1. MODUL I

**PERANCANGAN DIAGRAM DAN NORMALISASI**

* 1. **TUJUAN**
  2. Mahasiswa memahami permasalahan dan dapat menentukan entitas-entitas dari permasalahan tersebut.
  3. Mahasiswa dapat menentukan atribut-atribut dari suatu entitas.
  4. Mahasiswa dapat menentukan relasi antara masing-masing entitas
  5. Mahasiswa memahami dan dapat merancang suatu permasalahan yang diberikan berdasarkan teori yang telah dipelajari.
  6. Mahasiswa dapat melakukan proses normalisasi.
  7. **LANDASAN TEORI**
     1. **Elemen-elemen Diagram ER**

*Entity* *Relationship* *Diagram* (ERD) merupakan model data berupa notasi grafis dalam pemodelan data konseptual yang menggambarkan hubungan antara penyimpan. Adapun elemen-elemen diagram ER antara lain:

* + - 1. **Entitas**

Entitas adalah sesuatu atau objek di dunia nyata yang dapat dibedakan dari sesuatu atau objek lain [1]. Entitas menunjukkan objek-objek dasar yang terkait di dalam sistem yang dapat berupa objek fisik seperti rumah dan manusia ataupun objek konsep seperti pekerjaan dan perusahaan. Entitas biasanya diberi nama dengan kata benda. Terdapat dua jenis entitas, yaitu:

**Entitas Kuat**

Adalah entitas yang berdiri sendiri, keberadaannya tidak tergantung dengan entitas lain:

|  |
| --- |
|  |

**Gambar 1.1** Notasi entitas kuat

**Entitas Lemah**

Entitas yang keberadaannya bergantung pada entitas lain, artinya entitas tersebut tidak berguna jika tidak ada entitas yang lain. Berikut adalah notasinya:

|  |
| --- |
|  |

**Gambar 1.2** Notasi entitas lemah

* + - 1. **Atribut**

Atribut merupakan elemen dari sebuah entitas dan berfungsi mendeskripsikan karakter entitas. Atribut juga dikatakan sebagai properti deskriptif yang dimiliki oleh setiap anggota dari himpunan entitas [2]. Atribut berfungsi sebagai entitas penjelas. Simbol yang menyatakan atribut adalah simbol *elips*, dimana nama dari atribut dituliskan di dalam simbol. Nama dari atribut biasanya berupa kata tunggal.

Terdapat beberapa atribut yang digunakan untuk menggambarkan model ER diagram, yaitu:

1. **Atribut kunci**

Atribut yang digunakan untuk menentukan suatu entitas secara unik [2]. Contohnya id\_barang.

|  |
| --- |
|  |

**Gambar 1.3** Notasi atribut kunci

1. **Atribut *simple***

Atribut yang bernilai tunggal. Contohnya yaitu nama, harga dan status [2].

|  |
| --- |
|  |

**Gambar 1.4** Notasi atribut *simple*

1. **Atribut *multivalue***

Suatu atribut yang memiliki sekelompok nilai yang banyak [2]. Contohnya gelar dan hobi.

|  |
| --- |
|  |

**Gambar 1.5** Notasi atribut *multivalue*

1. **Atribut *composite***

Atribut yang masih dapat diuraikan lagi [2]. Contohnya alamat dan waktu.

|  |
| --- |
|  |

**Gambar 1.6** Notasi atribut *composite*

1. **Atribut derivatif**

Atribut yang dihasilkan dari turunan atribut yang lain [2]. Contohnya umur dari tanggal lahir.

|  |
| --- |
|  |

**Gambar 1.7** Simbol atribut derivatif

* + - 1. **Relasi**

Relasi adalah hubungan antara satu himpunan entitas dengan himpunan entitas yang lain [2].Pada diagram ER, relasi digambarkan dengan sebuah bentuk belah ketupat. Pada umumnya, relasi diberi nama dengan kata kerja sehingga memudahkan untuk melakukan pembacaan relasi. Berikut adalah notasi dari relasi:

|  |
| --- |
|  |

**Gambar 1.8** Simbol relasi

Kerelasian antar entitas dapat dikelompokkan dalam 3 jeNIS yang menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas lain yang biasa disebut dengan kardinalitas [3]. Tiga macam kardinalitas tersebut ialah:

1. **Relasi satu ke satu (*One to One*)**

Setiap entitas pada himpunan entitas A berhubungan dengan paling banyak satu entitas pada himpunan entitas B, dan begitu juga sebaliknya setiap entitas B pada himpunan entitas B berhubungan paling banyak satu entitas pada himpunan entitas A [2]. Simbol relasi satu ke satu yaitu:

|  |
| --- |
|  |

**Gambar 1.9** Notasi relasi satu ke satu

1. **Relasi Satu ke Banyak (*One to Many*)**

Setiap entitas pada himpunan entitas A berhubungan dengan banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas B berhubungan dengan paling banyak satu entitas pada himpunan entitas A [2]. Simbol relasi satu ke banyak yaitu:

|  |
| --- |
|  |

**Gambar 1.10** Notasi relasi satu ke banyak

1. **Relasi Banyak ke Satu (*Many to One*)**

Setiap entitas pada himpunan entitas A berhubungan dengan paling banyak satu entitas pada himpunan entitas B, tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas B berhubungan dengan banyak entitas pada himpunan entitas A [2]. Simbol relasi banyak ke satu yaitu:

|  |
| --- |
|  |

**Gambar 1.11** Notasi relasi banyak ke satu

1. **Relasi banyak ke banyak (*Many to Many*)**

Setiap entitas pada himpunan entitas A berhubungan dengan banyak entitas pada himpunan entitas B, dan begitu juga sebaliknya [2]. Simbol relasi banyak ke banyak yaitu:

|  |
| --- |
|  |

**Gambar 1.12** Notasi relasi banyak ke banyak

* + 1. **Proses Perancangan Diagram ER**

Langkah-langkah yang dapat dilakukan untuk menghasilkan diagram ER adalah sebagai berikut [2]:

1. Mengidentifikasi dan menetapkan seluruh himpunan entitas yang akan terlibat.
2. Menentukan atribut-atribut *key* dari masing-masing himpunan entitas.
3. Mengidentifikasi dan menetapkan seluruh himpunan relasi di antara himpunan entitas-himpunan entitas yang ada beserta *foreign-key*-nya.
4. Menentukan derajat/kardinalitas relasi untuk setiap himpunan relasi.
5. Melengkapi himpunan entitas dan himpunan relasi dengan atribut-atribut deskriptif (*non key*).
   * + - 1. **Normalisasi**

Normalisasi merupakan sebuah teknik dalam desain logika sebuah basis data yang mengelompokkan atribut dari suatu tabel/relasi sehingga membentuk struktur relasi yang baik (tanpa redundansi) [3].

Tujuan dilakukannya normalisasi adalah sebagai berikut:

1. Minimalisasi redundansi (pengulangan data).
2. Memudahkan identifikasi entitas.
3. Mempermudah pemodifikasian data.
4. Mencegah terjadinya anomali.

Tahapan normalisasi dimulai dari tahap paling ringan (1NF) hingga paling ketat (5NF). Urutan tahapnya yaitu 1NF, 2NF, 3NF, BCNF, 4NF, 5NF [1]. Biasanya hanya sampai pada tingkat 3NF, suatu tabel sudah dapat menghasilkan tabel yang berkualitas baik.

Berikut tahapan normalisasi dalam perancangan basis data:

1. **Bentuk Normal Pertama (1NF)**

Suatu tabel disebut memenuhi bentuk normal pertama jika:

1. Tidak adanya atribut *multivalue*, atribut *composite* atau kombinasinya.
2. Setiap atribut dalam tabel tersebut harus bernilai *atomic* (tidak dapat dibagi-bagi lagi).
3. Tidak ada perulangan grup atribut dalam satu baris.
4. **Bentuk Normal Kedua (2NF)**

Suatu tabel disebut memenuhi bentuk normal kedua jika:

1. Sudah memenuhi dalam bentuk normal pertama (1NF).
2. Semua atribut bukan kunci utama bergantung secara fungsional terhadap semua atribut kunci dan bukan hanya sebagian atribut kunci.
3. Jika ada ketergantungan parsial maka atribut tersebut harus dipisah pada tabel yang lain, dimana perlu ada tabel penghubung ataupun kehadiran *foreign-key* bagi atribut-atribut yang telah terpisah.
4. **Bentuk Normal Ketiga (3NF)**

Suatu tabel disebut memenuhi bentuk normal ketiga jika:

1. Sudah memenuhi bentuk normal kedua (2NF).
2. Setiap atribut yang bukan kunci tidak tergantung secara fungsional terhadap atribut bukan kunci yang lain dalam relasi tersebut (tidak ada ketergantungan transitif pada atribut bukan kunci).
3. **Keuntungan normalisasi dari permasalahan yang ada di dunia nyata**

Dari permasalahan yang ada di dunia nyata terdapat keuntungan normalisasi di antaranya sebagai berikut:

1. Meminimalkan ukuran penyimpanan yang diperlukan untuk penyimpanan data.
2. Meminimalkan resiko kandungan inkonsistensi data pada basis data.
3. Meminimalkan kemungkinan anomali pembaruan.
4. Memaksimalkan stabilitas struktur data.
5. Kinerja penelusuran yang lebih cepat.
6. Meningkatkan integritas data.

# KASUS KELOMPOK

Sekolah Dasar Internasional Wakanda (SBIW) akan merancang sebuah sistem informasi. Basis data yang dibentuk berdasarkan data-data berbagai entitas. Guru akan didata nama, kontak dan alamat sehingga terbentuk NUPTK. Lalu guru dibedaan menjadi guru honorer dan PNS yang memiliki NIP. Semua guru dapat mengajar mata pelajaran, dan guru pun menguasai tidak hanya satu saja mata pelajaran, namun terkadang mereka harus mengajar mata pelajaran yang tidak mereka pelajari, sehingga mata pelajaran dapat dikuasai dan diajar oleh banyak guru. Mata pelajaran hanya didata namanya saja, sehingga dibentuk kode mata pelajaran. Setiap mata pelajaran yang diajar guru akan didata pegisian kelas mana yang diajar. Sehingga guru dapat mengajar mata pelajaran yang sama di kelas yang berbeda, begitupun dikelas yang sama dapat mengajar mata pelajaran lain. Kelaspun harus di data nama kelasnya sehinggaa terbentuk kode kelas. Sebuah kelas dapat diwalikan oleh seorang guru non-honorer. Sebuah kelaspun memiliki banyak siswa, dimana dari mereka terdapat satu yang menjadi ketua kelas. Setiap siswa akan didata nama, alamay, nama wali, kontak wali, dan rangking mereka hingga dibentuk NIS. Setiap siswa dapat mempelajari banyak mata pelajaran, begitu pula sebaliknya setiap mata pelajaran dapat dipelajari banyak siswa. Setiap kegiatan belajar mengajar akan dilakukan pengambilan nilai.

# KASUS MODUL

* + - * 1. Melakukan perancangan ERD serta menentukan kamus data yang dimiliki diagram tersebut.
        2. Melakukan normalisasi 1NF, 2NF, dan 3NF terhadap ERD yang telah dirancang

# ANALISIS KASUS

* + 1. **Perancangan ERD serta penentuan kamus data dari diagram tersebut.**
       - 1. **Rancangan ER Diagram Sebelum Normalisasi**

|  |
| --- |
| X:\Kampus\Kuliah Angga\Semester 3\Sistem Basis Data\WhatsApp Image 2019-11-01 at 22.16.01.jpeg |

**Gambar 1.13** ERDSD Int­ernasional Wakanda

* + - * 1. **Kamus Data Sebelum Normalisasi**

guru : {NUPTK, nama\_guru, kontak\_guru, alamat\_guru}

PNS : {NIP}

mata\_pelajaran: {kode\_mapel, nama\_mapel}

kelas : {kode\_kelas, nama\_kelas}

siswa : {NIS, nama\_siswa, rank, alamat\_siswa, nama\_wali, kontak\_wali}

mempelajari : {nilai}

* + 1. **Tabel *Universal***

**Tabel 1.1** Tabel *universal*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| NUPTK | nama\_guru | NIP | kontak\_guru | alamat\_guru | kode\_jadwal | ... |
|  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ... | kode\_mapel | nama\_mapel | nilai | kode\_kelas | nama\_kelas | ... |
|  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ... | NIS | nama\_siswa | rank | alamat | nama\_wali | kontak\_wali |
|  |  |  |  |  |  |  |

Pada **Tabel 1.1** merupakan tabel *universal*. Tabel *universal* (*Universal* / *Star Table*) merupakan sebuah tabel yang merangkum semua kelompok data yang saling berhubungan, dan tabel ini bukan merupakan tabel yang baik. Tabel ini diperoleh dari rancangan ERD yang belum mengalami proses normalisasi yang dimana pada tabel ini diperoleh atribut-atribut yang berasal dari entitas “Guru”, “PNS”, “Mata\_Pelajaran”, “Honorer”, “Siswa”, dan “Kelas” serta relasi “Mempelajari”, “Menguasai”, “Mengajar”, dan “Mempelajari”. Proses normalisasi baru dapat dimulai setelah tabel *universal* terbentuk.

* + 1. **Normalisasi 1 *Normal Form*, 2 *Normal Form* dan 3 *Normal Form***

Pada Tahap ini, akan dilakukan normalisasi pertama yaitu 1NF, kemudian setelah mendapatkan hasil normalisasi 1NF, maka akan dilanjutkan proses normalisasi 2NF, apabila kondisi masih memungkinkan untuk dilakukan proses normalisasi 3NF, maka akan dilakukan proses normalisasi tabel dalam bentuk 3NF.

* + - * 1. **Tabel 1 *Normal Form***

Bentuk normalisasi yang pertama adalah 1NF. Di dalam tabel bentuk normal pertama ini tidak diperbolehkan adanya atribut yang tidak bernilai tunggal, yaitu atribut komposit dan *multivalue.* Tabel 1NF yang dibuat harus bernilai tunggal (atomik).Di bawah ini adalah tabel 1NF Sistem Informasi “Sistem Informasi SDI Wakanda”:

**Tabel 1.2** Tabel 1 *Normal Form*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| NUPTK | nama\_guru | NIP | kontak\_guru | alamat\_guru | ... |
|  |  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ... | kode\_mapel | nama\_mapel | nilai | kode\_kelas | nama\_kelas | ... |
|  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ... | NIS | nama\_siswa | rangking | alamat | nama\_wali | kontak\_wali |
|  |  |  |  |  |  |  |

Pada **Tabel 1.2**, sudah dapat dikatakan berbentuk 1NF dikarenakan pada tabel tersebut tidak ada kolom yang memiliki atribut bernilai banyak (*multivalued attribute*) atau lebih dari satu atribut dengan domain nilai yang sama.

* + - * 1. **Tabel 2 *Normal Form***

Bentuk normal tahap kedua (2NF) terpenuhi jika pada sebuah tabel, semua atribut yang tidak termasuk dalam *key primer* memiliki ketergantungan fungsional (KF) pada *key primer* secara utuh. Sebuah tabel dikatakan tidak memenuhi 2NF, jika ketergantungannya hanya bersifat parsial (hanya tergantung pada sebagian dari *key primer*).Untuk mengatasinya, tabel 1NF perlu dipisah menjadi bagian-bagian sebagai berikut:

Karena ada dua jenis guru didalam ERD, yakni guru honorer dan PNS maka tabel guru dipisah menjadi dua bagian, yakni:

**Tabel 1.3** Tabel guru 2 *Normal Form*

|  |  |  |  |
| --- | --- | --- | --- |
| NUPTK | nama\_guru | kontak\_guru | alamat\_guru |
|  |  |  |  |

**Tabel 1.4** Tabel guru 2 *Normal Form*

|  |  |
| --- | --- |
| NUPTK | NIP |
|  |  |

**Tabel 1.3** dan **Tabel 1.4** merupakan Tabel guru yang terdiri dari atribut pada entitas guru PNS yaitu NUPTK sebagai *primary key*. Tabel tersebut sudah berbentuk 2 NF yaitu data dikelompokkan berdasarkan entitas, dan Tabel tersebut sudah memenuhi 1 NF dan tidak terdapat ketergantungan sebagian.

**Tabel 1.5** Tabel mata\_pelajaran 2 *Normal Form*

|  |  |
| --- | --- |
| kode\_mapel | nama\_mapel |
|  |  |

**Tabel 1.5** merupakan Tabel mata\_pelajaran yang terdiri dari atribut pada entitas mata\_pelajaran yaitu kode\_mapel sebagai *primary key*, yang dimana menentukan atribut bukan kunci yaitu nama\_mapel. Tabel tersebut sudah berbentuk 2 NF yaitu data dikelompokkan berdasarkan entitas, dan Tabel tersebut sudah memenuhi 1 NF dan tidak terdapat ketergantungan sebagian.

**Tabel 1.6** Tabel kelas 2 *Normal Form*

|  |  |  |  |
| --- | --- | --- | --- |
| kode\_kelas | nama\_kelas | NIM\_ketua | NUPTK\_walikelas |
|  |  |  |  |

**Tabel 1.6** merupakan Tabel kelas yang terdiri dari atribut pada entitas kelas yaitu kode\_kelas sebagai *primary key*, yang menentukan nama\_kelas, NIM\_ketua, dan NUPTK\_walikelas. Tabel tersebut sudah berbentuk 2 NF yaitu data dikelompokkan berdasarkan entitas, dan Tabel tersebut sudah memenuhi 1 NF dan tidak terdapat ketergantungan sebagian.

**Tabel 1.7** Tabel siswa 2 *Normal Form*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| NIS | nama\_siswa | rank | Alamat\_siswa | nama\_wali | kontak\_wali | kode\_kelas |
|  |  |  |  |  |  |  |

**Tabel 1.7** merupakan Tabel siswa yang terdiri dari atribut pada entitas siswa yaitu NIS yang berlaku sebagai *primary key* yang kemudian menentukan nama\_siswa, rank, alamat\_siswa, nama\_wali, kontak\_wali, dan kode\_kelas. Tabel tersebut sudah berbentuk 2 NF yaitu data dikelompokkan berdasarkan entitas, dan Tabel tersebut sudah memenuhi 1 NF dan tidak terdapat ketergantungan sebagian. NIS bertindak sebagai *primary key* pada tabel ini.

* + - * 1. **Tabel 3 *NForm***

Pada tahapan ketiga yaitu 3NF, tabel sudah dapat dikatakan tabel yang baik. Oleh karena itu, ini adalah tahapan normalisasi yang terakhir. Syarat bagi suatu tabel untuk memenuhi 3NF adalah memenuhi 2NF dan tidak ada ketergantungan transitif, yang berarti ketergantungan atribut bukan kunci pada atribut bukan kunci lainnya. Dari tabel-tabel 2NF sistem informasi “Sistem Informasi SDI Wakanada”, yang perlu perbaiki adalah tabel relasi.

**Tabel 1.8** Tabel mengajar.

|  |  |
| --- | --- |
| NUPTK | Kode\_mapel |
|  |  |

**Tabel 1.8** merupakan Tabel mengajar yang terdiri dari atribut pada relasi mengajar, tabel ini dikatakan berbentuk 3NF dikarenakan tidak ada ketergantungan transitif pada setiap kandidat *key*, atribut NUPTK yang merupakan atribut bukan kunci pada relasi juga menentukan atribut kode\_mapel.

**Tabel 1.9** Tabel menguasai.

|  |  |
| --- | --- |
| NUPTK | kode\_mapel |
|  |  |

**Tabel 1.9** merupakan pengaplikasian relasi yang bertujuan untuk menghubungkan antara entitas guru dan entitas kelas yang diajar oleh guru yang bersangkutan. Tabel ini sudah dapat dikatakan tidak memiliki ketergantungan transitif pada setiap kandidat *key*, dikarenakan atribut NUPTK menentukan atribut kode\_mapel yang dimana kedua atribut tersebut merupakan atribut bukan kunci.

**Tabel 1.10** Tabel menghadiri

|  |  |  |
| --- | --- | --- |
| NUPTK | kode\_mapel | kode\_kelas |
|  |  |  |

**Tabel 1.10** merupakan pengaplikasian relasi yang bertujuan untuk menghubungkan guru, mata\_pelajaran, dan kelas. Hal ini bertujuan agar guru mengetahui mata pelajaran apa yang akan diajarkan pada kelas yang akan dihadiri olehnya, atribut NUPTK menentukan atribut kode\_mapel dan kode\_kelas yang dimana merupakan atribut yang sama-sama bukan atribut kunci.

**Tabel 1.11** Tabel mempelajari

|  |  |  |
| --- | --- | --- |
| NIS | Kode\_mapel | nilai |
|  |  |  |

**Tabel 1.11** merupakan pengaplikasian relasi yang bertujuan untuk menghubungkan antara entitas siswa dan entitas mapel yang dipelajari oleh para siswa yang bersangkutan. Pada tabel ini atribut NIS dan kode\_mapel menentukan atribut nilai, yang dimana pada tabel ini sudah bersifat 3 NF.

* + - * 1. **ERD setelah dinormalisasi**

|  |
| --- |
|  |

**Gambar 1.14** ERDSD Int­ernasional Wakanda setelah normalisasi

* + - * 1. **Kamus Data Setelah Normalisasi**

guru : {NUPTK, nama\_guru, kontak\_guru, alamat\_guru}

PNS : {NUPTK, NIP}

honorer : {NUPTK}

mata\_pelajaran :{kode\_mapel, nama\_mapel}

kelas : {kode\_kelas, nama\_kelas, NIS\_ketua, NUPTK\_walikelas}

siswa : {NIS, nama\_siswa, rank, alamat\_siswa, nama\_wali, kontak\_wali, kode\_kelas}

mengajar : {NUPTK, kode\_mapel}

menguasai : {NUPTK, kode\_mapel}

menghadiri : {NUPTK, kode\_jadwal, kode\_kelas}

mempelajari : {NIS, kode\_mapel, nilai}

# KESIMPULAN

Berdasarkan praktikum yang telah dilaksanakan, dapat diambil beberapa kesimpulan, antara lain:

* + - 1. Dalam membuat sebuah ERD, entitas dapat ditentukan dari sebuah objek yang melakukan suatu kegiatan dilingkungan pemakai dan tersimpan di dalam sistem. Entitas adalah obyek yang mewakili sesuatu dalam dunia nyata dan dapat dibedakan antara satu dengan lainnya (*unique*). Setiap entitas memiliki beberapa atribut yang mendeskripsikan karakteristik dari objek tersebut.
      2. Atribut adalah merupakan sifat-sifat atau karakteristik pada suatu entitas. Dalam sebuah ERD terdapat 2 jenis atribut, yaitu *identifier* (*key*) dan *descriptor* (*non-key*). Atribut *identifier* digunakan untuk mendeskripsikan entitas secara unik, sedangkan *descriptor* digunakan untuk menspesifikasikan entitas (tidak secara unik).
      3. Relasi merupakan hubungan yang terjadi pada suatu tabel dengan lainnya yang mempresentasikan hubungan antar objek di dunia nyata dan berfungsi untuk mengatur operasi suatu database. Relasi biasanya dikaitkan dengan hubungan antar entitas dan biasanya menggunakan kata kerja sebagai penamaannya. Dalam pembuatan sebuah relasi, kita harus mengetahui kelakuan sistem atau bagaimana sistem bekerja terlebih dahulu. Sehingga kemudian kita dapat mengetahui entitas mana saja yang melakukan suatu relasi / hubungan.
      4. Merancang ERD dapat dilakukan dengan menganalisa suatu permasalahan, menentukan entitas dan atributnya lalu menentukan relasi dan derajat kardinalitasnya dengan tepat, dan meninjau kembali ERD apakah memerlukan penghalusan dengan batasan relasi.
      5. Dalam pembentukan basis data yang baik, perlu dilakukan normalisasi pada basis data yang dibuat. Proses normalisasi dilakukan dengan tujuan salah satunya untuk meminimalisir redudansi data saat dilakukan perubahan (*update*). Adapun tingkatan dari normalisasi basis data yaitu 1NF, 2NF, 3NF, BCNF, 4NF dan 5NF. Namun dalam pelaksanaannya, normalisasi sampai pada 3NF sudah dikatakan baik dan meminimalisir kemungkinan terjadinya redudansi.