

BAB II TINJAUAN PUSTAKA

2.1 Konsep Sistem Informasi

Sistem adalah sekelompok komponen yang saling berhubungan, bekerja bersama untuk mencapai tujuan bersama dengan menerima *input* serta menghasilkan *output* dalam proses transformasi yang teratur.[1]

Sistem dapat juga didefinisikan sebagai sebuah tatanan (keterpaduan) yang terdiri atas sejumlah komponen fungsional (dengan satuan fungsi/tugas khusus) yang saling berhubungan dan secara bersama-sama bertujuan untuk memenuhi suatu proses/pekerjaan tertentu. [2]

Model umum sebuah sistem adalah terdiri dari *input*, proses, dan *output*. Hal ini merupakan konsep sebuah sistem yang sangat sederhana, sebab sebuah sistem dapat mempunyai beberapa masukan dan keluaran. Sistem yang dinamis memiliki tiga komponen atau fungsi dasar yang berinteraksi:

1. *Input*, melibatkan penangkapan dan perakitan berbagai elemen yang memasuki sistem untuk diproses. Contohnya bahan baku mentah dan usaha manusia harus terjamin dan diatur untuk pemrosesan.
2. Proses, melibatkan proses transformasi yang mengubah *input* menjadi *output*. Contohnya proses manufaktur, proses bernafasnya manusia, atau perhitungan matematika.
3. *Output*, melibatkan perpindahan elemen yang telah diproduksi oleh proses transformasi ke tujuan akhirnya. Contohnya barang jadi, layanan oleh manusia, dan informasi manajemen harus dipindahkan ke para pemakainya.

Informasi dapat didefinisikan sebagai data yang telah diubah menjadi konteks yang berarti dan berguna bagi para pemakai akhir tertentu. Jadi, informasi dapat dikatakan sebagai data yang telah diolah dan dimasukkan dalam konteks yang memberikan arti/nilai bagi pemakai akhir tertentu yang berkepentingan. [1]

Dapat disimpulkan bahwa sistem informasi merupakan kombinasi dari manusia, fasilitas atau alat teknologi, media, prosedur, dan pengendalian yang bermaksud menata jaringan komunikasi yang penting, proses atas transaksi-transaksi

tertentu dan rutin, membantu manajemen dan pemakai *intern* dan *extern*, serta menyediakan dasar pengambilan keputusan yang tepat.

2.2 Analisis dan Perancangan Sistem

Analisis sistem adalah studi masalah bisnis domain untuk merekomendasikan perbaikan dan menentukan kebutuhan bisnis dan prioritas untuk solusi.[3] Analisis sistem ini dimaksudkan untuk memberikan tim proyek dengan pemahaman yang lebih menyeluruh tentang masalah dan kebutuhan yang memicu proyek.

Analisa sistem adalah sebagai penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya.[4] Perancang sistem adalah penentuan proses dan data yang diperlukan oleh sistem baru, jika sistem itu berbasis komputer, perancangan dapat menyertakan spesifikasi peralatan yang akan digunakan. [5]

Dari Pendapat pendapat di atas, maka dapat disimpulkan bahwa perancangan sistem merupakan proses menerjemahkan kebutuhan pemakai informasi ke dalam suatu rancangan untuk memenuhi kebutuhan pemakai dan memberi gambaran yang lebih jelas untuk dijadikan pertimbangan.

2.3 Use Case Diagram

Use case diagram adalah teknik untuk merekam persyaratan fungsional sebuah sistem. *Use case* mendeskripsikan interaksi tipikal antara para pengguna sistem dengan sistem itu sendiri, dengan memberi sebuah narasi tentang bagaimana sistem tersebut digunakan.

Dalam bahasan *use case*, para pengguna disebut sebagai aktor. Aktor merupakan sebuah peran yang dimainkan seorang pengguna dalam kaitannya dengan sistem. Aktor dapat meliputi pelanggan, petugas layanan konsumen, manajer penjualan, dan analis produk. Aktor-aktor tersebut menggunakan *use case*. Seorang aktor dapat menggunakan banyak *use case*, sebaliknya, sebuah *use case* dapat digunakan oleh beberapa aktor. Biasanya, Anda memiliki banyak pelanggan, jadi

banyak orang yang menjadi aktor pelanggan. Selain itu, pengguna dapat menjadi lebih dari satu aktor, misalnya seorang manajer penjualan yang menangani tugas-tugas petugas layanan konsumen. Aktor tidak harus manusia. Jika sebuah sistem melakukan sebuah layanan untuk sistem komputer lain, sistem lain tersebut merupakan aktor. [6]

Use case awalnya ditentukan pada tahap persyaratan dari siklus hidup dan akan diperbaiki secara bertahap di sepanjang siklus hidup. Selama penemuan persyaratan, *use case* digunakan untuk menangkap *essense* masalah-masalah bisnis dan untuk memodelkan (pada tingkat tinggi) fungsionalitas dari sistem yang diajukan. Selain itu, *use case* merupakan titik awal untuk mengidentifikasi entiti data atau objek sistem.[3]

2.4 Konsep Basis Data

Basis data sendiri dapat didefinisikan dalam sejumlah sudut pandang seperti:

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar dapat dimanfaatkan kembali dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
3. Kumpulan *file*/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

Secara lebih lengkap, pemanfaatan basis data dilakukan untuk memenuhi sejumlah tujuan (objektif) seperti berikut ini:

1. Kecepatan dan Kemudahan (*speed*)

Pemanfaatan basis data memungkinkan untuk dapat menyimpan data atau melakukan perubahan/manipulasi terhadap data atau menampilkan kembali data tersebut dengan lebih cepat dan mudah, daripada jika menyimpan data secara manual (non elektronik) atau secara elektronik (tetapi tidak dalam bentuk penerapan basis data, misalnya dalam bentuk *spread sheet* atau dokumen teks biasa).

2. Efisiensi Ruang Penyimpanan (*Space*)

Karena keterkaitan yang erat antar kelompok data dalam sebuah basis data, maka redundansi (pengulangan) data pasti akan selalu ada. Banyaknya redundansi ini tentu akan memperbesar ruang penyimpanan (baik di memori utama maupun memori sekunder) yang harus disediakan. Dengan basis data, efisiensi/optimalisasi penggunaan ruang penyimpanan dapat dilakukan, karena dapat melakukan penekanan jumlah redundansi data, baik dengan menerapkan sejumlah pengkodean atau dengan membuat relasi-relaksi (dalam bentuk *file*) antar kelompok data yang saling berhubungan.

3. Keakuratan (*Accuracy*)

Pemanfaatan pengkodean atau pembentukan relasi antar data bersama dengan penerapan aturan/batasan (*constraint*) tipe data, domain data, keunikan data, dan sebagainya, yang secara ketat dapat diterapkan dalam sebuah basis data, sangat berguna untuk menekan ketidakakuratan pemasukan/penyimpanan data.

4. Ketersediaan (*Availability*)

Pertumbuhan data (baik dari sisi jumlah maupun jenisnya) sejalan dengan waktu akan semakin membutuhkan ruang penyimpanan yang besar. Padahal tidak semua data itu selalu digunakan/dibutuhkan. Karena itu, dapat dipilah adanya data utama/*master*/referensi, data transaksi, data histori, hingga data kadaluarsa. Data yang sudah jarang atau bahkan tidak pernah lagi digunakan dapat diatur untuk dilepaskan dari sistem basis data yang sedang aktif (menjadi *off-line*), baik dengan cara penghapusan atau dengan memindahkannya ke media penyimpanan *off-line* (seperti *removable disk* atau *tape*). Di sisi lain, karena kepentingan pemakaian data, sebuah basis data dapat memiliki data yang disebar di banyak lokasi geografis. Data nasabah sebuah bank, misalnya, dipisah-pisah dan disimpan di lokasi yang sesuai dengan keberadaan nasabah. Dengan pemanfaatan teknologi jaringan komputer, data yang berada di suatu lokasi/cabang, dapat juga diakses (menjadi tersedia/*available*) bagi lokasi/cabang lain.

5. Kelengkapan (*Completeness*)

Lengkap/tidaknya data yang dikelola dalam sebuah basis data bersifat relatif (baik terhadap kebutuhan pemakai maupun terhadap waktu). Bila seorang pemakai sudah menganggap bahwa data yang dipelihara sudah lengkap, maka pemakai yang lain belum tentu berpendapat sama. Atau yang sekarang dianggap

sudah lengkap, belum tentu di masa yang akan datang juga demikian. Dalam sebuah basis data, disamping data, juga harus menyimpan struktur (baik yang mendefinisikan objek-objek dalam basis data maupun definisi detail dari tiap objek, seperti struktur *file*/tabel atau indeks). Untuk mengakomodasi kebutuhan kelengkapan data yang semakin berkembang, maka tidak hanya dapat menambah *record-record* data, tetapi juga dapat melakukan perubahan struktur dalam basis data, baik dalam bentuk penambahan objek baru (tabel) atau dengan penambahan *field-field* baru pada suatu tabel.

6. Keamanan (*Security*)

Memang ada sejumlah sistem (aplikasi) pengelola basis data yang tidak menerapkan aspek keamanan dalam penggunaan basis data. Tetapi untuk sistem yang besar dan serius, aspek keamanan juga dapat diterapkan dengan ketat. Dengan begitu, dapat ditentukan siapa-siapa (pemakai) yang boleh menggunakan basis data beserta objek-objek di dalamnya dan menentukan jenis-jenis operasi apa saja yang boleh dilakukannya.

7. Kebersamaan Pemakaian (*Sharability*)

Pemakai basis data seringkali tidak terbatas pada satu pemakai saja, atau di satu lokasi saja atau oleh satu sistem/aplikasi saja. Data pegawai dalam basis data kepegawaian, misalnya, dapat digunakan oleh banyak pemakai, dari sejumlah departemen dalam perusahaan atau oleh banyak sistem (sistem penggajian, sistem akuntansi, sisten inventori, dan sebagainya). Basis data yang dikelola oleh sistem (aplikasi) yang mendukung lingkungan *multiuser* dapat memenuhi kebutuhan ini, tetapi tetap dengan menjaga/menghindari terhadap munculnya persoalan baru seperti inkonsistensi data (karena data yang sama diubah oleh banyak pemakai pada saat yang bersamaan) atau kondisi *deadlock* (karena ada banyak pemakai yang saling menunggu untuk menggunakan data).

Salah satu tujuan dari DBMS (*Database Management System*/Manajemen Sistem Basis Data) adalah untuk menyediakan fasilitas antarmuka (*interface*) dengan melihat/menikmati data (yang lebih ramah/*user oriented*) kepada pemakai/*user*. Untuk itu, sistem tersebut akan menyembunyikan detail tentang bagaimana data tersebut akan menyembunyikan detail tentang bagaimana data disimpan dan dipelihara. Karena itu, seringkali data yang terlihat oleh pemakai sebelumnya

berbeda dengan yang tersimpan secara fisik. Abstraksi data merupakan tingkatan/level bagaimana melihat data dalam sebuah sistem basis data.

Ada 3(tiga) level abstraksi data:

1. Level Fisik (*Physical Level*)

Merupakan level terendah dalam abstraksi data, yang menunjukkan bagaimana sesungguhnya suatu data disimpan. Pada level ini, pemakai melihat data sebagai gabungan dari struktur dan datanya sendiri. Pemakai juga berkompeten dalam mengetahui bagaimana representasi fisik dari penyimpanan/pengorganisasian data. Pada level ini berurusan dengan data sebagai teks, sebagai angka, atau bahkan melihatnya sebagai himpunan bit data.

2. Level Logik/Konseptual (*Conceptual Level*)

Merupakan level berikutnya dalam abstraksi data yang menggambarkan data apa yang sebenarnya (secara fungsional) disimpan dalam basis data dan hubungannya dengan data yang lain. Pemakai pada level ini yang misalnya, mengetahui bahwa data pegawai disimpan/direpresentasikan dalam beberapa *file*/tabel, seperti *file* pribadi, *file* pendidikan, *file* pekerjaan, *file* keluarga, dan sebagainya.

3. Level Penampakan (*View Level*)

Merupakan level tertinggi dari abstraksi data yang hanya menunjukkan sebagian dari basis data. Banyak *user* dalam sistem basis data tidak akan terlibat (*concern*) dengan semua data/informasi yang ada/disimpan. Para *user* umumnya hanya membutuhkan sebagian data/informasi dalam basis data yang kemunculannya di mata pemakai diatur oleh aplikasi *end-user*. Aplikasi ini juga yang mengkonversi data asli/fisik menjadi data bermakna/logik pada pemakai. Misalnya, data hari yang disimpan dalam bentuk kode (1 untuk Senin, 2 untuk Selasa, dan seterusnya) yang kemudian ditampilkan bukan dalam bentuk kodenya (1,2,3, dan seterusnya), tetapi sudah dalam bentuk nama harinya (Senin, Selasa, Rabu, dan seterusnya). Data yang ‘dinikmati’ pemakai juga bahkan sama sekali berbeda dengan representasi fisiknya, misalnya untuk data yang dapat divisualkan sebagai gambar, data yang dapat diperdengarkan sebagai suara, dan sebagainya. Data yang diperlihatkan juga bisa saja tidak berasal dari hanya sebuah tabel, tetapi mewakili relasi antar tabel, tetapi bagi pemakai yang menggunakannya terasa sebagai satu kesatuan data yang kompak.

Sebuah sistem pengelola basis data (DBMS) umumnya memiliki sejumlah komponen fungsional (modul) seperti:

1. *File Manager*, yang mengelola alokasi ruang dalam *disk* dan struktur data yang dipakai untuk merepresentasikan informasi yang tersimpan dalam *disk*. Sebenarnya sistem operasi (tempat dimana DBMS diaktifkan) juga memiliki modul *File Manager*. Tetapi *File Manager* di DBMS lebih difokuskan pada efisiensi dan efektivitas penyimpanan.
2. *Database Manager*, yang menyediakan *interface* antara data *low level* yang ada di basis data dengan program aplikasi dan *query* yang diberikan ke sistem.
3. *Query Processor*, yang menerjemahkan perintah-perintah dalam *query language* ke perintah *low-level* yang dapat dimengerti oleh *database manager*. Disamping itu, *Query Processor* akan mentransformasikan permintaan *user* ke bentuk yang lebih efisien, sehingga *query* menjadi lebih efektif.
4. *DML Precompiler*, yang mengkonversi perintah DML yang ditambahkan dalam sebuah program aplikasi ke pemanggilan prosedur normal dalam bahasa induk. *Precompiler* ini akan berinteraksi dengan *query processor*.
5. *DDL Compiler*, yang mengkonversi perintah-perintah DDL ke dalam sekumpulan tabel yang mengandung *metadata*. Tabel-tabel ini kemudian disimpan dalam kamus data.[2]

Sebuah sistem basis data pada dasarnya adalah komputerisasi sistem penyimpanan catatan. Basis data itu sendiri dapat dianggap sebagai jenis lemari elektronik, yaitu adalah sebuah tempat penyimpanan atau wadah untuk koleksi komputerisasi data arsip. Pengguna sistem ini dapat melakukan beragam operasi terhadap berkas, sebagai contoh:

1. Menambahkan berkas baru, berkas kosong pada basis data
2. Memasukan data kedalam berkas yang ada
3. Mengambil kembali data dari berkas yang ada
4. Mengganti data dalam berkas yang ada
5. Menghilangkan data dari berkas yang ada
6. Memindahkan berkas yang ada dari basis data

Sistem basis data tersedia pada mesin-mesin yang berkisar dari komputer perorangan terkecil hingga *mainframe* terbesar. Tidak perlu dikatakan, fasilitas yang

disediakan oleh sistem manapun ditentukan hingga luasan tertentu oleh ukuran dan daya mesin yang mendasar. Khususnya, sistem pada mesin besar (“sistem besar”) cenderung digunakan oleh banyak orang (“*multi-user*”), dimana mesin yang lebih kecil (“sistem kecil”) cenderung dipakai satu orang (“*single-user*”). Sistem pemakai tunggal adalah sebuah sistem dimana paling banyak satu orang dapat membuat akses pada basis data pada waktu tertentu; sistem pamakai banyak adalah sistem dimana banyak pemakai dapat membuat akses pada basis data pada waktu yang sama. [7]

2.5 *Mobile Application*

Aplikasi *mobile* biasanya membantu para penggunanya untuk terkoneksi dengan layanan *internet* yang biasa diakses pada PC atau mempermudah mereka untuk menggunakan aplikasi *internet* pada piranti yang bisa dibawa.

Menurut Buyens, aplikasi adalah suatu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktivitas. Misalnya termasuk perangkat lunak perusahaan, software akuntansi, perkantoran, grafis perangkat lunak dan pemutar *media*. Dapat disimpulkan bahwa aplikasi merupakan *software* yang berfungsi untuk melakukan berbagai bentuk pekerjaan atau tugas-tugas tertentu seperti penerapan, penggunaan dan penambahan data.

Program aplikasi merupakan program siap pakai. Program yang direka untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain. Contoh aplikasi ialah program pemroses kata dan *web browser*. Aplikasi akan menggunakan sistem operasi (OS) komputer dan aplikasi yang lainnya mendukung. Klasifikasi aplikasi dapat dibagi menjadi 2 (dua) yaitu :

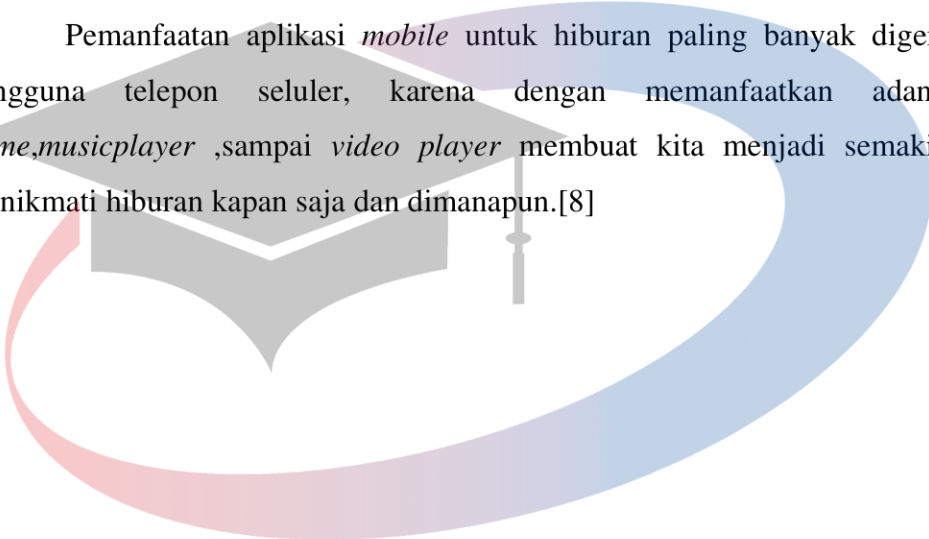
1. Aplikasi *software* spesialis, program dengan dokumentasi tergabung yang dirancang untuk menjalankan tugas tertentu.
2. Aplikasi paket, dengan dokumentasi tergabung yang dirancang untuk jenis masalah tertentu.

Menurut Buyens Aplikasi *mobile* berasal dari kata *application* dan *mobile*. *Application* yang artinya penerapan, lamaran, penggunaan. Secara istilah aplikasi adalah program siap pakai yang di reka untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain dan dapat digunakan oleh sasaran yang dituju

sedangkan *mobile* dapat di artikan sebagai perpindahan dari suatu tempat ke tempat lain.

Kata *mobile* mempunyai arti bergerak atau berpindah, sehingga aplikasi *mobile* menurut Rangsang Purnama adalah sebutan untuk aplikasi yang berjalan di *mobile device*. Dengan menggunakan aplikasi *mobile* ,dapat dengan mudah melakukan berbagai macam aktifitas mulai dari hiburan, berjualan, belajar, mengerjakan pekerjaan kantor, browsing dan lain sebagainya.

Pemanfaatan aplikasi *mobile* untuk hiburan paling banyak digemari oleh pengguna telepon seluler, karena dengan memanfaatkan adanya fitur *game, musicplayer* ,sampai *video player* membuat kita menjadi semakin mudah menikmati hiburan kapan saja dan dimanapun.[8]



UNIVERSITAS
MIKROSKIL