

BAB 2

KAJIAN PUSTAKA

2.1 Rekam Medis

2.1.1 Definisi Rekam Medis

Permenkes RI Nomor 269/Menkes/SK/III Tahun 2008 tentang Rekam Medis menyatakan bahwa, “Rekam Medis adalah berkas yang berisi catatan dan dokumen tentang identitas pasien, pemeriksaan, pengobatan, tindakan, dan pelayanan lain yang telah diberikan kepada pasien”.

Pedoman Penyelenggaraan dan Prosedur Rekam Medis Rumah Sakit di Indonesia Revisi II (Departemen Kesehatan RI, 2006:11) menyatakan bahwa:

“Rekam Medis adalah keterangan baik yang tertulis maupun yang terekam tentang identitas, anamnesa, penentuan fisik laboratorium, diagnose segala pelayanan, dan tindakan medis yang diberikan kepada pasien dan pengobatan baik yang di rawat inap, rawat jalan, maupun yang mendapatkan pelayanan gawat darurat”.

2.1.2 Tujuan Rekam Medis

Berdasarkan Pedoman Penyelenggaraan dan Prosedur Rekam Medis Rumah Sakit di Indonesia Revisi II (Departemen Kesehatan RI, 2006:12) tujuan rekam medis adalah menunjang tercapainya tertib administrasi dalam rangka upaya peningkatan pelayanan kesehatan di rumah sakit. Tanpa didukung suatu sistem pengelolaan rekam medis yang baik dan benar, mustahil tertib administrasi rumah sakit akan berhasil sebagaimana yang diharapkan.

2.1.3 Kegunaan Rekam Medis

Pedoman Penyelenggaraan dan Prosedur Rekam Medis Rumah Sakit di Indonesia Revisi II (Departemen Kesehatan RI, 2006:13) menyatakan kegunaan rekam medis dapat dilihat dari berbagai aspek yang dikenal dengan sebutan *ALFREDS* (*Administrative, Legal, Financial, Research, Education, Documentation, and Service*) sebagai berikut:

- a. *Administrative* (Aspek Administrasi)
Suatu berkas rekam medis mempunyai nilai administrasi, karena isinya menyangkut tindakan berdasarkan wewenang dan tanggung jawab tenaga medis dan paramedis dalam mencapai tujuan pelayanan kesehatan.
- b. *Legal* (Aspek Hukum)
Suatu berkas rekam medis mempunyai nilai hukum, karena isinya menyangkut masalah adanya jaminan kepastian hukum atas dasar keadilan dalam rangka usaha menegakkan hukum serta penyediaan bahan tanda bukti untuk penegakan hukum.
- c. *Financial* (Aspek Keuangan)
Suatu berkas rekam medis mempunyai nilai keuangan, karena isinya dapat dijadikan sebagai bahan untuk menetapkan biaya pembayaran layanan pada fasilitas pelayanan kesehatan. Tanpa adanya bukti catatan tindakan/pelayanan, maka pembayaran tidak dapat dipertanggungjawabkan. Data/informasi yang ada dapat digunakan sebagai aspek keuangan.
- d. *Research* (Aspek Penelitian)
Suatu berkas rekam medis mempunyai nilai penelitian, karena informasi yang dikandungnya dapat digunakan sebagai bahan penelitian dan pengembangan ilmu pengetahuan dibidang kesehatan.
- e. *Education* (Aspek Pendidikan)
Suatu berkas rekam medis mempunyai nilai penelitian, karena isinya menyangkut data/informasi tentang perkembangan kronologis dari kegiatan pelayanan rekam medis yang diberikan kepada pasien. Informasi tersebut dapat dipergunakan sebagai bahan/referensi pengajaran dibidang profesi pemakai.

f. *Documentation*(Aspek Dokumentasi)

Suatu berkas rekam medis mempunyai nilai dokumentasi, karena isinya menyangkut sumber ingatan yang harus didokumentasikan dan dipakai sebagai bahan pertanggungjawaban laporan rumah sakit.

g. *Service*(Aspek Medis)

Suatu berkas rekam medis mempunyai nilai medik, karena catatan tersebut dipergunakan sebagai bahan dasar untuk merencanakan pengobatan perawatan yang harus diberikan kepada seorang pasien.

2.1.4 Rekam Medis Elektronik

Rekam Medis Elektronik (RME) sebagai dokumen elektronik didukung oleh UU ITE Nomor 11 Tahun 2008 yang menyatakan “Informasi Elektronik dan/atau Dokumen Elektronik dan/atau hasil cetaknya merupakan alat bukti hukum yang sah”, sehingga menjadikan RME ini memiliki dasar hukum yang sah. Implementasi penggunaan RME di rumah sakit adalah adanya Sistem Informasi Manajemen Rumah Sakit (SIMRS), dimana pengelolaan data pasien dan rumah sakit saling terintegrasi.

2.2 *Assembling*

Mathar (2018:78) menyatakan bahwa, “*Assembling* adalah perakitan dokumen rekam medis dengan menganalisis kelengkapan berkas rekam medis (*review*). *Assembling* berkas rekam medis yaitu satu fungsi unit rekam medis yang berfungsi sebagai peneliti kelengkapan isi dan perakitan berkas rekam medis sebelum disimpan”.

Surat Keputusan Menteri Kesehatan RI No. 337/MENKES/SK/III/2007 tentang Standar Profesi Perkam Medis dan Informasi Kesehatan dalam kompetisi

ke 3 “Manajemen Rekam Medis dan Informasi Kesehatan” kompetensi perekam medis dan informasi kesehatan di Indonesia adalah mampu menyusun (*assembling*) rekam medis dengan baik dan benar berdasarkan ketentuan.

Health Information Management Physician Record Company Illinios(1992)

menyatakan bahwa:

“*Review* rekam medis merupakan prosedur analisa kuantitatif harus menegaskan laporan mana yang akan dilakukan, kapan dan keadaan yang bagaimana karena jika sewaktu-waktu ada pasien yang merasa telah di malpraktek pihak rumah sakit bisa menunjukkan BRM yang merupakan bukti tindakan apa saja yang dilakukan dan merupakan bukti hukum”.

Huffman (1994: 237) pada bagian *assembling* diketahui tipe ketidaklengkapan berkas rekam medis ada 2 (dua), yaitu:

1. *Incomplete Medical Record* (IMR)

Rekam medis dengan kekurangan yang spesifik yang masih dapat dilengkapi oleh pemberi pelayanan kesehatan.

2. *Delinquent Medical Record* (DMR)

Rekam medis yang masih tidak lengkap sesudah melewati batas waktu yang ditentukan.

Standart Nasional Akreditasi Rumah Sakit (SNARS) Edisi 1(2017: 366)

menyatakan bahwa:

“Dalam upaya perbaikan kinerja, rumah sakit secara teratur melakukan evaluasi atau *review* rekam medis. Setiap rumah sakit sudah menetapkan isi dan format rekam medis pasien dan mempunyai proses untuk melakukan assesmen terhadap isi dan kelengkapan berkas rekam medis. Proses tersebut merupakan bagian dari kegiatan peningkatan kinerja rumah sakit yang dilaksanakan secara berkala. *Review* rekam medis berdasar atas sampel yang mewakili PPA yang

memberikan pelayanan dan jenis pelayanan yang diberikan. Proses *review* melibatkan staf medis, keperawatan, serta PPA lainnya yang relevan dan mempunyai otoritas untuk mengisi rekam medis pasien. *Review* berfokus pada ketepatan waktu, kelengkapan, dapat terbaca, keabsahan, dan lain-lain dari rekam medis serta informasi klinis. Isi rekam medis yang dipersyaratkan oleh peraturan dan perundang-undangan dimasukkan ke dalam proses *review* rekam medis, hasil *review* dilaporkan secara berkala kepada Pimpinan RS”.

2.3 Aplikasi

“Aplikasi adalah program yang menentukan aktivitas pemrosesan informasi yang dibutuhkan untuk menyelesaikan tugas-tugas khusus dari pemakai komputer” (Indrajani, 2015: 4).

Menurut Sugiar (2014: 83) aplikasi adalah program yang dibuat untuk melaksanakan tugas tertentu yang dibutuhkan oleh pengguna komputer (*user*).

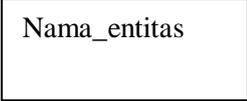
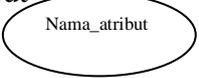
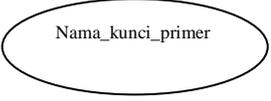
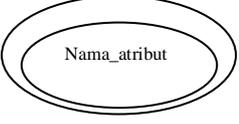
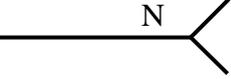
2.4 Entity Relationship Diagram (ERD)

Menurut Sukanto dan Shalahuddin (2014: 50), “*Entity Relationship Diagram (ERD)* adalah pemodelan awal basis data yang akan dikembangkan berdasarkan teori himpunan dalam bidang matematika untuk pemodelan basis data relasional”. Sehingga jika penyimpanan basis data menggunakan OODBMS maka perancangan basis data tidak perlu menggunakan ERD.

Menurut Sukanto dan Shalahuddin (2014: 50), ERD memiliki beberapa aliran notasi seperti notasi Chen (dikembangkan oleh Peter Chen), Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi lain. Beberapa aliran notasi dari ERD, yang paling banyak

digunakan adalah notasi dari Chen. Berikut adalah simbol-simbol yang digunakan pada ERD dengan notasi Chen:

Tabel 2.1 Simbol-simbol pada *Entity Relationship Diagram* (ERD)

Simbol	Keterangan
Entitas/ <i>Entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.
Atribut 	Atribut merupakan <i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas.
Atribut Kunci Primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama).
Atribut Multinilai/ <i>Multivalue</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.
Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja.
Asosiasi/ <i>Assosiation</i> 	Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki multiplicity kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan one to many menghubungkan entitas A dan entitas B

Sumber: Sukamto dan Shalahuddin (2014: 50)

2.5 *Database*

2.5.1 Pengertian *Database*

Menurut Indrajani(2015: 70), *Database* atau basis data adalah sebuah kumpulan data yang saling berhubungan secara logis dan merupakan sebuah penjelasan dari data tersebut, yang didesain untuk menemukan data yang dibutuhkan oleh sebuah organisasi.

Database atau basis data diartikan sebagai kumpulan dari tabel. Satu tabel merepresentasikan suatu entitas tertentu, dan entitas terdiri atas beberapa atribut (Heryanto, 2012: 1).

Kadir (2014: 218) menyatakan bahwa, “*Database* adalah suatu pengorganisasian sekumpulan data yang saling terkait sehingga memudahkan aktivitas untuk memperoleh informasi. *Database* dimaksudkan untuk mengatasi *problem* pada sistem yang memakai pendekatan berbasis berkas”.

2.5.2 Manfaat *Database*

Menurut Heryanto (2012: 5) salah satu manfaat *database* yang paling utama adalah untuk memudahkan dalam mengakses data. Kemudahan pengaksesan data ini sebagai implikasi dari keteraturan data yang merupakan syarat mutlak dari suatu *database* yang baik.

2.5.3 Elemen-Elemen *Database*

Menurut Kadir (2014: 62) elemen-elemen *database* relasional terdiri dari sebagai berikut:

1. Tabel
Tabel merupakan kumpulan informasi secara logis yang terkait dan diperlakukan sebagai unit. Setiap tabel selalu terdiri atas lajur vertikal yang biasa disebut dengan kolom atribut (*column/field*) dan lajur horizontal yang biasa disebut dengan baris data (*row/record*).
2. Kolom (*Field*)
Kolom merupakan atribut data yang ada pada suatu tabel pada *database*.
3. Baris (*Record*)
Baris adalah kejadian tunggal yang berisi data di dalam tabel. Suatu tabel terdiri dari baris (*record*) yang mengisi setiap kolom (*field*).
4. Kunci (*Key*)
 - a. *Primary Key*
Primary Key adalah suatu kolom yang memiliki nilai unik dan digunakan untuk mengidentifikasi setiap baris di dalam tabel. Karena sifatnya yang unik maka dapat digunakan sebagai acuan dalam menampilkan, mengubah dan menghapus baris data. Dengan adanya *primary key* juga dapat mencegah terjadinya duplikasi data karena sifatnya yang unik.
 - b. *Foreign Key*
Foreign Key merupakan nilai kolom pada suatu tabel yang berfungsi sebagai penghubung antara tabel dan dapat memberikan keterkaitan antara baris data pada satu tabel dengan baris data pada tabel lainnya.

2.6 Perangkat Lunak Pendukung

2.6.1 Microsoft Visual Basic 6.0

Subari dan Yuswanto (2008: 1) menyatakan bahwa “*Microsoft Visual Basic 6.0* adalah sebuah bahasa pemrograman yang digunakan untuk membuat pemrogram aplikasi berbasis orientasi objek atau *Object Oriented Program (OOP)*”. *Microsoft Visual Basic 6.0* selain disebut sebagai

bahasa pemrograman (*language program*), juga sering disebut sarana (*tool*) untuk menghasilkan program-program aplikasi berbasis *windows*.

Secara umum ada beberapa manfaat yang diperoleh dari pemakaian program Visual Basic, diantaranya:

1. Dipakai dalam membuat program aplikasi berbasis *windows*.
2. Dipakai dalam membuat obyek-obyek pembantu program, seperti fasilitas *Help*, *Control ActiveX*, aplikasi internet, dan sebagainya.
3. Digunakan untuk menguji program (*debugging*) dan menghasilkan program akhir EXE yang bersifat *executable*, atau dapat langsung dijalankan.

2.6.2 *MicrosoftStructure Query Language (SQL) Server 2000*

Yuswanto dan Subari (2005: 1) menyatakan bahwa, “SQL Server 2000 merupakan bahasa pemrograman yang dirancang khusus untuk komunikasi dengan database relasional guna mendukung aplikasi dengan arsitektur client-server”. Konsep penerapannya adalah database ditempatkan di komputer pusat yang disebut dengan server dan informasinya digunakan bersama-sama oleh user-user yang menjalankan aplikasi pada komputer lokal yang disebut dengan client.

Menurut PPS-STIKOM 001 PPS Modul Aplikasi 1 (2008: 8) SQL menyediakan perintah/*query* untuk melakukan tugas:

1. Menampilkan data (*Select*), menambah (*Insert*), mengubah (*Update*), menghapus (*Delete*) baris data di dalam tabel. Tugas-tugas tersebut disebut dengan *Data Manipulation Language (DML)*.

2. Membuat (*create*), mengganti (*alter*), membuang (*drop*) objek, menentukan integritas *database*. Tugas-tugas tersebut disebut dengan *Data Definition Language* (DDL).

2.6.3 *Crystal Report 8.5*

Hadi (2014: 13) menyatakan bahwa *crystal report* dirancang untuk membuat laporan yang dapat digunakan dengan bahasa pemrograman berbasis windows, seperti *Borland Delphi*, *Visual Basic*, *Visual C/C++*, dan *Visual Interdev*. Beberapa kelebihan yang dimiliki *Crystal Report* adalah:

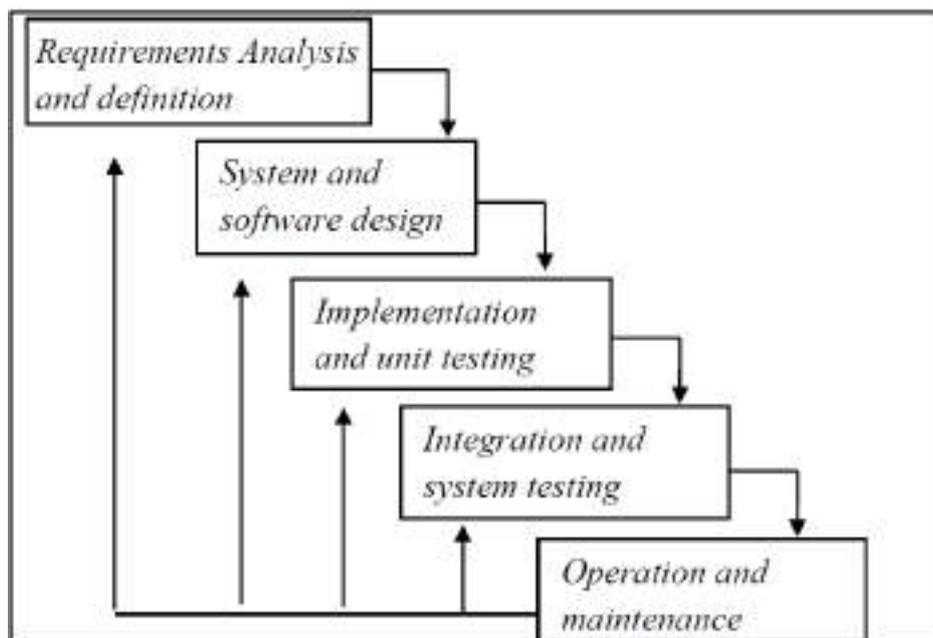
1. Dari segi pembuatan laporan tidak terlalu rumit yang memungkinkan *programmer* pemula sekalipun dapat membuat laporan yang sederhana tanpa melibatkan banyak kode pemrograman.
2. Integrasi dengan bahasa-bahasa pemrograman lain yang memungkinkan dapat digunakan oleh banyak *programmer* dengan masing-masing keahlian.
3. Fasilitas import hasil laporan yang mendukung format-format populer seperti Microsoft Word, Excel, Access, Adobe Acrobat Reader, HTML dan sebagainya.

2.7 Pengembangan Sistem Informasi

Sukanto dan Shalahuddin (2014: 26) menyatakan bahwa, “Metodologi Pengembangan Sistem Informasi yaitu *System Development Life Cycle* atau sering disingkat dengan SDLC adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang

digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya”.

Penelitian ini menggunakan metodologi pengembangan sistem “*System Development Life Cycle (SDLC)*” dengan model *waterfall*. Metode air terjun atau istilah lainnya *waterfall* model adalah model yang memacu tim pengembang untuk mengumpulkan dan menentukan apa yang seharusnya dilakukan sebelum sistem dikembangkan (Simarmata, 2010: 67). Model ini cocok untuk sistem yang mengedepankan kualitas dibandingkan biaya pengembangan atau waktu pengembangan. Kelebihan *waterfall* model adalah memberikan kemudahan serta kejelasan interpretasinya. Model ini terstruktur serta cocok diadaptasi untuk *management control*.



Gambar 2.1 Rekayasa Perangkat Lunak Sistem Informasi Menggunakan Model Air Terjun (*Waterfall*)

Sumber: Sukamto dan Shalahuddin (2014: 29)

Gambar 2.1 di atas didapatkan tahapan-tahapan dalam model *waterfall* sebagai berikut:

1. Analisis Kebutuhan

Langkah ini merupakan analisa terhadap kebutuhan sistem. Pengumpulan data dalam tahap ini bisa melakukan sebuah penelitian, wawancara atau studi literatur. Sistem analis menggali informasi sebanyak-banyaknya dari *user* sehingga tercipta sebuah sistem komputer yang bisa melakukan tugas-tugas yang diinginkan oleh *user* tersebut. Tahapan ini menghasilkan dokumen *user requirment* atau bisa dikatakan sebagai data yang berhubungan dengan keinginan *user* dalam pembuatan sistem. Dokumen ini lah yang menjadi acuan sistem analis untuk menerjemahkan ke dalam bahasa pemrogram.

2. Desain Sistem

Tahapan dimana dilakukan penuangan pikiran dan perancangan sistem terhadap solusi dari permasalahan yang ada dengan menggunakan perangkat pemodelan sistem seperti diagram alir data (*Data Flow Diagram*), diagram hubungan entitas (*Entity Relationship Diagram*) serta struktur dan bahasan data.

3. Penulisan Kode Program

Penulisan kode program atau *coding* merupakan penerjemahan *design* dalam bahasa yang bisa dikenali oleh komputer. Dilakukan oleh *programmer* yang akan meterjemahkan transaksi yang diminta oleh *user*. Tahapan ini lah yang merupakan tahapan secara nyata dalam mengerjakan

suatu sistem. Dalam artian penggunaan komputer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan *testing* terhadap sistem yang telah dibuat tadi. Tujuan *testing* adalah menemukan kesalahan-kesalahan terhadap sistem tersebut dan kemudian bisa diperbaiki.

4. Pengujian Program

Tahapan akhir dimana sistem yang baru diuji kemampuan dan keefektifannya sehingga didapatkan kekurangan dan kelemahan sistem yang kemudian dilakukan pengkajian ulang dan perbaikan terhadap aplikasi menjadi lebih baik dan sempurna.

5. Penerapan Program dan Pemeliharaan

Perangkat lunak yang sudah disampaikan kepada pelanggan pasti akan mengalami perubahan. Perubahan tersebut bisa karena mengalami kesalahan karena perangkat lunak harus menyesuaikan dengan lingkungan (peripheral atau sistem operasi baru) baru, atau karena pelanggan membutuhkan perkembangan fungsional.

2.8 Pengujian Sistem

Pengujian sebuah aplikasi dapat dilakukan dengan menggunakan pengujian *blackbox testing*. Menurut Pressman (2010) *Black-Box testing* berfokus pada persyaratan fungsional perangkat lunak yang memungkinkan *engineers* untuk memperoleh set kondisi *input* yang sepenuhnya akan melaksanakan persyaratan fungsional untuk sebuah program. *Black-Box testing* berusaha untuk menemukan kesalahan dalam kategori berikut:

1. Fungsi yang tidak benar atau fungsi yang hilang
2. Kesalahan antarmuka
3. Kesalahan dalam struktur data atau akses *database* eksternal
4. Kesalahan perilaku (*behavior*) atau kesalahan kinerja
5. Inisialisasi dan pemutusan kesalahan

Tes ini dirancang untuk menjawab beberapa pertanyaan-pertanyaan berikut ini:

- a. Bagaimana validitas fungsional diuji?
- b. Bagaimana perilaku dan kinerja sistem diuji?
- c. Apa kelas *input* akan membuat kasus uji yang baik?
- d. Apakah sistem *sensitive* terhadap nilai input tertentu?
- e. Bagaimana batas-batas kelas data yang terisolasi?
- f. Kecepatan dan volume data seperti apa yang dapat ditolerir sistem?
- g. Efek apakah yang akan menspesifikasikan kombinasi data dalam sistem operasi?

Tabel 2.2 Contoh Penerapan *Black Box Testing*

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Verifikasi Login	Black Box
	Verifikasi Password	
Pengolahan Data Pasien	Input data	Black Box
	Hapus data	
	Edit data	
Pengolahan Data Rekam Medis	Input data	Black Box
	Cari data	
Pendaftaran	Input data	Black Box
	Cari Data	
Petugas Kesehatan	Input Data	Black Box
	Hapus data	
	Edit data	
	Cari Data	
Poli	Input Data	Black Box
	Hapus data	
	Edit data	
Kartu Identitas Berobat	Cari Data	Black Box
	Input Data	
	Hapus data	
	Edit data	

Sumber: Nugroho dkk., 2015