BAB III

OBJEK DAN METODE PENELITIAN

3.1 Objek Penelitian

Objek dari penelitian yang akan dilakukan adalah Hotel Setiabudhi Indah, yang terletak di Jalan Dr. Setiabudhi No. 266 Bandung.

3.1.1 Sejarah Perusahaan

Hotel Setiabudhi Indah dibangun sekitar bulan November 1989 diatas sebidang tanah yang sebelumnya sudah terdapat sebuah bangunan peninggalan zaman Belanda. Ketika mengurus izin bangunan, terdapat syarat bahwa jika ingin membangun, tampak muka rumah tidak boleh diubah. Dan syarat tersebut disetujui oleh bapak Barli Sasmita sebagai pemilik yang baru.

Fisik bangunan selesai pada Bulan Januari 1991 yang terdiri dari 3 bagian, dengan 40 kamar, sebuah restoran, dan sebuah ruangan pertemuan (*meeting room*).

Bagian depan hotel hanya terdiri dari 1 lantai dimana terdapat muka rumah yang dalam syarat tidak boleh diubah. Selain itu juga terdapat sebuah restoran, *lobby, front office, drug store* dan toko *souvenir* yang menjual kerajinan yang terbuat dari keramik, dan lain-lain.

Bagian tengah terdiri dari 3 lantai dimana terdapat kamar sejumlah 40 buah yang terbagi atas beberapa tipe, yaitu *superior, deluxe, standard* dan ekonomi. Selain itu juga, dibagian tengah ini terdapat *room boy station*.

Bagian belakang terdiri dari 4 lantai dimana terdapat *basement, laundry, driver room*, dan ruang pertemuan (*meeting room*).

Pada saat fisik banngunan sudah selesai, daya listrik baru mencapai 1200 watt dan saat itu belum mendapatkan penambahan dari PLN. Hingga pada awal bulan September 1993 permohonan penambahan daya listrik baru diberikan oleh PLN. Setelah itu, dilakukan perkrutan karyawan. 2 minggu setelah perekrutan karyawan, dan dilakukan *finishing*, maka Hotel Setiabudhi Indah pun resmi dibuka pada tanggal 25 September 1993.

3.1.2 Visi dan Misi Perusahaan

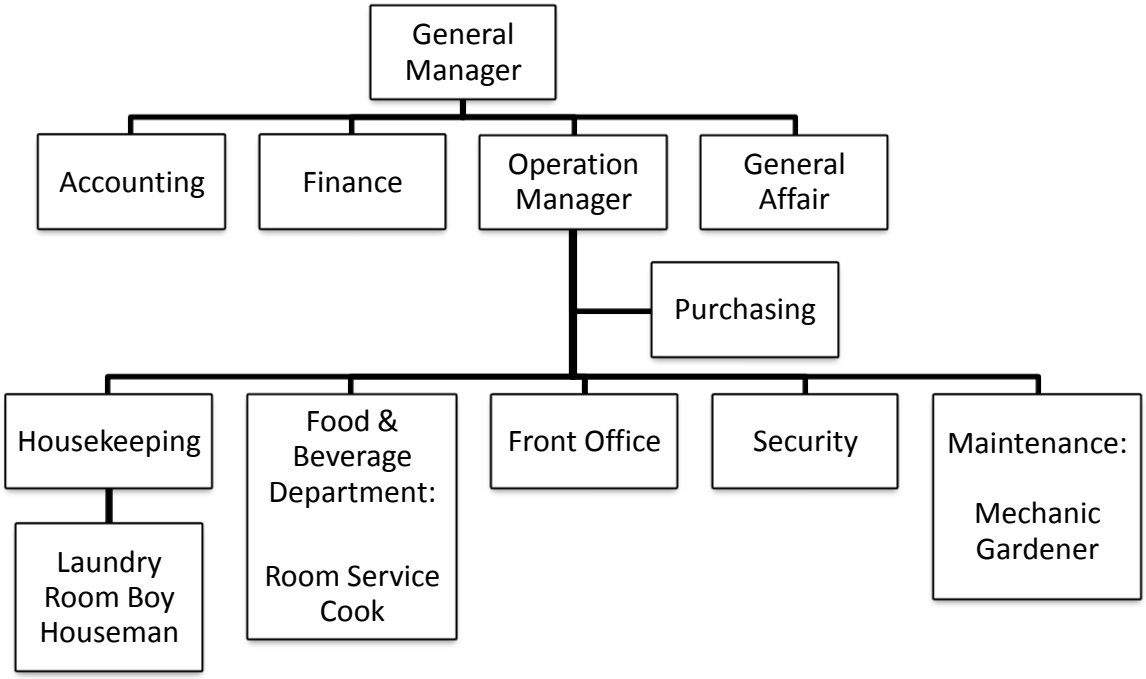
Visi dari Hotel Setiabudhi Indah adalah memberikan pelayan yang terbaik untuk memberikan kepuasan bagi para pelanggan atau tamu hotel.

Misi dari Hotel Setiabudhi Indah adalah berpartisipasi dalam perkembangan pariwisata di Bandung, serta dapat mengembangkan bisnis perhotelan yang lebih luas.

3.1.3 Struktur Organisasi Perusahaan

Struktur organisasi adalah penggambaran secara grafik yang menggambarkan struktur kerja dari suatu struktur organisasi.

Adapun struktur organisasi yang terdapat pada Hotel Setiabudhi Indah dapat dilihat pada gambar dibawah ini:



Gambar 3.1 Struktur Organisasi Hotel Setiabudhi Indah

(Sumber : Hotel Setiabudhi Indah)

3.1.4 Deskripsi Kerja

Deskripsi tugas merupakan suatu rincian yang menunjukan posisi, tanggung jawab, wewenang, fungsi dan tugas-tugas yang harus dikerjakan oleh seorang personil di dalam suatu organisasi.

Adapun deskripsi mengenai tugas, wewenang, dan tanggung jawab pada Hotel Setiabudhi Indah yang terlibat dalam sistem informasi reservasi hotel adalah sebagai berikut:

a. Accounting

1. Menerima dan mengolah data dari front office mengenai penerimaan dan pengeluaran yang terjadi setiap harinya.
2. Menyajikan data tersebut guna kebutuhan analisa operasional hotel dalam satu kurun waktu tertentu melalui laporan yang dihasilkan (laporan laba/rugi, neraca).
3. Menghitung dan memberikan upah kepada karyawan setiap bulannya.

b. Room service

1. Menerima telepon dari tamu yang ingin memesan makanan.
2. Mengantar makanan yang dipesan.
3. Memberikan faktur kepada *Front Office* atas pemesanan yang terjadi.

c. Cook

1. Membeli bahan-bahan keprluan sehari-hari untuk memasak.
2. Melaporkan bahan makanan yang hampir habis kepada manager.
3. Memasak makan baik untuk tamu hotel maupun tamu restoran.
4. Memasak sarapan, makan siang, dan makan malam untuk karyawan lainnya.

d. Front Office

1. Menerima telepon yang masuk.

2. Melayani tamu yang datang langsung atau melalui telepon untuk menanyakan tentang reservasi hotel, tipe kamar, dan harga jual kamar.
3. Memperhatikan kondisi lingkungan sekitar (penerangan, musik latar).
4. Bertanggung jawab atas kelancaran operasional pelayanan tamu yang akan *check in* maupun *check out*.
5. Melakukan pengecekan status berjalan (*room status / room rate / room condition*).
6. Melakukan pengecekan status yang akan datang (*meeting room / room rate / walk in guest*).
7. Memeriksa sarana pendukung yang diperlikan, seperti: kondisi kamar, anak kunci, brosur-brosur, formulir, dan peralatan komputer.
8. Berfungsi sebagai *receptionist*, operator dan kasir.

e. Laundry

1. Memberikan pelayanan dalam hal pencucian dan penyetrikaan pakaian.

3.2 Metode Penelitian

Metode penelitian adalah kegiatan untuk memperoleh fakta-fakta atau prinsip-prinsip (baik kegiatan untuk penemuan, pengujian atau pengembangan) dari suatu pengetahuan dengan cara mengumpulkan, mencatat dan menganalisa data yang dikerjakan secara sistematis berdasarkan ilmu pengetahuan (metode ilmiah).

Metode yang digunakan dalam melaksanakan penelitian merupakan dasar penyusunan rancangan penelitian dan merupakan penjabaran dari metode ilmiah secara umum.

3.2.1 Metode Perancangan Sistem

Metode yang digunakan dalam pengembangan perangkat lunak ini adalah dengan metode :

a. Metode Deskripsi, dimana metode ini bersifat memberikan penjelasan, dengan penggunaan metode ini rumusan masalah tentang sistem reservasi hotel yang sedang berjalan pun dapat terjawab.
b. Metode *Action*, dimana metode ini bersifat tindakan, dimana tindakan yang dimaksud adalah tindakan dalam proses pembangunan dan implementasi yang diusulkan pada perusahaan tersebut, dengan penggunaan metode ini maka rumusan masalah tentang bagaimana pembangunan sistem dan implementasi sistem pada hotel tersebut dapat terjawab.

3.2.2 Jenis dan Metode Pengumpulan Data

Agar data dan informasi sesuai dengan permasalahan yang ada pada Hotel Setibudhi Indah, maka penulis menggunakan metode penelitian lapangan yaitu penelitian yang dilakukan dengan cara mendatangi langsung objek penelitian.

3.2.2.1 Sumber Data Primer

Data primer adalah data yang menggunakan metode Penelitian Lapangan (*Field Research*), yaitu penelitian yang dilakukan dengan cara mendatangi

langsung tempat yang dijadikan objek penelitian. Dalam hal ini penulis melakukan pengumpulan data dengan cara :

a. Pengamatan (Observasi)

   Observasi merupakan salah satu teknik pengumpulan fakta atau data (*fact finding technique*) yang cukup efektif untuk mempelajari suatu sistem. Observasi adalah pengamatan langsung suatu kegiatan yang sedang dilakukan.

   Pengamatan yang sudah penulis lakukan yaitu berupa pengamatan mengenai kegiatan-kegiatan atau aktivitas yang telah dan sedang dilakukan berkaitan dengan sistem pengolahan data tamu yang dilakukan oleh bagian *Front Office* pada Hotel Setiabudhi Indah.

b. Wawancara (Interview)

   Wawancara (*interview*) telah diakui sebagai teknik pengumpulan fakta atau data (*fact finding technique*) yang penting dan banyak dilakukan dalam penembangan sistem. Wawancara memungkinkan analis sistem sebagai pewawancara (*interviewer*) untuk mengumpulkan data secara tatap muka langsung dengan orang yang diwawancarai (*people interviewed*).

   Dalam metode pengumpulan data ini penulis mengadakan wawancara langsung dengan bagian *Front Office*, adapun pertanyaan-pertanyaan yang diajukan adalah pertanyaan yang berkaitan dengan masalah pengolahan data tamu di Hotel Setiabudhi Indah untuk mendapatkan data-data dan informasi yang dibutuhkan dalam penyusunan laporan ini.

3.2.2.2 Sumber Data Sekunder

Data sekunder merupakan cara pengumpulan data dengan cara mempelajari data yang telah tersedia atau diberikan oleh pihak yang bersangkutan (pihak perusahaan) kepada penulis.

Cara yang digunakan untuk mengumpulkan data sekunder adalah metode dokumentasi, yaitu dengan mengumpulkan data dan informasi yang diperlukan dari sumber-sumber kebanyakan dari materi sejenis dokumen yang berkenaan dengan masalah yang diteliti. Sumber data tersebut diperoleh dari dokumen-dokumen perusahaan yang dapat digunakan sebagai tahap analisis sistem. Adapun dokumen-dokumen tersebut antara lain: dokumen perusahaan yang berisi sejarah, tujuan, kegiatan dan struktur organisasi, kemudian dokumen form reservasi, dokumen room status, dokumen bukti reservasi, dokumen tagihan laundry, dan dokumen tagihan restoran.

3.2.3 Metode Pendekatan dan Pengembangan Sistem

Metode pendekatan dan pengembangan sistem digunakan untuk memenuhi kebutuhan pengembangan sistem sehingga sistem yang dihasilkan akan sesuai dengan yang diharapkan.

3.2.3.1 Metode Pendekatan Sistem

Melalui pendekatan terstruktur, permasalahan yang komplek di organisasi dapat dipecahkan dan hasil dari sistem akan mudah untuk dipelihara, fleksibel, lebih memuaskan pemakainya, mempunyai dokumentasi yang baik, tepat waktu,

sesuai dengan anggaran biaya pengembangan, dapat meningkatkan produktivitas dan kualitasnya akan lebih baik.

3.2.3.2 Metode Pengembangan Sistem

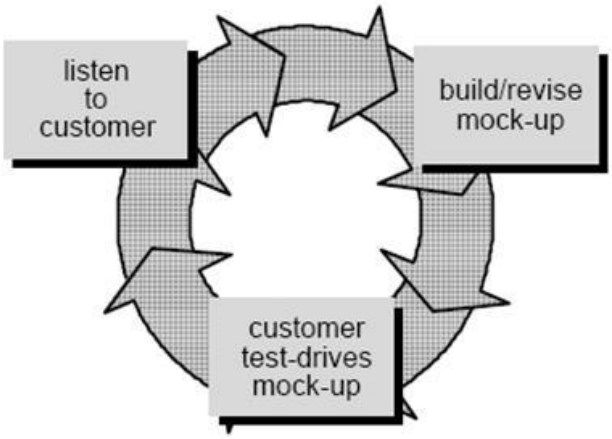
Prototyping merupakan salah satu metode pengembangan perangkat lunak yang banyak digunakan. Dengan metode *prototyping* ini pengembang dan pelanggan dapat saling berinteraksi selama proses pembuatan sistem.

Sering terjadi seorang pelanggan hanya mendefinisikan secara umum apa yang dikehendakinya tanpa menyebutkan secara detail *output* apa saja yang dibutuhkan, pemrosesan dan data-data apa saja yang dibutuhkan. Sebaliknya disisi pengembang kurang memperhatikan efesiensi algoritma, kemampuan sistem operasi dan interface yang menghubungkan manusia dan komputer.

Untuk mengatasi ketidakserasian antara pelanggan dan pengembang, maka harus dibutuhakan kerjasama yanga baik diantara keduanya sehingga pengembang akan mengetahui dengan benar apa yang diinginkan pelanggan dengan tidak mengesampingkan segi-segi teknis dan pelanggan akan mengetahui proses-proses dalm menyelasaikan sistem yang diinginkan. Dengan demikian akan menghasilkan sistem sesuai dengan jadwal waktu penyelesaian yang telah ditentukan.

Kunci agar model *prototype* ini berhasil dengan baik adalah dengan mendefinisikan aturan-aturan main pada saat awal, yaitu pelanggan dan pengembang harus setuju bahwa *prototype* dibangun untuk mendefinisikan kebutuhan. *Prototype* akan dihilangkan sebagian atau seluruhnya dan perangkat

lunak aktual aktual direkayasa dengan kualitas dan implementasi yang sudah ditentukan.



Gambar 3.2 Metode Pengembangan Prototype

Sumber [http://sinau-ben-pinter.blogspot.com/2009/05/software-engineering.html/ 5 April 2011]

Keterangan Gambar:

a. *Prototype* paradigma dimulai dengan mengumpulkan kebutuhan-kebutuhan *customer.*

b. *Developer* dan customer bertemu dan mendefinisikan obyektif *software* secara menyeluruh, mengidentifikasi kebutuhan-kebutuhan yang diketahui dari area pekerjaan.

c. Setelah itu dibuat *quick design. quick design* difokuskan pada representasi aspek *software* yang bisa dilihat *customer/user* (misal: format input dan output). *Quick design* cenderung ke pembuatan *prototype.*

d. *Prototype* dievaluasi customer/user dan digunakan untuk menyempurnakan kebutuhan *software* yang akan dikembangkan.

Tahapan-tahapan *prototyping*

Tahapan-tahapan dalam *prototyping* adalah sebagai berikut:

1. Pengumpulan kebutuhan

   Pelanggan dan pengembang bersama-sama mendefinisikan format seluruh perangkat lunak, mengidentifikasikan semua kebutuhan, dan garis besar sistem yang akan dibuat.

2. Membangun *prototyping*

   Membangun *prototyping* dengan membuat perancangan sementara yang berfokus pada penyajian kepada pelanggan (misalnya dengan membuat input dan format output).

3. Evaluasi *protoptyping*

   Evaluasi ini dilakukan oleh pelanggan apakah *prototyping* yang sudah dibangun sudah sesuai dengan keinginann pelanggan. Jika sudah sesuai maka langkah 4 akan diambil. Jika tidak mengulangu langkah 1, 2 , dan 3.

4. Mengkodekan sistem

   Dalam tahap ini *prototyping* yang sudah di sepakati diterjemahkan ke dalam bahasa pemrograman yang sesuai.

5. Menguji sistem

   Setelah sistem sudah menjadi suatu perangkat lunak yang siap pakai, harus dites dahulu sebelum digunakan. Pengujian ini dilakukan dengan *White Box*, *Black Box*, Basis *Path*, pengujian arsitektur dan lain-lain.

6. Evaluasi Sistem

   Pelanggan mengevaluasi apakah sistem yang sudah jadi sudah sesuai dengan yang diharapkan . Jika ya, langkah 7 dilakukan; jika tidak, ulangi langkah 4 dan 5.

7. Menggunakan sistem

   Perangkat lunak yang telah diuji dan diterima pelanggan siap untuk digunakan .

Keunggulan dan Kelemahan *Prototyping*

a. Keunggulan *prototyping* adalah:
   1. Adanya komunikasi yang baik antara pengembang dan pelanggan.
   2. Pengembang dapat bekerja lebih baik dalam menentukan kebutuhan pelanggan.
   3. Pelanggan berperan aktif dalam pengembangan sistem.
   4. Lebih menghemat waktu dalam pengembangan sistem.
   5. Penerapan menjadi lebih mudah karena pemakai mengetahui apa yang diharapkannya.
b. Kelemahan *prototyping* adalah :
   1. Pelanggan kadang tidak melihat atau menyadari bahwa perangkat lunak yang ada belum mencantumkan kualitas perangkat lunak secara

keseluruhan dan juga belum memikirkan kemampuan pemeliharaan untuk jangja waktu lama.

2. Pengembang biasanya ingin cepat menyelesaikan proyek. Sehingga menggunakan algoritma dan bahasa pemrograman yang sederhana untuk membuat *prototyping* lebih cepat selesai tanpa memikirkan lebih lanjut bahwa program tersebut hanya merupakan cetak biru sistem .
3. Hubungan pelanggan dengan komputer yang disediakan mungkin tidak mencerminkan teknik perancangan yang baik.

(Sumber : http://202.91.15.14/upload/files/3098_MATERI_1.PDF/

5 April 2011)

3.2.3.3 Alat Bantu Analisis dan Perancangan

Alat bantu analisis dan perancangan digunakan untuk memudahkan dalam perancangan sistem informasi reservasi hotel yang dikembangkan penulis adalah :

1. *Flowmap*

Flowmap adalah penggambaran secara grafik dari langkah-langkah dan urut-urutan prosedur dari suatu program. Flowmap dapat didedikasikan untuk menganalisis dan menampilkan interaksi atau aliran data. Jenis data dalam arti khusus ada dua lokasi geografis yang berbeda terhubung ke masing-masing item data: Sebuah lokasi tempat asal aliran dimulai dan lokasi tujuan di mana aliran berakhir. Aliran data itu sendiri dapat orang (misalnya komuter, berbelanja, pengunjung rumah sakit), barang, penggunaan layanan pertanian atau telekomunikasi dan sebagainya. Fungsinya yaitu mendefinisikan hubungan antara

bagian (pelaku proses), proses (manual/berbasis komputer) dan aliran data (dalam bentuk dokumen keluaran dan masukan).

2. Diagram Konteks

Menurut Al-Bahra Bin Ladjamudin (2005 : 64) diagram konteks adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem. Diagram konteks merupaka level tertinggi dari DFD yang menggambarkan seluruh input ke sistem atau output dari sistem. Ia akan memberikan gambaran tentang keseluruhan sistem. Sistem dibatasi oleh boundary (dapat digambarkan dengan garis putus). Dalam diagram konteks hanya ada satu proses. Tidak boleh ada store dalam diagram konteks.

3. Data Flow Diagram (DFD)

Menurut Al-Bahra Bin Ladjamudin (2005 : 64) Diagram aliran data merupakan model dari sistem untuk mengambarkan pembagian sistem ke modul yang lebih kecil. Salah satu keuntungan menggunakan diagram aliran data adalah memudahkan pemakai atau user yang kurang menguasai bidang komputer untuk mengerti sistem yang akan dikerjakan.

4. Kamus Data

Menurut Al-Bahra Bin Ladjamudin (2005 : 70) kamus data sering disebut juga dengan sistem *data dictionary* adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Dengan menggunakan kamus data, analisis sistem dapat mendefinisikan data yang mengalir disistem dengan lengkap. Pada tahap analisis, kamus data digunakan sebagai alat komusikasi antara analisis sistem dengan pemakai sistem tentang data

yang mengalir di sistem, yaitu tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem.

5. Perancangan Basis Data

Definisi basis data menurut Al-bahra Bin Ladjamudin (2005:129) menjelaskan bahwa basis data (*database*) adalah sekumpulan *data store* (bisa dalam jumlah yang sangat besar) yang tersimpan dalam *magnetic disk, optical disk, magnetic drum* atau media penyimpanan skunder lainnya.

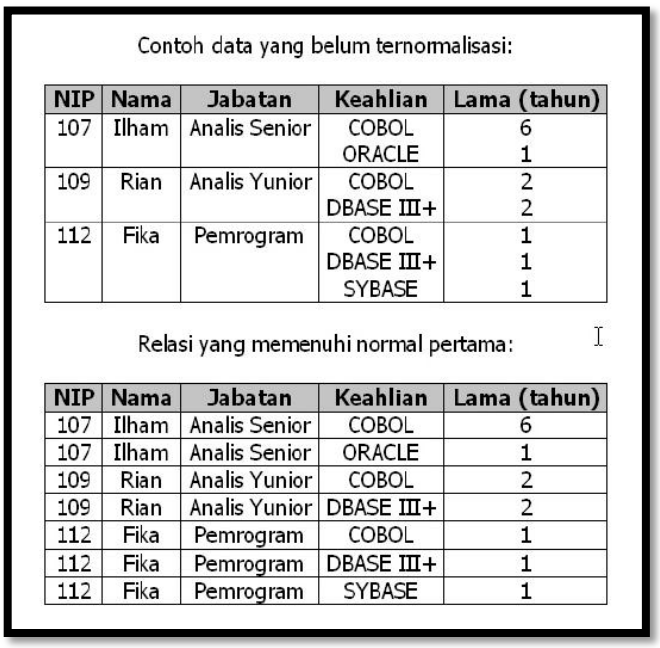
a. Normalisasi

Menurut Al-Bahra Bin Ladjamudin (2005 : 168) Normalisasi adalah proses pengelompokkan data kedalam bentuk tabel atau relasi atau file untuk menyatakan entitas dan hubungan mereka sehingga terwujud satu bentuk *database* yang mudah untuk dimodifikasi.. Hasil dari proses normalisasi adalah himpunan-himpunan data dalam bentuk normal (*normal form*). Ada beberapa bentuk normal, yaitu :

1. Bentuk Normal I (*First Normal Form* / 1-NF).

   Bentuk normal pertama biasa dikenakan pada tabel yang belum ternormalisasi. Tabel yang belum ternomalisasi adalah tabel yang memiliki atribut yang berulang.

   Suatu relasi memenuhi 1-NF jika dan hanya jika setiap atribut dari relasi tersebut hanya memiliki nilai tunggal dalam satu baris atau *record*.



Contoh data yang belum ternormalisasi:

NIP	Nama	Jabatan	Keahlian	Lama (tahun)
107	Ilham	Analis Senior	COBOL ORACLE	6 1
109	Rian	Analis Yunior	COBOL DBASE III+	2 2
112	Fika	Pemrogram	COBOL DBASE III+ SYBASE	1 1 1

Relasi yang memenuhi normal pertama:

NIP	Nama	Jabatan	Keahlian	Lama (tahun)
107	Ilham	Analis Senior	COBOL	6
107	Ilham	Analis Senior	ORACLE	1
109	Rian	Analis Yunior	COBOL	2
109	Rian	Analis Yunior	DBASE III+	2
112	Fika	Pemrogram	COBOL	1
112	Fika	Pemrogram	DBASE III+	1
112	Fika	Pemrogram	SYBASE	1

Gambar 3.3 Contoh Normalisasi Bentuk Normal 1 (1-NF)

(Sumber : http://www.scribd.com/doc/4533135/Normalisasi/ 6 April 2011)

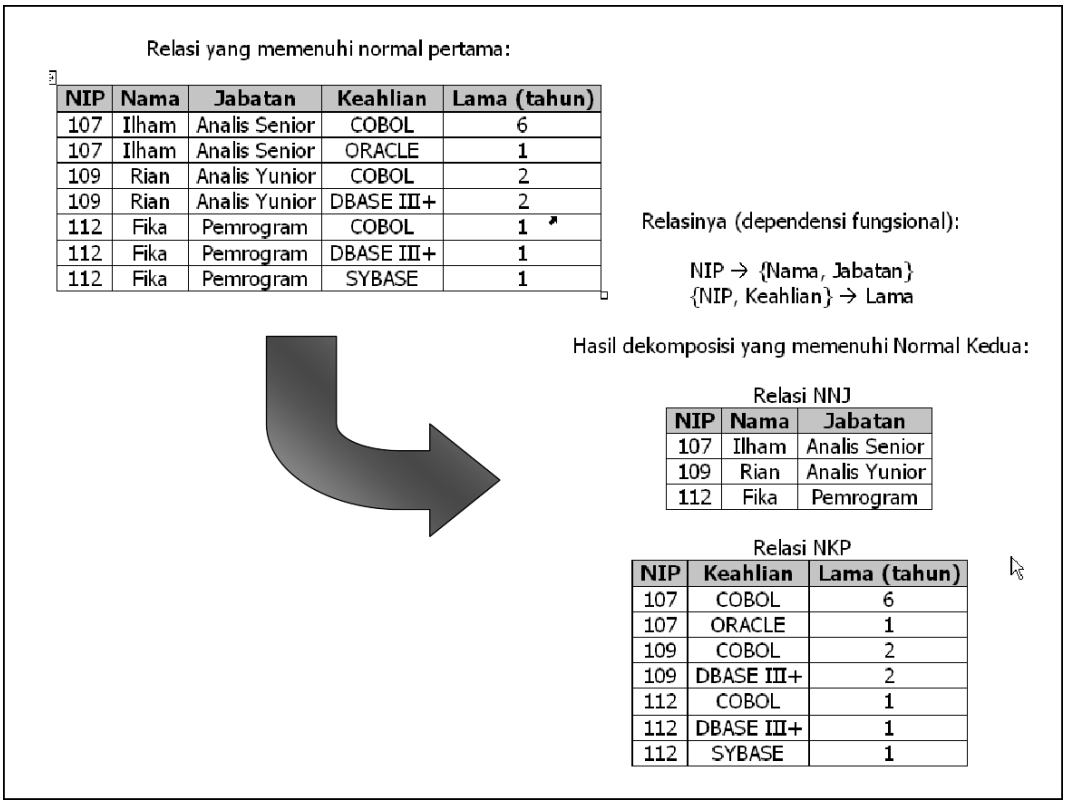
2. Bentuk Normal II (*Second Normal Form* / 2-NF).

   Suatu relasi berada dalam bentuk normal kedua jika dan hanya jika:

   a. Berada pada bentuk normal pertama
   b. Semua atribut yang bukan kunci memiliki dependensi sepenuhnya terhadap kunci primer

   Atribut yang bukan kunci adalah atribut yang bukan merupakan bagian kunci primer. Dalam ungkapan yang lebih praktis, bentuk normal kedua mensyaratkan setiap atribut bergantung kepada kunci primer. Untuk mengubah suatu relasi yang tergolong sebagai bentuk normal pertama ke bentuk normal kedua perlu dilakukan dekomposisi terhadap relasi tersebut. Proses dekomposisi dapat dilakukan dengan menggambarkan diagram

dependensi fungsional terlebih dahulu.Berdasarkan diagram ini, relasi dalam bentuk normal pertama dipecah ke dalam sejumlah relasi.

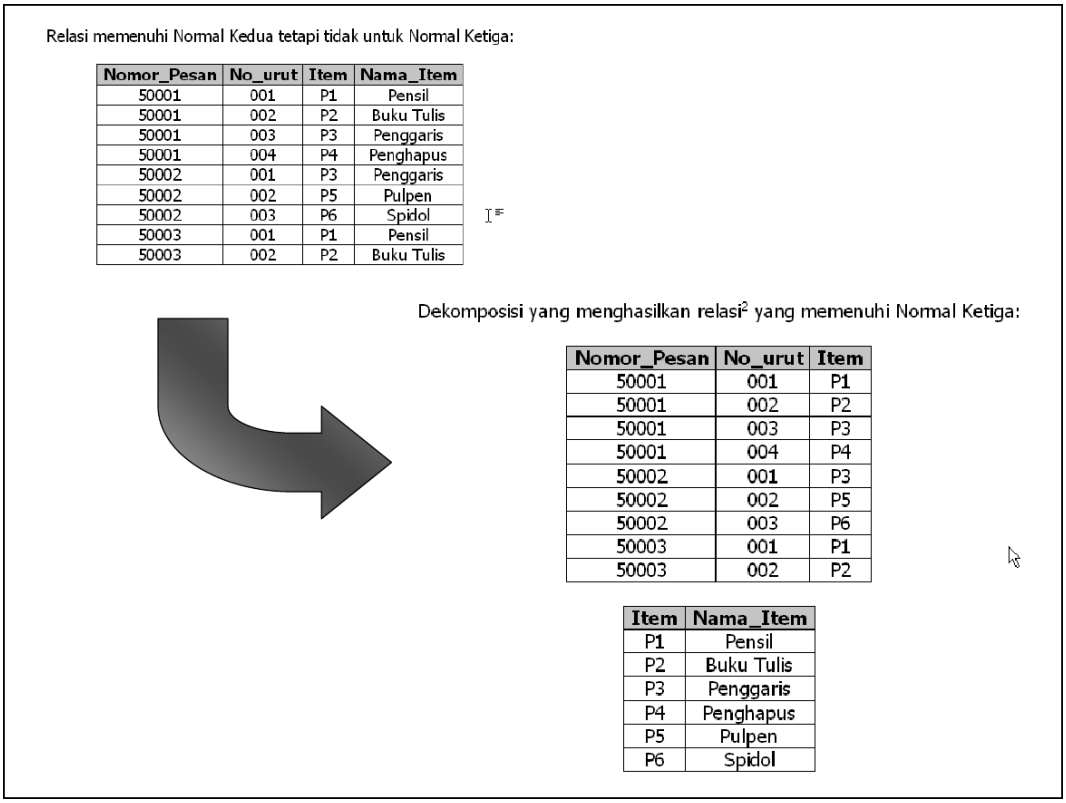


Gambar 3.4 Contoh Normalisasi Bentuk Normal 2 (2-NF)

(Sumber : http://www.scribd.com/doc/4533135/Normalisasi/ 6 April 2011)

3. Bentuk Normal III (*Third Normal Form* / 3-NF).

   Berada Pada bentuk Normal II, dan setiap atribut bukan kunci tidak memiliki *dependensi transitif* terhadap kunci *primer*. Notasi dependensi transitif : $X \rightarrow Y$, $Y \rightarrow Z$ Maka: $X \rightarrow Z$



Gambar 3.5 Contoh Normalisasi Bentuk Normal 3 (3-NF)

(Sumber : http://www.scribd.com/doc/4533135/Normalisasi/ 6 April 2011)

4. Bentuk Normal *Boyce-Codd* (*Boyce-Codd Normal Form* / BCNF).

   Suatu relasi disebut memenuhi bentuk normal Boyce-Codd jika dan hanya jika semua penentu (determinan) adalah kunci kandidat (atribut yang bersifat unik).

   BCNF merupakan bentuk normal sebagian perbaikan terhadap 3NF. Suatu relasi yang memenuhi BCNF selalu memenuhi 3NF, tetapi tidak untuk sebaliknya. Suatu relasi yang memenuhi 3NF belum tentu memenuhi BCNF. Dalam banyak literatur disebutkan bahwa BCNF adalah perbaikan

dari 3NF, karena bentuk normal ketigapun mungkin masih mengandung anomali sehingga masih perlu dinormalisasi lebih lanjut.



Siswa	Kursus	Tutor
Anwar	Bahasa Perancis	Pierre
Anwar	Bahasa Inggris	Richard
Budi	Bahasa Perancis	Pierre
Cecep	Bahasa Inggris	Suzanne

Hasil dekomposisi baru:

Siswa	Tutor
Anwar	Pierre
Anwar	Richard
Budi	Pierre
Cecep	Suzanne

Tutor	Kursus
Pierre	Bahasa Perancis
Richard	Bahasa Inggris
Suzanne	Bahasa Inggris

Gambar 3.6 Contoh Normalisasi Bentuk BCNF

(Sumber : http://www.scribd.com/doc/4533135/Normalisasi/ 6 April 2011)

Pada gambar 3.9, tabel yang pertama telah memenuhi bentuk 3-NF tapi tidak memenuhi BCNF, karena:

a. Seorang siswa dapat mengambil sejumlah kursus bahasa;
b. Setiap tutor hanya mengajar pada sebuah kursus bahasa;
c. Setiap siswa pada setiap kursus bahasa hanya diajar oleh seorang tutor;
d. Suatu kursus bahasa yang sama bisa memiliki lebih dari satu tutor.

Cara mengkonversi relasi yang telah memenuhi 3NF ke BCNF:

a. Carilah semua penentu;

b. Bila terdapat penentu yang bukan berupa kunci kandidat, maka:

   1. Pisahkan relasi tersebut,dan

   2. Buat penentu tersebut sebagai kunci primer.

Kegunaan normalisasi :

1. Meminimasi pengulangan informasi.
2. Memudahkan indentifikasi entitas / obyek.

b. Tabel Relasi

Merupakan hubungan yang terjadi pada suatu tabel dengan tabel yang lainnya, yang berfungsi untuk mengatur operasi suatu database. Hubungan yang dapat dibentuk dapat mencakupi 3 (tiga) macam hubungan yaitu ;

1. One-To-One (1 – 1)

Artinya satu record pada entity A ber-relasi paling banyak satu record juga pada entity B, begitu juga sebaliknya, satu record pada entity B, ber-relasi paling banyak satu record juga dengan entity A. Dalam diagram E-R, relasi ini disimbolkan dengan angka 1.

Contoh Dalam proses belajar mengajar secara privat misalnya, seorang (satu) tutor hanya mengajar satu siswa, begitu juga sebaliknya, satu siswa hanya diajar oleh satu tutor**.**

2. One-To-Many (1 –N)

Artinya satu record pada entity A ber-relasi dengan beberapa record pada entity B, tapi tidak sebaliknya, setiap record pada entity B ber-relasi paling

banyak satu record dengan entity A. Dalam diagram E-R, relasi ini disimbolkan dengan angka 1 untuk menyatakan satu dan huruf M atau N untuk menyatakan banyak.

Contoh : Dalam proses belajar mengajar di sekolah dasar misalnya, satu orang guru mengajar beberapa (banyak) murid, tetapi satu kelas (beberapa murid) hanya di ajar oleh satu guru.

3. **Many-To-Many (N–N)**

   Artinya beberapa record pada entity A dapat berelasi dengan beberapa record juga pada entity B, begitu juga sebaliknya, beberapa record pada entity B dapat ber-relasi dengan beberapa record juga pada entity A.. Dalam diagram E-R, relasi ini disimbolkan dengan huruf M atau N untuk menyatakan banyak.

   Contoh : Dalam hubungan antara mahasiswa dengan dosen pada perguruan tinggi, yaitu seorang seorang dosen mengajar banyak mahasiswa, sebaliknya seorang mahasiswa dapat diajar oleh beberapa dosen, sehingga terjadi hubungan banyak ke banyak.

(Sumber : http://harymulyadi.files.wordpress.com/2010/02/memahami-konsep-database.pdf/ 5 April 2011)

3.2.4 Pengujian Sistem

Pengujian *software* merupakan proses untuk mencari kesalahan pada setiap sistem perangkat lunak, mencatat hasilnya, mengevaluasi setiap aspek pada setiap komponen sistem dan mengevaluasi semua fasilitas dari perangkat lunak yang dikembangkan. Pengujian *software* sangat penting untuk dilakukan.

Pengujian ini bertujuan untuk menjamin kualitas *software,* dan juga menjadi peninjauan terakhir terhadap spesifikasi, desain dan pengkodean.

Penggujian *software* adalah elemen kritis dari jaminan kualitas perangkat lunak dan mempresentasikan kajian pokok dari spesifikasi, desain dan pengkodean. Metode pengujian perangkat lunak yang penyusun pakai dalam pengujian *software* sistem informasi reservasi hotel pada Hotel setiabudhi Indah adalah metode *black box* yang mana *software* diuji setelah *software* selesai di buat.

Black box testing berfokus pada persyaratan fungsional *software* dengan demikian memungkinkan perekayasa yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Pengujiam *black box* berusaha menentukan kesalahan dalam kategori ;

1. Fungsi-fungsi yang tidak benar atau hilang.
2. Kesalahan interface.
3. Kesalahan dalam struktur data atau akses database eksternal.
4. Kesalahan kerja.
5. Inisialisasi dan kesalahatan terminasi.