

BAB II

LANDASAN TEORI

2.1. Pengertian Puskesmas

Puskesmas adalah unit pelaksana teknis dinas kesehatan kabupaten/kota yang bertanggung jawab menyelenggarakan pembangunan kesehatan di suatu wilayah kerja (Kebijakan Dasar Puskesmas Depkes RI, 2004)

2.2. Rekam Medik

Menurut Hatta (2008) rekam medis adalah berkas yang berisikan catatan dan dokumen tentang identitas pasien, pemeriksaan, pengobatan, tindakan dan pelayanan lain kepada pasien pada sarana pelayanan kesehatan.

Menurut Hatta (2008) tujuan utama (primer) rekam medis terbagi menjadi 5 kepentingan yaitu untuk:

- a. Pasien, rekam kesehatan merupakan alat bukti utama yang mampu membenarkan adanya pasien dengan identitas yang jelas dan telah mendapatkan berbagai pemeriksaan dan pengobatan di sarana pelayanan kesehatan dengan segala hasil serta konsekuensi biayanya.
- b. Pelayanan pasien, rekam kesehatan mendokumentasikan pelayanan yang diberikan oleh tenaga kesehatan, penunjang medis dan tenaga lain yang bekerja dalam berbagai fasilitas pelayanan kesehatan. Dengan demikian rekaman itu membantu pengambilan keputusan terapi, tindakan, dan penentuan diagnosis pasien.

- c. Manajemen pelayanan, rekam kesehatan yang lengkap memuat segala aktivitas yang terjadi dalam manajemen pelayanan sehingga digunakan dalam menganalisis berbagai penyakit, menyusun pedoman praktik, serta untuk mengevaluasi mutu yang diberikan.
- d. Menunjang pelayanan, rekam kesehatan yang rinci akan mampu menjelaskan aktivitas yang berkaitan dengan penanganan sumber-sumber yang ada pada organisasi pelayanan di rumah sakit, menganalisa kecenderungan yang terjadi dan mengomunikasikan informasi diantara klinik yang berbeda.
- e. Pembiayaan, rekam kesehatan yang akurat mencatat segala pemberian pelayanan kesehatan yang diterima pasien. Informasi ini menentukan besarnya pembayaran yang harus dibayar, baik secara tunai atau melalui asuransi.

2.3. Sistem Informasi

2.3.1. Konsep Dasar Sistem

Telah diketahui bahwa informasi merupakan hal yang sangat penting bagi manajemen di dalam pengambilan keputusan. Ada pertanyaan dari mana informasi tersebut bisa didapatkan?. Jawabannya adalah dari sistem informasi atau disebut juga *processing system* atau *information system* atau *information-generating system*.

2.3.1.1. Pengertian Sistem

Sistem merupakan kumpulan dari objek-objek seperti manusia, sumber daya, konsep, dan prosedur untuk melakukan suatu fungsi atau tujuan (Turban, 1998). Sistem ini terbagi menjadi tiga bagian: input, proses, dan output. Bagian-bagian tersebut dikelilingi oleh lingkungan dan selalu meliputi mekanisme umpan

balik. Sebagai contoh, pengambil keputusan dapat dibagi dikategorikan sebagai sebuah sistem.

1. *Input* merupakan elemen-elemen yang memasuki sistem untuk diolah oleh sistem sehingga menghasilkan output. Input dapat berupa data, *raw material* ataupun manusia.
2. *Process* merupakan semua elemen yang memungkinkan untuk mengubah input atau output.
3. *Output* merupakan produk jadi atau suatu konsekuensi berada dalam sistem.
4. *Feedback* merupakan sebuah aliran data dari output ke pengambilan keputusan yang memusatkan pada sistem output atau kinerja. Berdasarkan informasi ini pengambil keputusan yang bertindak sebagai control memutuskan untuk mengubah input, proses atau keduanya.
5. *Environment of the system* (lingkungan sistem) terbentuk dari beberapa elemen yang berada di luar sistem, tidak termasuk input, proses, dan output. Meskipun berada di luar sistem, *environment* mempengaruhi kinerja dari suatu sistem dalam mencapai tujuan.
6. *Boundary* merupakan pemisah antara sistem dan lingkungan. Sistem berada di dalam *boundary*, sedangkan *environment* berada di luarnya.

2.3.1.2. Karakteristik Sistem

Menurut pendapat Jogianto Hartono dalam bukunya yang berjudul *Analisis dan Desain Sistem Informasi: Pendekatan Terstruktur, Teori dan Praktik Aplikasi Bisnis* berpendapat bahwa. sistem mempunyai beberapa karakteristik atau sifat sifat tertentu, antara lain :

1. Komponen Sistem (*Component*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang saling bekerja sama membentuk suatu komponen sistem atau bagian-bagian dari sistem.

2. Batasan Sistem (*Boundary*)

Merupakan daerah yang membatasi suatu sistem dengan sistem yang lain atau dengan lingkungan kerjanya.

3. Lingkungan Luar Sistem (*InterfacEnvironmente*)

Suatu sistem yang ada di luar dari batas sistem yang dipengaruhi oleh operasi sistem.

4. Penghubung Sistem (*Interface*)

Media penghubung antara suatu subsistem dengan subsistem lain. Adanya penghubung ini memungkinkan berbagai sumber daya mengalir dari suatu subsistem ke subsistem lainnya.

5. Masukan Sistem (*Input*)

Energi yang masuk ke dalam sistem, berupa perawatan dan sinyal. Masukan perawatan adalah energi yang dimasukkan supaya sistem tersebut dapat berinteraksi.

6. Keluaran Sistem (*Output*)

Hasil energi yang diolah dan diklasifikasikan menjadi keseluruhan yang berguna dan sisa pembuangan.

7. Pengolahan Sistem (*Process*)

Suatu sistem dapat mempunyai suatu bagian pengolahan yang akan mengubah masukan menjadi keluaran.

8. Sasaran Sistem (*Object*)

Tujuan yang ingin dicapai oleh sistem, akan dikatakan berhasil apabila mengenai sasaran atau tujuan.

2.3.1.3. Klasifikasi Sistem

Menurut pendapat Jogianto Hartono dalam bukunya yang berjudul *Analisis dan Desain Sistem Informasi: Pendekatan Terstruktur, Teori dan Praktik Aplikasi Bisnis* berpendapat bahwa suatu sistem dapat diklasifikasikan menjadi seperti berikut:

1. Sistem abstrak dan sistem fisik

Sistem abstrak adalah suatu sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, sedangkan sistem fisik adalah sistem yang ada secara fisik.

2. Sistem alamiah dan sistem buatan manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam sedangkan sistem buatan manusia adalah sistem yang dirancang oleh manusia.

3. Sistem tertentu dan sistem tak tentu

Sistem tertentu adalah suatu sistem yang operasinya dapat diprediksi secara tepat sedangkan sistem tak tertentu adalah sistem dengan perilaku ke depan yang tidak dapat diprediksi.

4. Sistem tertutup dan sistem terbuka

Sistem tertutup adalah sistem yang tidak terpengaruh oleh lingkungan luar atau otomatis, sedangkan sistem terbuka adalah sistem yang berhubungan dan terpengaruh oleh lingkungan luar.

2.3.1.4. Pengendalian Sistem

Karena suatu sistem tidak ada yang tertutup, maka sistem harus mempunyai daya membela diri atau system harus mempunyai sistem pengendalian. Pengendalian dari suatu system dapat berupa pengendalian umpan balik (*feedback control system*), pengendalian umpan maju (*feed forward control system*) dan pengendalian pencegahan (*preventive control system*).

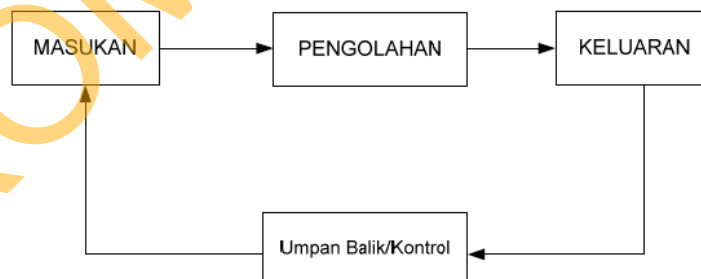
a. Sistem Pengendalian Umpan Balik

Bentuk dasar dari sistem yang sederhana terdiri dari masukan, pengolahan dan keluaran yang tidak menyediakan suatu system pengendalian.



Gambar 2.1 Bentuk Dasar Suatu Sistem

Untuk maksud pengendalian, dapat ditambahkan suatu system pengendalian umpan balik sebagai berikut ini.



Gambar 2.2 Sistem Pengendalian Umpan Balik

Pengendalian umpan balik merupakan proses mengukur keluaran dari sistem yang dibandingkan dengan suatu standar tertentu, bilamana terjadi perbedaan

perbedaan atau penyimpangan-penyimpangan akan dikoreksi untuk memperbaiki masukan system selanjutnya.

b. Sistem Pengendalian Umpan Maju

Sistem pengendalian umpan maju (*feed forward control system*) disebut juga dengan istilah (*positif feedback*) umpan balik positif. *positif feedback* mencoba mendorong proses dari sistem supaya menghasilkan hasil balik yang positif. Sistem ini merupakan perkembangan dari sistem pengendalian umpan balik.

c. Sistem Pengendalian Pencegahan

Sistem pengendalian pencegahan mencoba untuk mengendalikan sistem dimuka sebelum proses dimulai dengan mencegah hal-hal yang merugikan untuk masuk ke dalam sistem. Sistem pengendalian intern (*internal control*) merupakan contoh penerapan dari sistem pengendalian pencegahan.

2.3.2. Konsep Dasar Informasi

Informasi merupakan salah satu bagian yang sangat penting dalam suatu organisasi, jika informasi kurang baik maka sistem akan tidak berjalan dengan baik. Sumber informasi data adalah kumpulan fakta-fakta atau kejadian-kejadian yang nyata. Kejadian adalah sesuatu yang terjadi pada saat tertentu.

2.3.2.1. Definisi Informasi

Menurut Davis dalam Kadir (2003:28), informasi adalah “data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat bagi pengambilan keputusan saat ini atau saat mendatang”.

Hal yang sama dikemukakan oleh Suyanto (2000: 6), Informasi adalah “data yang telah diletakkan dalam konteks yang lebih berarti dan berguna yang dikomunikasikan kepada penerima untuk digunakan di dalam pembuatan keputusan”

2.3.2.2. Pengembangan Informasi

Pengembangan informasi dapat berarti menyusun sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau untuk memperbaiki sistem yang sudah ada. Sistem yang sudah lama perlu diperbaiki atau bahkan diganti, dapat disebabkan oleh beberapa hal, diantaranya yaitu :

1. Kesalahan yang tidak sengaja, yang menyebabkan kebenaran data kurang terjamin.
2. Tidak efisiensinya operasi pengolahan data tersebut.
3. Adanya instruksi-instruksi atau kebijaksanaan yang baru baik dari pemimpin atau dari luar organisasi seperti peraturan pemerintah.

2.3.2.3. Nilai Informasi

Dari beberapa pendapat di atas yang mengemukakan definisi dari arti informasi, maka informasi mencakup data yang diberi konteks, kemudian diolah untuk disajikan sehingga diterima sebagai sebuah informasi, yang dapat menambah pengetahuan seseorang. Sedangkan pengertian nilai itu sendiri adalah “sesuatu yang berharga, bermutu, menunjukkan kualitas, dan berguna bagi manusia”. Sesuatu itu bernilai berarti sesuatu itu berharga atau berguna bagi kehidupan manusia.

Menurut Jogiyanto (2005:31), “nilai informasi ditentukan oleh dua hal, yaitu manfaat dan biaya mendapatkannya. Informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya. Nilai informasi secara nyata memiliki karakteristik khusus terhadap tingkat ukuran, kebutuhan, dinamika, kemanfaatan dan keterpakaian informasi itu sendiri. Tetapi nilai tersebut tidak dapat diukur secara nyata”.

Dari pendapat di atas maka dapat diketahui bahwa nilai informasi sangat tergantung pada isi, cara perolehan dan manfaatnya bagi pengguna dalam mendukung aktifitas yang sedang ia lakukan. Hal ini didukung oleh pendapat yang menyatakan bahwa pada umumnya nilai informasi harus mencakup

1. Isi informasi (luas bidang cakupan)
2. Kecermatan pembuatan dan format penyajian
3. Kemutakhiran informasi (*up-to-dateness*)
4. Kualitas informasi (kredibilitas dan akseptibilitas)
5. Frekuensi penyajian informasi (Suryana yang dikutip oleh Koswara, 1998:102)

2.3.2.4. Pengukuran Nilai Informasi

Sebagian besar informasi tidak dapat ditafsir keuntungannya dengan nilai uang, tetapi dapat ditafsir nilai efektifitasnya. Untuk menentukan nilai suatu informasi maka dapat ditentukan berdasarkan sifatnya.

Menurut Sutabri (2005:31) sifat atau karakteristik yang dapat menentukan nilai informasi dapat dijabarkan sebagai berikut :

1. Mudah Diperoleh

Informasi mempunyai nilai yang lebih sempurna apabila dapat diperoleh secara mudah. Informasi yang penting dan sangat dibutuhkan menjadi tidak bernilai jika sulit diperoleh. Informasi dapat diperoleh dengan mudah jika memiliki suatu sistem.

2. Luas dan Lengkap

Informasi mempunyai nilai yang lebih sempurna apabila mempunyai lingkup atau cakupan yang luas dan lengkap. Informasi sepotong dan tidak lengkap menjadi tidak bernilai, Karena tidak dapat digunakan secara baik.

3. Ketelitian

Begitu juga dengan ketelitian, informasi akan lebih sempurna apabila mempunyai ketelitian yang tinggi atau akurat. Informasi yang tidak akurat akan mengakibatkan kesalahan pengambilan keputusan.

4. Kecocokan

Informasi harus sesuai dengan kebutuhan informasi pengguna, sehingga informasi itu memiliki nilai karena bermanfaat.

5. Ketepatan Waktu

Sifat ini berhubungan dengan waktu yang dilalui lebih pendek dari pada siklus untuk mendapatkan informasi. Informasi penting dan bernilai menjadi tidak bernilai apabila terlambat diterima, karena tidak dapat dimanfaatkan dalam proses pengambilan keputusan.

6. Kejelasan

Informasi yang jelas akan meningkatkan kesempurnaan nilai informasi, kejelasan informasi dipengaruhi oleh bentuk dan format informasi.

7. Fleksibilitas / Keluwesan

Berkaitan dengan kegunaan informasi untuk berbagai pengambilan keputusan. Makin banyak keputusan yang diambil dari suatu informasi makin luwes informasi tersebut.

8. Dapat Dibuktikan

Berkaitan dengan tepat tidaknya informasi itu diuji kebenarannya oleh beberapa orang sehingga dapat memperoleh kesimpulan yang sama. Kebenaran informasi bergantung pada validitas data sumber yang diolah.

9. Tidak Ada Prasangka

Informasi semakin bernilai jika didalamnya tidak dimasukkan unsur opini, sebab dengan memasukkan unsur opini maka informasi bersifat bias.

10. Dapat Diukur

Pengukuran informasi umumnya dimaksudkan untuk mengukur dan melacak kembali validitas sumber data yang digunakan.

2.3.2.5. **Kualitas Informasi**

Informasi dapat dikatakan mempunyai sejumlah karakteristik berbeda-beda, yang digunakan menggambarkan kualitas tersebut. Perbedaan antara baik dan buruk informasi dapat diidentifikasi dengan mempertimbangkan apakah atau tidak informasi tersebut memiliki beberapa atau semua atribut dari kualitas informasi .

Menurut O'Brien dan Marakas (2006) melakukan pendekatan lebih terstruktur dan menggambarkan atribut dari kualitas informasi dapat dibagi menjadi tiga kategori dasar:

1. *Time* menggambarkan jangka waktu informasi yang ditawarkan dan frekuensi di mana informasi tersebut diterima.
 - a. *Timeliness*, informasi harus tersedia ketika dibutuhkan. Jika informasi disediakan terlalu dini, hal itu mungkin tidak berlaku lagi ketika dibutuhkan. Jika informasi disediakan terlalu lama, itu tidak akan ada gunanya.
 - b. *Currency*, informasi harus mencerminkan keadaan saat ini ketika disediakan. Sekaligus dapat melangkah lebih jauh dan menunjukkan bahwa informasi terkini juga harus mengindikasikan area atau keadaan untuk mengubah pada saat informasi tersebut digunakan
 - c. *Frequency*, selain tersedia ketika dibutuhkan, informasi juga harus tersedia sesering yang dibutuhkan. Ini biasanya berarti informasi harus diberikan pada jarak waktu secara teratur. Contohnya adalah beberapa organisasi mungkin memerlukan laporan penjualan mingguan sementara lainnya hanya perlu laporan bulanan.
 - d. *Time period*, informasi harus dapat melingkupi periode waktu yang benar. Perkiraan penjualan, contoh, mungkin termasuk informasi mengenai kinerja masa lalu, kinerja saat ini dan prediksi kinerja jadi penerima memiliki pandangan dari masa lalu, saat ini dan keadaan masa yang akan datang.
2. *Content* menggambarkan ruang lingkup dan isi dari informasi.
 - a. *Accuracy*, informasi yang berisi kesalahan hanya memiliki nilai terbatas pada suatu organisasi

- b. *Relevance*, informasi yang diberikan harus relevan untuk situasi tertentu dan harus memenuhi informasi yang dibutuhkan penerima.
 - c. *Completeness*, semua informasi yang diperlukan untuk memenuhi informasi yang dibutuhkan penerima harus tersedia. Informasi yang tidak lengkap dapat kompromi dengan atribut kualitas informasi lainnya, seperti ruang lingkup dan akurasi.
 - d. *Conciseness*, hanya informasi yang bersangkutan untuk informasi dibutuhkan penerima harus tersedia. Selain itu, informasi harus disediakan dalam bentuk yang paling kompleks. Contoh, jumlah penjualan biasanya disediakan dalam bentuk grafik atau tabel. Itu akan menjadi tidak biasa untuk mereka yang disediakan gambaran bagian dari text.
3. *Form* menggambarkan bagaimana informasi disajikan ke penerima.
- a. *Clarity*, informasi harus mempresentasikan dalam bentuk yang sesuai dimaksudkan penerima. Penerima harus dapat mencari item tertentu secara cepat dan harus dapat mengerti informasi tersebut dengan mudah.
 - b. *Detail*, informasi harus mengandung tingkat detail yang benar dalam rangka memenuhi informasi yang dibutuhkan penerima.
 - c. *Order*, informasi harus disediakan dalam urutan yang benar.
 - d. *Presentation*, informasi harus mempresentasikan dalam bentuk yang sesuai dimaksudkan penerima. Perbedaan metode dapat digunakan untuk membuat informasi lebih jelas dan mudah diperoleh oleh penerima.

- e. *Media*, informasi harus mempresentasikan penggunaan media yang benar. Informasi resmi, contoh, sering mempresentasikan dalam bentuk laporan yang di print, sedangkan presentasi mungkin menggunakan proyektor..

2.3.3. Konsep Sistem Informasi

Menurut pendapat Jogianto Hartono dalam bukunya yang berjudul *Analisis dan Desain Sistem Informasi: Pendekatan Terstruktur, Teori dan Praktik Aplikasi Bisnis* berpendapat bahwa Konsep Sistem Informasi : “Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan”.

Dari definisi-definisi tersebut dapat disimpulkan bahwa sistem informasi adalah suatu sistem yang dibutuhkan dalam mengolah transaksi-transaksi yang bersifat manajerial yang membutuhkan kombinasi antara prosedur kerja, informasi, manusia dan teknologi dalam pembuatan laporan-laporan.

2.3.4. Komponen Sistem Informasi

Menurut pendapat Jogianto Hartono dalam bukunya yang berjudul *Analisis dan Desain Sistem Informasi: Pendekatan Terstruktur, Teori dan Praktik Aplikasi Bisnis* berpendapat bahwa suatu sistem informasi memiliki komponen-komponen sebagai berikut:

1. Perangkat keras (hardware), mencakup berbagai peranti fisik seperti komputer dan printer.

2. Perangkat lunak (*software*) atau program, yaitu sekumpulan instruksi yang memungkinkan perangkat keras memproses data.
3. Prosedur, yaitu sekumpulan aturan yang dipakai untuk mewujudkan pemrosesan data dan pembangkitan keluaran yang dikehendaki.
4. Orang, yaitu semua pihak yang bertanggung jawab dalam pengembangan sistem informasi, pemrosesan dan penggunaan keluaran sistem informasi.
5. Basis data (*database*), yaitu sekumpulan tabel, hubungan dan lain-lain yang berkaitan dengan penyimpanan data.

Jaringan komputer dan komunikasi data, yaitu sistem penghubung yang memungkinkan sumber (*resource*) dipakai secara bersama atau diakses oleh sejumlah pemakai.

2.4. Software Requirement

Sebuah *requirements* dalam konteks rekayasa perangkat lunak adalah sebuah kemampuan yang harus dimiliki dari suatu *software* (Dorfman and Richard, 1990). Kemampuan ini dapat ditujukan untuk memecahkan suatu permasalahan ataupun diperlukan untuk memenuhi ketentuan-ketentuan tertentu (seperti standar tertentu, keputusan manajemen, ataupun alasan-alasan politis).

Secara luas, *software systems requirements engineering* (RE) adalah proses untuk menemukan suatu himpunan *requirements* yang tepat sehingga suatu perangkat lunak dapat memenuhi kegunaannya. Proses ini dilakukan dengan cara mengenali para *stakeholder* serta kebutuhan mereka serta mendokumentasikannya di dalam bentuk yang dapat digunakan untuk analisa, komunikasi, dan implementasi yang mengikutinya.

Terdapat lima aktivitas utama di dalam proses *requirements engineering* :

1. *Requirements elicitation*
2. *Requirements analysis and negotiation*
3. *Requirements documentation*
4. *Requirements validation*

2.4.1. Requirements Elicitation

Pada tahap ini dikumpulkan berbagai *requirements* dari para *stakeholder*. Seorang pelanggan mempunyai masalah yang dapat ditangani oleh solusi berbasis komputer. Tantangan ini ditanggapi oleh seorang pengembang. Di sinilah komunikasi dimulai antara pelanggan, pengembang, dan calon pengguna dari sistem yang akan dibuat.

Sejalan dengan proses RE secara keseluruhan, tujuan dari *requirements elicitation* adalah:

- a. Untuk mengetahui masalah apa saja yang perlu dipecahkan dan mengenali perbatasan-perbatasan sistem (*system boundaries*).
- b. Untuk mengenali siapa saja para *stakeholder*.
- c. Untuk mengenali tujuan dari sistem; yaitu sasaran-sasaran yang harus dicapainya.

Terdapat beberapa jenis teknik pengumpulan *requirements*, yaitu:

- a. *Traditional techniques* merupakan berbagai cara pengumpulan data. Cara-cara ini termasuk kuesioner, survey, wawancara, serta analisis dari berbagai dokumentasi yang ada seperti struktur organisasi, petunjuk pelaksanaan (*juklak*) serta manual-manual dari sistem yang sudah ada.

- b. *Group elicitation techniques* bertujuan untuk mengembangkan dan mendapatkan persetujuan *stakeholder*, sementara memanfaatkan dinamika kelompok untuk memperoleh pengertian yang lebih mendalam. Cara-cara ini termasuk *brainstorming* dan *focus group*, juga berbagai *workshop* RAD/JAD (*workshop* untuk membangun sebuah konsensus dengan menggunakan seorang fasilitator yang netral).
- c. *Prototyping techniques* membuat suatu implementasi parsial dari *software* yang akan dibangun untuk membantu para pengembang, pengguna, serta pelanggan untuk lebih mengerti berbagai *requirements* sistem. Digunakan untuk mendapatkan umpan-balik yang cepat dari para *stakeholder*, teknik ini juga dapat digabungkan dengan berbagai teknik yang lain, seperti misalnya digunakan di dalam sebuah acara *group elicitation* ataupun sebagai basis dari sebuah kuesioner.
- d. *Model-driven techniques* menempatkan suatu model khusus dari jenis informasi yang akan dikumpulkan untuk digunakan sebagai pedoman proses *elicitation*, termasuk di antaranya adalah *goal based methods*.
- e. *Cognitive techniques* termasuk serangkaian cara yang semulanya dikembangkan untuk *knowledge acquisition* untuk digunakan di *knowledge-based systems*. Teknik-teknik ini termasuk *protocol analysis* (di mana seorang ahli melakukan sebuah tugas sembari mengutarakan pikiran-pikirannya), *laddering* (menggunakan berbagai pemeriksaan untuk mendapatkan struktur dan isi dari pengetahuan *stakeholder*), *card sorting* (meminta para *stakeholder* untuk menyusun kartu-kartu secara berkelompok, di mana setiap kartu tertera nama sebuah *domain entity*), dan *repertory grids*

(membuat sebuah *attribute matrix* for entities di mana para *stakeholder* diminta untuk mengisi matriks tersebut).

- f. *Contextual techniques* muncul pada tahun 1990-an sebagai sebuah pilihan di luar *traditional* maupun *cognitive techniques*. Termasuk di antaranya penggunaan teknik etnografis seperti pengamatan terhadap para peserta. Juga termasuk *ethnomethodology* dan analisis percakapan, yang keduanya menggunakan analisis terinci untuk mengenali pola-pola dalam percakapan dan interaksi.

Dalam aktivitas *requirements elicitation*, ada baiknya untuk mengkategorikan berbagai *requirements* yang ditemukan. Suatu *requirements* dapat diklasifikasi sebagai *functional requirements*, *non-functional requirements*, maupun *constraints*. Sedangkan Kotonya (1998) mengatakan bahwa suatu *requirements* dapat diklasifikasikan menjadi *very general requirements*, *functional requirements*, *implementation requirements*, *performance requirements*, dan *usability requirements*.

2.4.2. **Requirements Analysis**

Model-model yang dihasilkan dalam tahap ini ditujukan untuk analisa terhadap berbagai *requirements* yang ada. Para *stakeholder* berunding untuk mendapatkan suatu himpunan *requirements* akhir yang akan digunakan untuk tahap pengembangan selanjutnya. Menurut Kotonyo (1998) setelah selesainya tahap idealnya ini akan berlaku:

- a. Berbagai *requirements* dari masing-masing *stakeholder* tidak bertentangan.
- b. Informasi di dalam semua *requirements* harus lengkap.
- c. Berbagai *requirements* yang ada harus selaras dengan anggaran yang dimiliki.

Walaupun dengan adanya batasan-batasan tersebut, seluruh *requirements* sebaiknya mudah diubah ataupun disesuaikan.

2.4.3. *Software Requirements Specification*

Tahap ini adalah penulisan dari *requirements document*, yang terkadang disebut dokumen *Software Requirements Specification (SRS)*, dokumen ini sebaiknya:

- a. Hanya menetapkan perilaku sistem sebagaimana terlihat dari luar
- b. Menetapkan batasan-batasan (*constraints*) yang diberikan kepada implementasinya.
- c. Mudah diubah.
- d. Berguna sebagai alat referensi untuk pemeliharaan sistem.
- e. Memuat gambaran akan siklus kehidupan sistem di masa yang akan datang.

Untuk meningkatkan *readability*, beberapa standar dokumentasi SRS telah dikembangkan. Namun menurut seorang ahli, serangkaian standar dan *template* apabila berdiri sendiri tidak dapat digunakan sebagai cara yang mandraguna untuk memberi struktur bagi sekumpulan *requirements*; tetapi struktur yang digunakan haruslah dikembangkan sendiri-sendiri tergantung dari masalah yang sedang ditangani. Masalah standarisasi notasi dan pendokumentasian *requirements* membuat pendekatan sistematis terhadap RE menjadi sulit. McDermid (1994) memberikan sebuah daftar praktis ciri-ciri yang diinginkan pada sebuah *requirements document*:

- a. *Unambiguous*. Idealnya, hanya ada satu interpretasi terhadap sebuah *requirements document*.

- b. *Complete*. Semua aspek yang bersangkutan haruslah dijelaskan secara lengkap di dalam *requirements document*.
- c. *Consistent*. Tidak ada pernyataan yang bertentangan dalam *requirements document*
- d. *Verifiable*. Setelah sebuah sistem diimplementasikan, sebaiknya dapat dipastikan bahwa sistem tersebut memenuhi *requirements* awal.
- e. *Validatable*. Suatu *requirements* sebaiknya dapat diperiksa oleh pelanggan untuk memastikan bahwa *requirements* tersebut memang memenuhi kebutuhannya.
- f. *Modifiable*. Perubahan sebaiknya mudah dilakukan dan efek dari perubahan ini terhadap bagian-bagian lain sebaiknya minimal.
- g. *Understandable*. Semua *stakeholder* sebaiknya dapat mengerti *requirements* seperti ditetapkan di dalam dokumen.
- h. *Testable*. Semua *requirements* sebaiknya cukup kuantitatif untuk digunakan sebagai titik tolak pengujian sistem.
- i. *Traceable*. Harus dimungkinkan adanya pengacuan (*reference*) antar berbagai bagian di dokumen *requirements* ataupun ke bagian-bagian lain dari proses pembuatan perangkat lunak.

2.4.4. **Requirements Validation**

Dalam tahap ini, dokumen dari tahap sebelumnya diperiksa agar memenuhi kriteria-kriteria sebagai berikut (Kotonyo, 1998):

- a. Lengkap.
- b. Konsisten.

- c. Tunduk pada keputusan-keputusan yang diambil pada tahap *requirements analysis*.

Apabila ada *requirements* yang tidak memenuhi kriteria-kriteria tersebut, mungkin ada baiknya bagi proses RE untuk kembali ke tahap-tahap sebelumnya. Beberapa contoh masalah *requirements* yang terungkap pada tahap validasi antara lain :

- a. Kurang/tidak cocok dengan bakuan-bakuan kualitas.
- b. Kata-kata yang digunakan kurang baik sehingga *requirements* menjadi ambigu.
- c. Berbagai kesalahan yang terdapat pada model-model baik – model sistem ataupun model permasalahan yang hendak dipecahkan.
- d. Pertentangan antar *requirements* yang tidak ditemukan pada tahap analisis.

2.5. Software Design

2.5.1. Context Diagram

Context Diagram merupakan langkah awal dari analisis struktur dan level teratas dari diagram arus data dan merupakan penggambaran sistem secara garis besar. Diagram Konteks menggambarkan hubungan aliran-aliran data ke dalam dan ke luar sistem atau entitas-entitas yang terletak di luar sistem (*output*) atau menerima data dari sistem tersebut (*input*),. Satu hal yang perlu diperhatikan, diagram konteks hanya menggunakan satu lingkungan proses yang mewakili proses dari semua sistem.

Context Diagram terdiri dari :

1. Entitas yaitu manusia atau organisasi dalam sistem yang berkomunikasi dengan sistem yang ada.

2. Aliran Data yaitu informasi yang masuk ke dalam sistem dan keluar dari sistem.
3. Lingkaran yang berisi sistem yang akan diuraikan di Data Flow Diagram (DFD)

2.5.2. *Data Flow Diagram*

Data Flow Diagram (DFD) adalah suatu bagian yang menggambarkan arus data dalam suatu perusahaan, yang digambarkan dengan sejumlah simbol tertentu untuk menunjukkan perpindahan data yang terjadi dalam proses suatu sistem bisnis (Kendall & Kendall, 2004).

Ada 4 macam simbol yang digunakan untuk menggambarkan arus data dalam DFD, yaitu :

1. Kesatuan Luar (*External Entity*)

Merupakan kesatuan (*external entity*) dilingkungan luar sistem dapat berupa orang, organisasi atau sistem lainya yang menerima input atau memberi input dari sistem. kesatuan luar digambarkan dalam bentuk kotak.

2. Arus Data (*Data Flow*)

Menuju arus dari data yang dapat berupa input bagi sistem disimbolkan dengan bentuk panah.

3. Proses (*Process*)

Kegiatan yang dilakukan oleh sistem dari arus data yang masuk untuk menghasilkan arus data keluaran, proses disimbolkan dengan bentuk lingkaran.

4. Data Simpanan (*Data Store*)

Data simpanan merupakan simpanan dari data yang dapat berupa suatu file atau database pada sistem komputer, simpanan data dapat disimbolkan dengan garis horizontal paralel yang ditutup salah satu ujungnya.

2.5.3. *Entitas Relationship Diagram*

Entitas Relationship Diagram (ERD) adalah “*Suatu penyajian data dengan menggunakan Entitas dan hubungannya*”. ERD dapat menggambarkan hubungan antar entity dengan jelas dan menggambarkan batasan jumlah entity dan partisipasi antar entity sehingga mudah dimengerti oleh pemakai. komponen-komponen ERD, yaitu :

1. Entitas (Entity)

Entity adalah sesuatu yang dapat dibedakan dalam dunia nyata, dimana informasi yang berkaitan dengannya dikumpulkan. Simbol untuk entity adalah persegi panjang.

2. Hubungan (Relationship)

Relationship adalah asosiasi yang terjadi antara satu atau lebih entity. Hubungan digambarkan dengan bentuk belah ketupat. Tiap belah ketupat diberi label kata kerja.

3. *Attribute*

Atribut adalah karakteristik dari entity atau relationship, yang menyediakan penjelasan detail tentang entity atau relationship tersebut. Simbol attribute adalah bentuk Oval atau elips.

4. Cardinality Ratio Cardinality Ratio adalah batasan yang menjelaskan jumlah keterhubungan satu entity dengan entity lainnya. Jenis Cardinality Ration antara lain:

a. One to one relationship (1 : 1)

Hubungan antara file pertama dengan file kedua adalah satu berbanding satu, atau dapat pula sebaliknya file kedua dengan file pertama adalah satu berbanding satu.

b. One to many relationship (1 : M / M : 1)

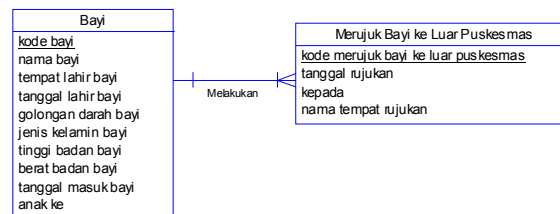
Hubungan antara file pertama dengan file kedua adalah berbanding banyak atau dapat pula dibalik banyak lawan satu..

c. Many to many relationship (M : N / N : M)

Hubungan antara file pertama dengan file kedua adalah banyak berbanding banyak atau dapat pula di balik file kedua dengan file pertama banyak berbanding banyak.

2.5.4. *Conceptual Data Model*

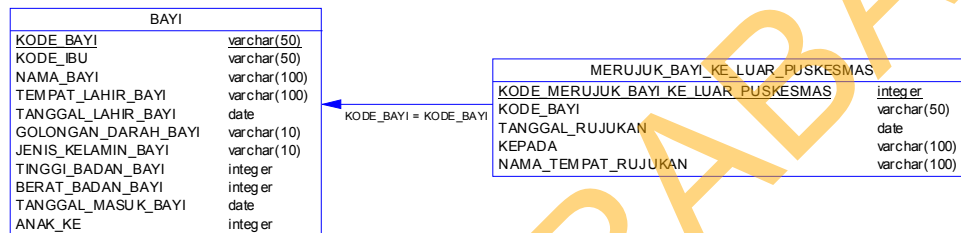
Conceptual Data Model (CDM) pada prinsipnya sama dengan pemodelan pada relational model, dimana pendekatan data yang disajikan menggunakan Entity-Relationship Diagram (ER-D). Pada level ini dapat disajikan secara general dan spesifik terhadap suatu basis data (database). Pada gambar 2 terlihat contoh relasi antara tabel guru dengan sekolah, dengan kardinalitas satu ke banyak (1:n/one to many). Dimana banyak guru mengajar di beberapa sekolah.



Gambar 2.3 Tampilan *Conceptual Data Model* Relasi 1:n Bayi dengan Merujuk

2.5.5. *Physical Data Model*

Physical Data Model (PDM) merupakan model fisik yang spesifik yang berdasarkan target database yang diinginkan sebelumnya (telah terdefiniskan). PDM merupakan hasil generate *Conceptual Data Model* yang telah ada. Dengan level PDM, dapat mengenerate (menurunkan) kembali kedalam sebuah bahasa definisi data



Gambar 2.4 Tampilan *Physical Data Model* dari tabel Bayi dan Merujuk

2.6. *Software Testing*

Menurut standart ANSI/IEEE 1059, *testing* adalah proses menganalisa suatu entitas *software* untuk mendeteksi perbedaan antara kondisi yang ada dengan kondisi yang diinginkan (*defects/error/bugs*) dan mengevaluasi fitur-fitur dari entitas *software*.

Menurut Romeo (2003), *testing software* adalah proses mengopersikan *software* dalam suatu kondisi yang dikendalikan untuk:

1. Verifikasi

Apakah telah berlaku sebagaimana yang di tetapkan (menurut spesifikasi).

2. Mendeteksi Error.

3. Validasi

Apakah spesifikasi yang ditetapkan telah memenuhi keinginan atau kebutuhan pengguna yang sebenarnya.

Menurut Romeo (2003), test case merupakan tes yang dilakukan berdasarkan pada suatu inisialisasi, masukan kondisi ataupun hasil yang telah ditentukan sebelumnya. Metode testing ini dibagi menjadi dua, yaitu:

2.6.1. *White Box Testing*

White box testing atau glass box testing atau clear box testing adalah suatu metode test case yang menggunakan struktur kendali dari desain prosedural. Metode desain test case ini dapat menjamin:

- a. Semua jalur (path) yang independen/terpisah dapat dites setidaknya sekali tes.
- b. Semua logika keputusan dapat dites dengan jalur yang salah atau jalur yang benar.
- c. Semua loop dapat dites terhadap batasannya dan ikatan operasional.
- d. Semua struktur internal data dapat dites untuk memastikan validasinya.

2.6.2. *Black box testing*

Black box testing atau behavioral testing atau specification-based testing, input/output testing atau functional testing dilakukan tanpa sepengetahuan detail struktur internal dari sistem atau komponen yang dites. Black box testing berfokus pada kebutuhan fungsional pada *software*, berdasarkan spesifikasi kebutuhan dari *software*.

Menggunakan black box testing, perancang *software* dapat menggunakan sekumpulan kondisi masukan yang dapat secara penuh memeriksa

keseluruhan kebutuhan fungsional pada suatu program. Kategori error dapat diketahui melalui black box testing, antara lain :

- a. Fungsi yang hilang atau tidak benar.
- b. Error dari antar muka.
- c. Error dari struktur data atau akses eksternal database.
- d. Error dari kinerja atau tingkah laku.
- e. Error dari inialisasi dan terminasi.

STIKOM SURABAYA