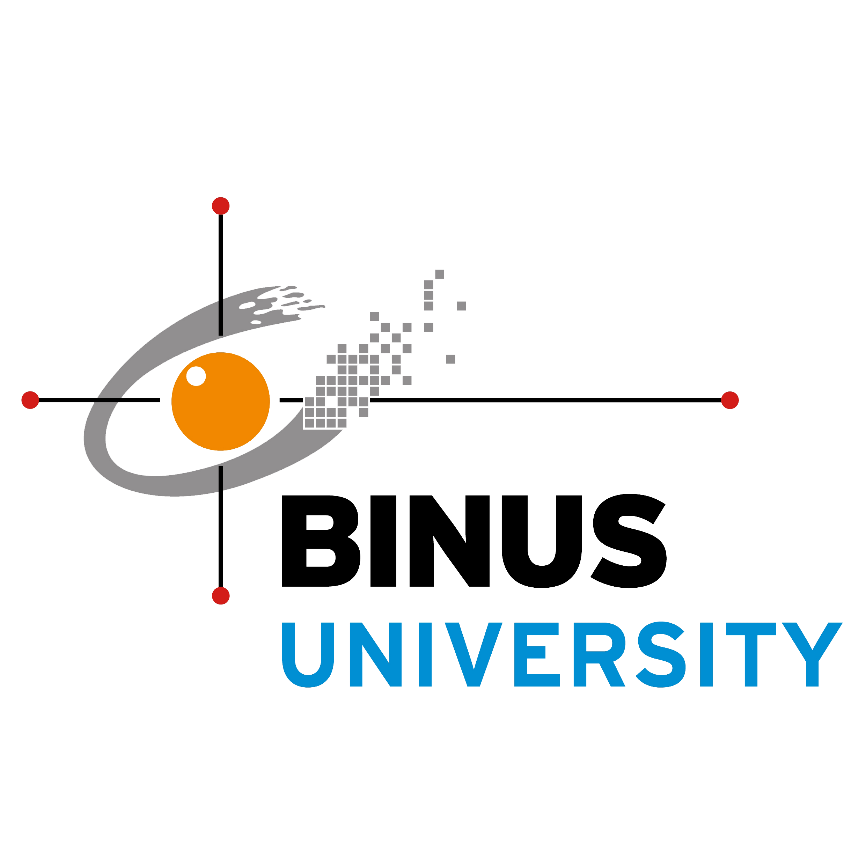
Assurance of Learning

Database Technology (COMP6893051)



Benedictus Cristiano Budi – 2602137496

Evan Pangguary - 2602123950

Muhammad Alib Dzakwan Fatih Ramadhan - 2602219975

Timotius Priestian Rajaska- 2602175462

School of Computer Science

Bina Nusantara University

2023

**CHAPTER I**

Kesehatan merupakan salah satu industri yang terus berkembang di Indonesia masa kini dikarenakan penduduk Indonesia yang berjumlah 278 juta jiwa terus mendorong industri tersebut untuk berkembang dan berinovasi. Setiap penduduk Indonesia berpotensi untuk terkena penyakit sehingga menyebabkan industri kesehatan menjadi industri yang akan selalu dibutuhkan. Oleh karena itu industri kesehatan dituntut untuk selalu siap melayani masyarakat di situasi manapun, seperti pada saat pandemi COVID-19 melanda Indonesia.

Masa pandemi merupakan masa-masa yang sulit bagi setiap warga Indonesia. Banyak warga yang jatuh sakit terkena COVID-19 dan tidak sedikit yang meninggal dunia karena virus tersebut. Warga berbondong-bondong mendatangi puskesmas atau rumah sakit untuk perawatan sampai sebagian besar rumah sakit di Indonesia penuh akan pasien COVID-19. Namun, hal-hal tersebut berhasil teratasi karena pemerintah Indonesia dan industri kesehatan langsung bereaksi untuk menanggulangi pandemi COVID-19. Pemerintah memerintahkan warga untuk menghentikan aktivitas serta melarang warga untuk keluar dari tempat tinggal mereka kecuali dalam situasi darurat sampai wabah COVID-19 mereda. Semenjak wabah COVID-19 melanda Indonesia, industri kesehatan sudah memulai mengembangkan vaksin untuk melawan virus COVID-19. Tidak hanya itu, mereka juga menyediakan masker bagi para warga serta berusaha untuk melayani warga yang terjangkit COVID-19 sebanyak mungkin. Walaupun Indonesia berhasil melewati pandemi COVID-19, banyak korban yang meninggal dunia di masa pandemi tersebut. Hal tersebut mendorong pemerintah dan industri kesehatan untuk berkembang dan berinovasi agar Indonesia siap untuk menghadapi pandemi jika suatu saat terjadi lagi.

**CHAPTER II**

Masalah yang teridentifikasi adalah management ruangan dan data pasien yang kurang efisien sehingga dapat menimbulkan berbagai macam masalah seperti salah pengobatan dan menempatkan pasien ke ruangan yang salah / sedang dipakai. Menggunakan DBMS sebagai media penyimpanan data menjadi solusi yang tepat untuk mengurangi anomali dalam mengoperasikan data dan redundansi data.

Dalam suatu rumah sakit pasti memiliki data dan salah satu hal yang dapat menjadi masalah jika tidak membuat database adalah volume data yang besar. Jika Rumah sakit menyimpan data pasien, riwayat medis pasien serta informasi kesehatan lain. Maka seiring berjalannya waktu rumah sakit akan menyimpan data yang besar. Agar tidak terjadi sesuatu yang tidak diinginkan seperti kerumitan saat mencari data pasien / data lain yang dibutuhkan dengan cepat, maka lebih baik dibuat database agar tidak mengalami hal tersebut.

Masalah lain yang memiliki kemungkinan untuk terjadi jika tidak membuat database adalah terjadi masalah / kerumitan dalam manajemen. contohnya masalah dalam pencatatan pasien baru, update tentang informasi pasien, manajemen penjadwalan dengan dokter, manajemen informasi tentang inventaris obat dan peralatan medis, manajemen faktur dan pembayaran. Agar hal ini tidak terjadi maka sebaiknya membuat database.

Dikarenakan data yang disimpan di rumah sakit adalah data penting pasien yang terdiri dari nama dan data lain pasien yang sangat sensitif. Maka database diperlukan agar dapat meminimalkan resiko terjadinya kebocoran data pasien, serta dapat mencegah serangan cyber agar data yang dimiliki pasien terjaga dengan baik serta dapat meningkatkan kepercayaan pasien terhadap rumah sakit.

**CHAPTER III**

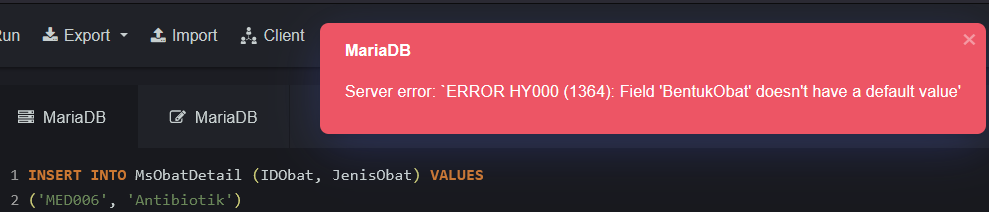
**Insert Anomaly**  
Terjadi dimana data gagal untuk dimasukkan ke dalam database karena input yang kurang lengkap.

Contoh:



Semua atribut wajib ada valuenya (not null), jika mencoba input seperti  
INSERT INTO TABLE MsObatDetail (ObatIDObat, JenisObat) VALUES (“OB001”, “Antibiotik”)

Operasi diatas tidak akan berhasil dijalankan dikarenakan BentukObat memilki constraint NOT NULL sementara operasi yang dilakukan hanya menginput value untuk ObatIDObat dan JenisObat, BentukObat dibiarkan kosong maka terjadi insert anomaly dan data tidak bisa dimasukkan ke dalam tabel.



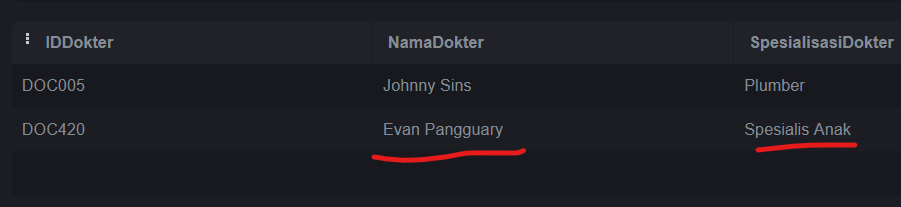
**Delete Anomaly**

Terjadi dimana data yang dihapus dapat menimbulkan redundansi data / data penting terhapus secara tidak sengaja.

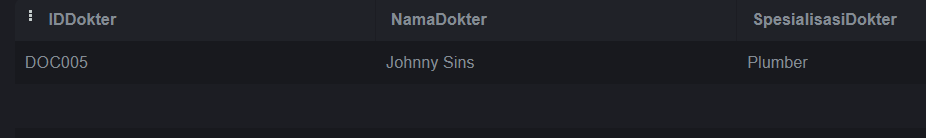
Contoh:



Ketika terjadi sebuah kasus dimana seorang dokter kehilangan “lisensi” spesialisasinya, maka ingin dihapus spesialisasinya dari database. Tapi apabila dokter tersebut hanya memiliki satu spesialisasi maka otomatis data dokter itu tidak akan menghilang dari database sehingga jika ingin mengakses informasi spesifik lainnya tentang dokter tersebut, tidak akan ada data yang bisa ditampilkan.

Sebelum Delete  


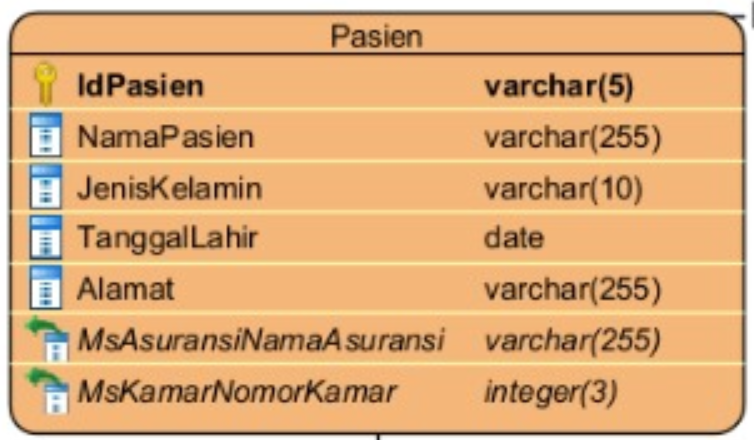
Setelah Delete (Hilang karena tidak ada tabel atau row lain yang menyimpan data Evan Pangguary)

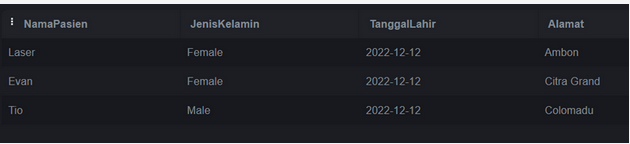


**Update Anomaly**

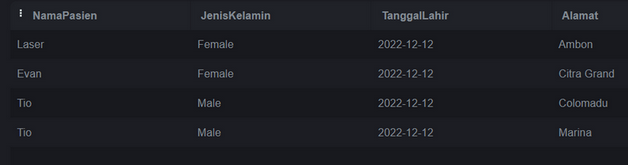
Terjadi dimana data diperbarui tetapi masih tersimpan versi lamanya sehingga mengakibatkan redudansi data.

Contoh:





Jika terjadi kasus dimana pasien mengubah alamatnya, maka akan dilakukan update. Update anomaly akan terjadi dimana data yang lama tidak dihapus sehingga 1 pasien memiliki 2 alamat yang sama, ini menyebabkan redundansi data.

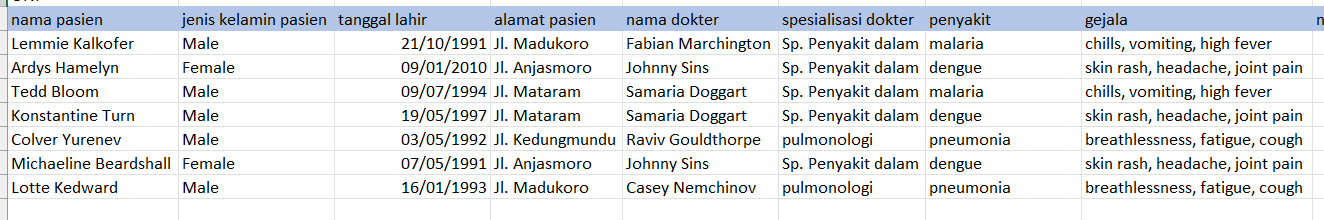


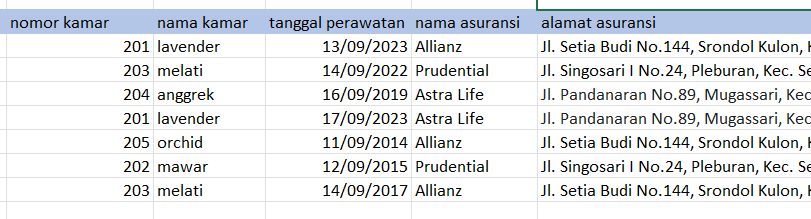
**CHAPTER IV**

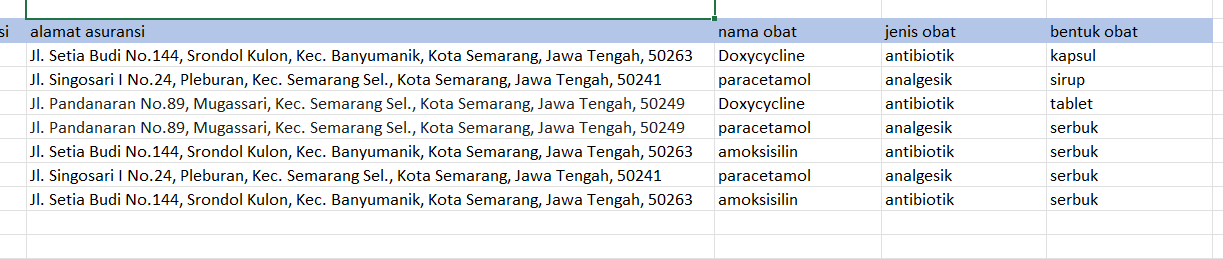
Normalisasi database adalah proses desain database untuk mengurangi redudansi data, meminimalkan terjadinya anomali insert delete atau update, dan meningkatkan integritas data. Dalam sektor database kesehatan terlebih pada rumah sakit, normalisasi menjadi penting untuk mengelola data pasien, dokter, perawatan, obat, dan lain lain dengan efisien.

Proses normalisasi memisahkan tabel menjadi entitas yang lebih kecil dan lebih terfokus, menghilangkan ketergantungan transitive, dan memastikan bahwa setiap atribut hanya bergantung pada Primary Key. Hasil dari hal tersebut adalah desain database yang lebih efisien, mudah dikelola untuk sektor kesehatan terutama rumah sakit, dan mencegah masalah anomali

**UNF**

****

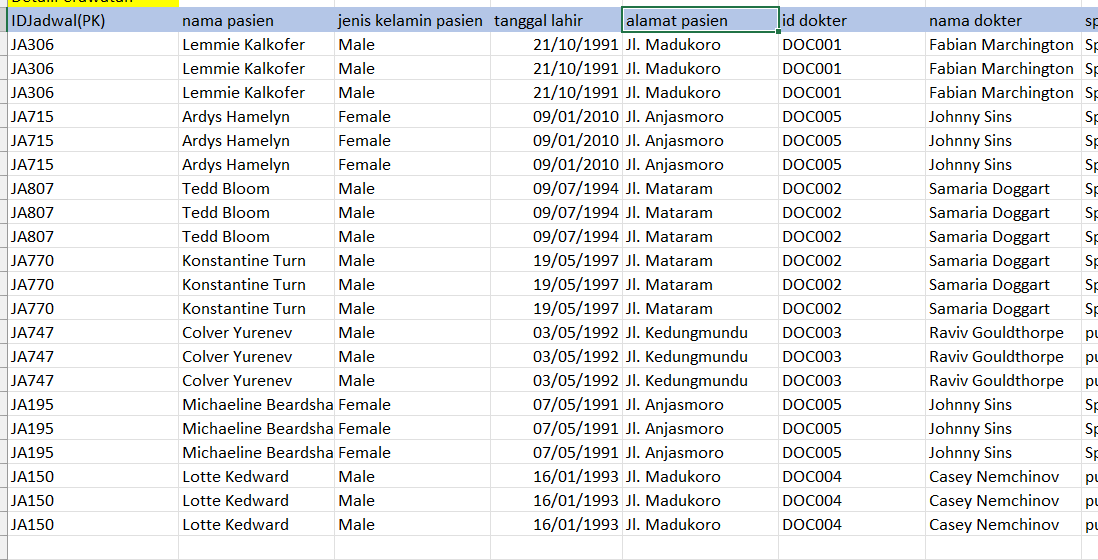
****

****

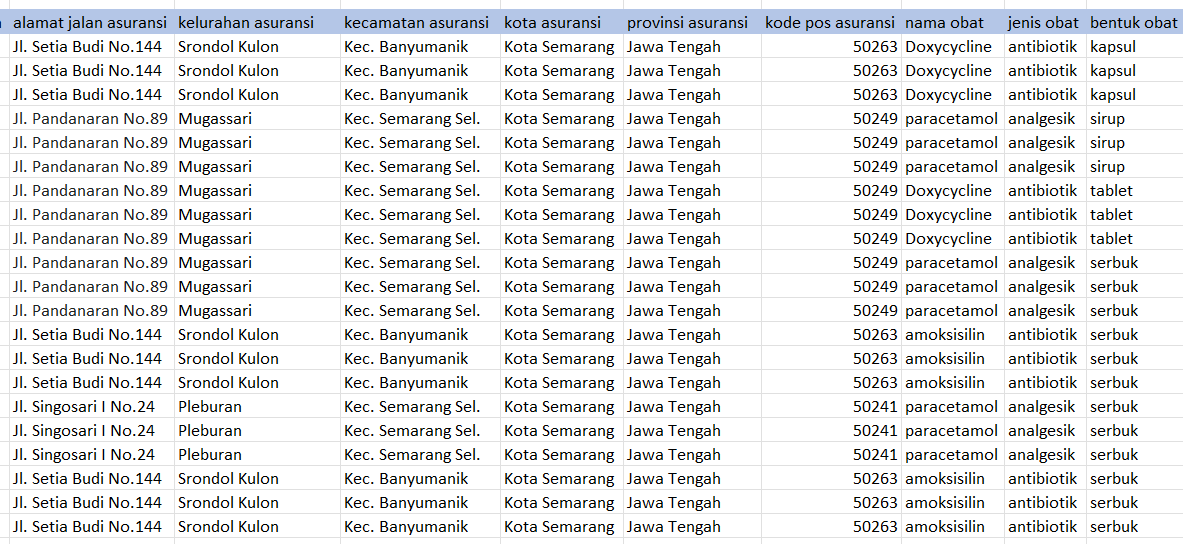
Tabel 4.1 UNF

Tabel di atas merupakan tabel yang berisi dataset sebuah rumah sakit dalam bentuk unnormal form. Dalam chapter ini, kami akan melakukan normalisasi database pada tabel di atas sampai bentuk normal ketiga atau third normal form (3NF).

**First Normal Form (1NF)**





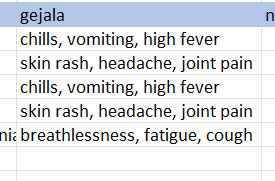


Tabel 4.2 DetailPerawatan

Normalisasi database 1NF adalah tingkatan pertama dalam normalisasi database. Tingkatan ini memiliki beberapa aturan yang bertujuan untuk meminimalisir duplikasi dan menyimpan data dalam bentuk yang paling sederhana.

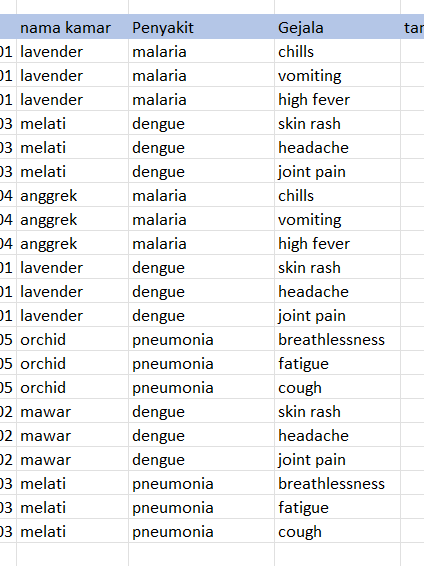
1. Menggunakan baris dalam bentuk terurut untuk menyampaikan informasi tidak diperbolehkan
2. Mencampur tipe data yang berbeda dalam satu kolom tidak diperbolehkan
3. Tabel tanpa memiliki primary key tidak diperbolehkan
4. Kelompok berulang tidak diperbolehkan
5. Setiap sel tidak dapat memiliki nilai lebih dari satu

Berdasarkan empat aturan di atas, membuat beberapa perbaikan pada tabel UNF untuk memenuhi aturan 1NF.

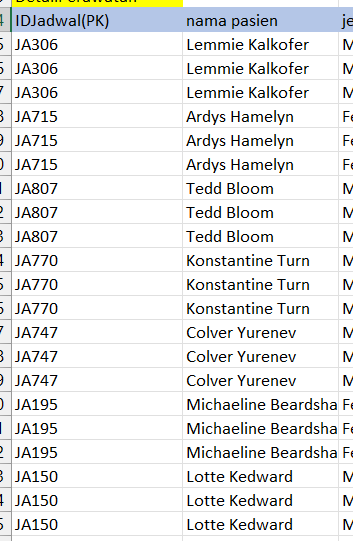


Gambar 4.1 Kolom Gejala

Pada kolom “gejala” pada tabel UNF, terdapat nilai yang melebihi satu. Hal tersebut melanggar aturan dari 1NF sehingga kami memisahkan nilai-nilai tersebut dan menempatkan mereka dalam bentuk baris. Kami juga menciptakan *Primary Key* untuk tabel *DetailPerawatan* untuk memenuhi aturan 1NF ketiga.



Gambar 4.2 Kolom Penyakit dan Gejala

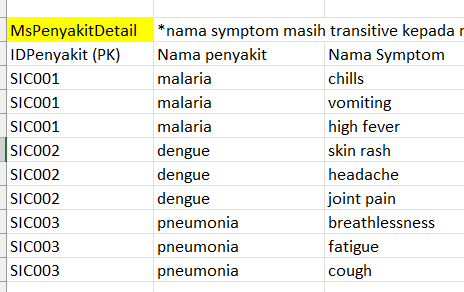


Gambar 4.3 Primary Key IDJadwal

**Second Normal Form (2NF)**



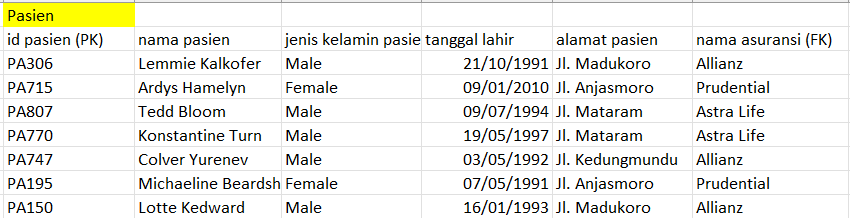
Tabel 4.3 DetailPerawatan



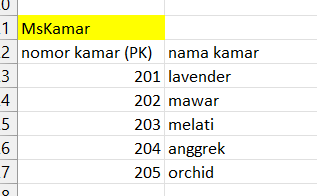
Tabel 4.4 MsPenyakitDetail



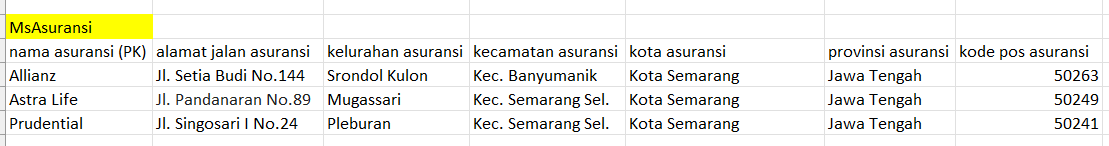
Tabel 4.5 MsDokter



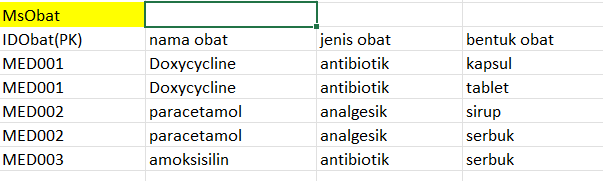
Tabel 4.6 Pasien



Tabel 4.7 MsKamar



Tabel 4.8 MsAsuransi



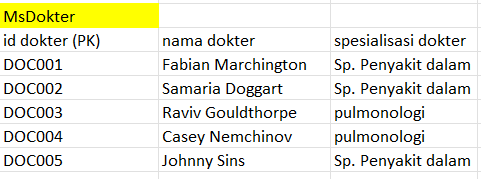
Tabel 4.9 MsObat

Normalisasi database 2NF adalah tingkatan kedua dalam normalisasi database yang menyatakan bahwa setiap kolom dalam tabel harus bergantung pada semua Primary Key. Tingkatan ini bertujuan untuk mengorganisir data dengan lebih baik dan mengurangi redudansi agar data dapat lebih mudah dikelola dan dijaga.

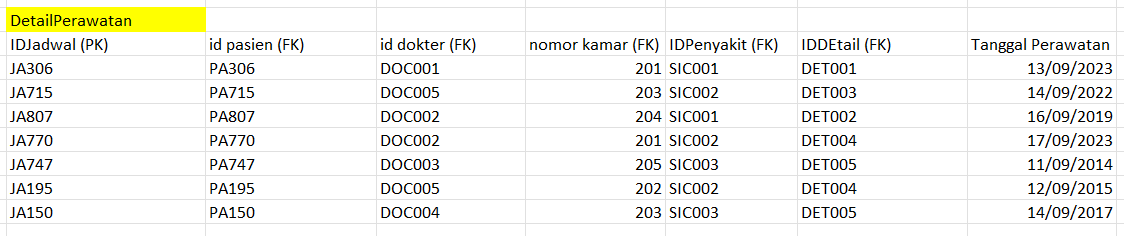
1. Harus memenuhi 1NF
2. Semua atribut non-key harus bergantung pada seluruh Primary Key

Berdasarkan dua aturan di atas, kami membagi tabel 1NF menjadi 7 tabel berbeda, yaitu tabel *DetailPerawatan*, *MsPenyakitDetail*, *MsDokter*, *Pasien*, *MsKamar*, *MsAsuransi* dan *MsObat*. Hal ini dilakukan untuk memenuhi aturan 2NF kedua dengan cara memisahkan atribut atau kolom yang bukan key serta menambahkan *Primary Key* pada tabel yang baru agar kolom-kolom tersebut dapat bergantung pada seluruh *Primary Key* dan juga menghilangkan ketergantungan parsial.

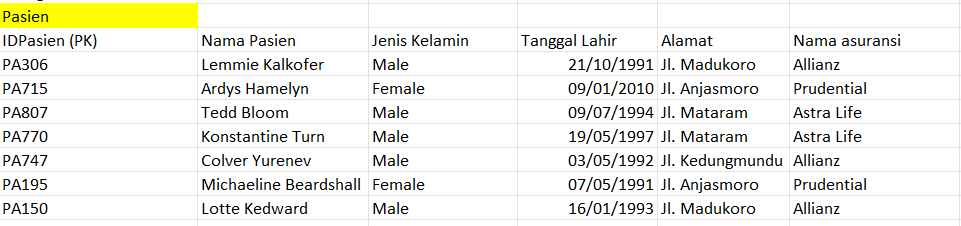
**Third Normal Form (3NF)**



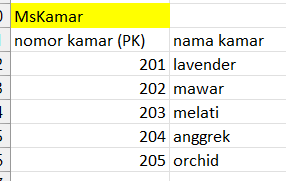
Tabel 4.10 MsDokter



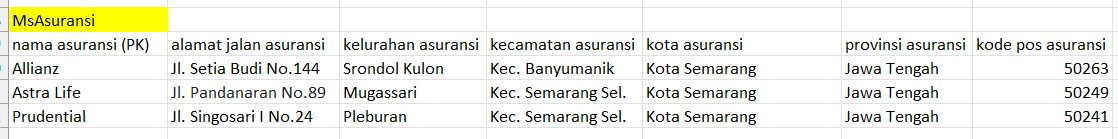
Tabel 4.11 DetailPerawatan



Tabel 4.12 Pasien

****

Tabel 4.13 MsKamar

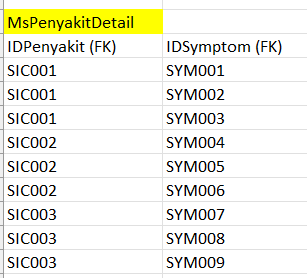


Tabel 4.14 MsAsuransi

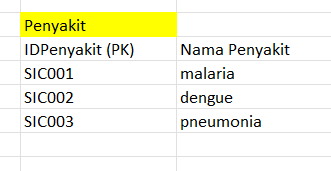
Normalisasi database 3NF adalah tingkatan lebih lanjut daripada 2NF dalam normalisasi database yang menyatakan bahwa Primary Key harus sepenuhnya mendefinisikan seluruh kolom Non-Key dan kolom Non-Key tidak boleh bergantung kepada Key yang lain. Tingkatan ini merupakan salah satu tahap normalisasi yang paling penting dalam desain database yang bertujuan untuk memastikan bahwa data memiliki struktur yang efisien dan tertata dengan baik, serta untuk menghindari masalah ketergantungan yang tidak diperlukan.

1. Harus memenuhi 2NF
2. Atribut non-key dalam tabel hanya dapat bergantung pada Primary Key dan tidak boleh bergantung pada atribut non-key lainnya

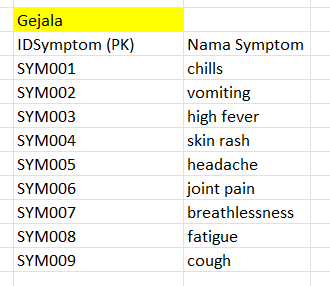
Berdasarkan dua aturan di atas, kami membagi tabel 2NF menjadi 10 tabel berbeda, yaitu tabel *DetailPerawatan*, *MsPenyakitDetail*, *Penyakit*, *Gejala*, *MsDokter*, *Pasien*, *MsKamar*, *MsAsuransi*, *MsObat* dan *DetailObat*. Kami menambahkan tabel *Penyakit*, *Gejala, MsObat* dan *DetailObat* untuk memenuhi aturan 3NF kedua dan menghilangkan ketergantungan transitive dengan cara memisahkan kolom “Nama Penyakit” dan “Nama Symptom” pada tabel *MsPenyakitDetail* menjadi dua tabel berbeda yaitu tabel *Penyakit* dan *Gejala* dengan *Primary Key* masing-masing dan memisahkan kolom “Jenis Obat” dan “Bentuk Obat” pada tabel *MsObat* dan menempatkan mereka pada tabel baru bernama *DetailObat* dengan *IDDetail* sebagai *Primary Key*. Pada tabel *MsPenyakitDetail*, kami menempatkan *Primary Key IDPenyakit* dan *Primary Key IDSymptom* sebagai *Composite Primary Key* agar kedua tabel tersebut dapat saling terhubung dan juga memiliki setiap baris memiliki *Primary Key* yang unik dan pada tabel *MsObat*, kami menempatkan *IDDetail* sebagai *Foreign Key* agar tabel *DetailObat* dapat terhubung ke *MsObat*.



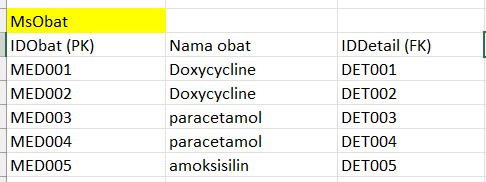
Tabel 4.15 MsPenyakitDetail



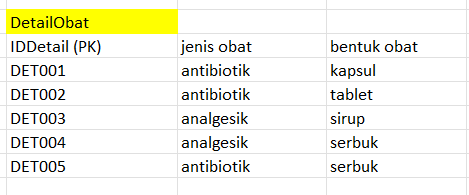
Tabel 4.16 Penyakit



Tabel 4.17 Gejala



Tabel 4.18 MsObat



Tabel 4.19 DetailObat

**CHAPTER V**

Industri kesehatan memegang peranan vital di Indonesia, khususnya dalam mengantisipasi situasi darurat. Salah satu aspek krusial dari persiapan ini adalah efisiensi dalam manajemen ruangan dan data pasien, dan untuk mencapai hal tersebut, penerapan teknologi database menjadi penting. Dalam konteks ini, normalisasi database menjadi langkah penting untuk meningkatkan efisiensi operasional dan mengatasi potensi masalah yang dapat muncul dalam pengelolaan data rumah sakit. Normalisasi memastikan struktur data yang terorganisir dengan baik, mengurangi risiko kesalahan, duplikasi, dan meningkatkan aksesibilitas data. Dengan demikian, penggunaan database yang terorganisir dan dinormalisasi tidak hanya menjadi prasyarat untuk efisiensi operasional, tetapi juga sebagai solusi untuk mengoptimalkan pengelolaan data kesehatan di lingkungan rumah sakit, memastikan respons yang cepat dan akurat dalam menghadapi tantangan kesehatan mendesak.

Berikut adalah link yang berisi query untuk database kami:

<https://docs.google.com/document/d/1wIVHZf-Ql-PGmxLuS279GKwhOsUU8ukTGA_jNLuJ0BQ/edit?usp=sharing>

Berikut adalah gambar ERD untuk database kami:

