

BAB 2

KAJIAN PUSTAKA

2.1 Rekam Medis Elektronik (RME)

Rekam medis adalah catatan tertulis yang mencantumkan nama pasien, informasi kontak, hasil tes, perawatan yang diterima, dan layanan lain yang diberikan. Rekam medis elektronik adalah yang diproduksi menggunakan sistem yang dibuat khusus untuk memelihara catatan medis. (Permenkes Nomor 24 Tahun 2022, 2022). Menurut (Permenkes Nomor 24 Tahun 2022, 2022) pengaturan rekam medis bertujuan untuk meningkatkan standar pelayanan kesehatan, Memberikan kejelasan hukum penyelenggaraan rekam medis dan administrasi. Menjamin keamanan, privasi, integritas, dan aksesibilitas data pasien, dan Mewujudkan administrasi dan organisasi rekam medis digital yang terintegrasi.

2.2 Database

Database berasal dari kata dasar *base* yang berarti pangkalan, gudang dan data berarti hasil mentah yang di dapatkan dilapangan yang sesuai dengan fakta yang belum diolah menjadi informasi, yang direkam dalam bentuk angka, huruf, simbol, teks, suara, dan gambar. Jadi, database adalah kumpulan pengelompokan data (arsip) yang terhubung yang dikoordinasikan di antara data tersebut sehingga dapat digunakan kembali dengan mudah (Ali, 2019). Menurut Ali, 2019 tujuan *database* adalah:

1. Kecepatan dan kemudahan (*speed*)
2. Efisiensi ruang penyimpanan (*Space*)
3. Keakuratan (*accuracy*)
4. Ketersediaan (*Availability*)

5. Kelengkapan (*Completeness*)
6. Keamanan (*security*)
7. Kebersamaan pemakaian (*sharability*) (Ali, 2019)

Komponen *database* menurut (Ali, 2019) adalah :

1. Perangkat Keras (*hardware*)
2. *Operating system*
3. Program aplikasi
4. *Database*
5. DBMS (*database management system*)
6. *User*

Tabel dalam *database* adalah tabel yang menyimpan data yang dikelompokkan di dalam bentuk baris dan kolom seperti layaknya lembar kerja. Setiap baris mewakili *record* dan setiap kolom adalah *atribut* dan *field*. Setiap *field* mengandung satu jenis informasi (Ali and Arifin, 2018). Tabel pada *database* yaitu:

1. Tabel Master

Tabel master merupakan tabel yang berdiri sendiri dan mewakili entitas tertentu. Tabel master pada akhirnya akan digunakan untuk melakukan operasi manipulasi data termasuk pembaruan, penyisipan, dan penghapusan. Contohnya tabel barang, mahasiswa, dosen.

2. Tabel Transaksi

Tabel transaksi adalah tabel yang independen atau bergantung pada tabel lain dan menampilkan hasil transaksi pada formulir transaksi. Ada dua kolom dalam tabel transaksi. Tabel ini dibagi menjadi dua yaitu:

a. Transaksi Header

Hanya satu entri, satu penulisan, dan satu penyimpanan yang dibuat untuk setiap tabel bidang. Dengan kata lain, ID hanya akan ditulis, diketik, atau diulang satu kali.

b. Transaksi Detail

Setiap tabel bidang dapat ditulis, dimasukkan, dan disimpan beberapa kali. Artinya, banyak transaksi yang bisa dilakukan dengan menggunakan nomor yang sama. (Ali and Arifin, 2018).

2.3 Entity Relationship Diagram (ER-Diagram)

Entity Relationship Diagram adalah representasi dari hubungan antara entitas dalam diagram. Semua kondisi (fakta) dari “dunia nyata” yang dipelajari pembaca diwakili oleh satu atau lebih atribut yang membentuk setiap entitas. Pembaca mencoba mengubah status "dunia nyata" ke dalam bentuk basis data menggunakan ER-Diagram (Ali, 2019). Dalam buku (Ali, 2019) yang berjudul “Penerapan Teknologi Basis Data Di Bidang Rekam Medis” simbol-simbol ER-Diagram yaitu:

1. Entitas

Entitas adalah orang/benda yang melambangkan sesuatu yang berbeda, apakah itu berbentuk objek atau tidak. Contohnya adalah sebuah mobil yang diperkirakan di depan rumah, seorang pegawai di perusahaan.

2. Atribut

Atribut adalah *header/field* yang dikenal sebagai atribut berfungsi untuk mengkarakterisasi fitur entitas. Contohnya adalah entitas mahasiswa seperti NIM, nama mahasiswa, agama mahasiswa dll.

3. Relasi

Relasi menunjukkan adanya hubungan antara sejumlah elemen yang berasal dari entitas berbeda.

4. Link

Link adalah garis penghubung antar simbol dalam ER-Diagram.

2.4 *Structure Query Language (SQL) Server 2000*

SQL adalah salah satu *software system* manajemen database yang populer di kalangan pemrograman *web*. Karena pembaca harus memasukkan query *SQL* (atau perintah *SQL*) ke dalam fungsi ini setiap kali digunakan, fungsi *SQL* dapat dianggap sebagai juru bahasa *queri*. Dengan kata lain, tanpa fungsi *SQL*, *SQL* tidak dapat dihasilkan (Ali and Arifin, 2018). Bahasa dalam *database* dibagi menjadi dua kelompok yaitu *DDL (data definition language)* dan *DML (data manipulation language)*. *DDL (data definition language)* adalah bahasa basis data yang digunakan untuk menggambarkan desain basis data secara keseluruhan. Perintah-perintah dalam *statemen DDL* yaitu *create* (membuat), *alter* (mengubah), *drop* (menghapus). *DML (data manipulation language)* berguna untuk mengambil dan mengubah data dari *database*. Perintah-perintah *DML* adalah *select*, *insert*, *update*, *delete* (Ali and Arifin, 2018).

2.5 Dreamweaver

Dreamweaver merupakan *Adobe System* membuat perangkat lunak editor *web* yang digunakan untuk membuat dan merancang fitur menarik untuk situs *web*. Paket *Adobe Creative Suite (CS)*, yang terdiri dari desain grafis, penyuntingan video, dan program pengembangan aplikasi *web*, berisi *Dreamweaver CS6*. Pada

tanggal 21 April 2012, Adobe Creative Suite 6 (CS6) telah tersedia. Fitur baru dari Dreamweaver versi sebelumnya termasuk *Fluid Grid Layout*, yang memudahkan pengelolaan beberapa halaman *web*, manajemen pengelola *font web*, yang memungkinkan pembaca menggunakan *font* yang disimpan di *server web*, dan Panel Gaya CSS ekstra, yang dapat digunakan untuk membuat grafik seperti bayangan pada elemen bentuk tertentu, gradien, sudut membulat, dan sebagainya. *Phone GAP* mempermudah konsumen untuk membuat aplikasi *smartphone* berkat *CSS Transitions* efek khusus pada elemen (Nurhadi, Indrayuni and Sinnun, 2015).

2.6 Black Box Testing

Menurut Shibab pada (Nugroho *et al.*, 2022) *Black box Testing* adalah pengujian dapat menentukan satu set kebutuhan masukan dan melakukan tes pada persyaratan fungsional program selama pengujian yang difokuskan pada persyaratan fungsional dari perangkat lunak. *Black Box* berfokus pada spesifikasi fungsional dari perangkat lunak, *Tester* dapat mendefinisikan kumpulan kondisi *input* dan melakukan pengetesan pada spesifikasi fungsional program (Febrina, 2018). Menurut (Rouf, 2012) pada penelitian (Alattaqwa *et al.*, 2022) metode *black box* adalah teknik yang memungkinkan pengembang perangkat lunak untuk mendapatkan satu set kondisi *input* yang sepenuhnya memanfaatkan semua kebutuhan fungsional untuk sebuah program untuk mengevaluasi apakah semua fungsi perangkat lunak telah beroperasi sesuai dengan kriteria fungsional yang telah ditetapkan (Alattaqwa *et al.*, 2022)

2.7 Peminjaman dan Pengembalian

Karena rekam medis memiliki begitu banyak kegunaan, maka wajar jika banyak yang tertarik untuk menggunakannya. Pihak yang merawat pasien secara

langsung (dokter, perawat, fisioterapis, dll) dan petugas rekam medis berhak meminjam rekam medis (RSUD H.Abdul Aziz Marabahan, 2019). Peminjaman dokumen rekam medis dilakukan dengan cara tertulis (RSUD H.Abdul Aziz Marabahan, 2019). Petugas yang sedang bertugas menerima rekam medis keluar dari penyimpanannya harus dicatat pada buku peminjaman dokumen rekam medis dan juga meminta keterangan lebih lanjut dokumen rekam medis yang dipinjamkan (RSUD H.Abdul Aziz Marabahan, 2019). Rekam medis yang tidak berada pada tempat penyimpanan rekam medis, kemungkinannya yaitu:

1. Rekam medis dikirim ke poliklinik yang bersangkutan karena pasiennya berobat.
2. Rekam medis di kirim ke ruang pasie dirawat karena pasien yang datang kembali untuk melakukan perawatan.
3. Dipinjam karena ada keperluan seperti:
 - a. Pembuatan ringkasan medis
 - b. Surat Keterangan Medis Pasien
 - c. Penelitian yang dilakukan
 - d. Hukum pengadilan

Batas waktu peminjaman rekam medis adalah 1 x 24 jam untuk alasan kontrol di poliklinik/rawat jalan, dan harus dikembalikan ke tempat penyimpanan dengan segera, dalam keadaan baik, dan utuh (RSUD H.Abdul Aziz Marabahan, 2019).

2.8 Metode *Waterfall*

Menurut pressman pada (Wahid, 2020) "*Linear Sequential Model*," juga dikenal sebagai teknik air terjun, juga dikenal sebagai pendekatan air terjun, adalah cara pengembangan perangkat lunak yang sistematis dan teratur yang dimulai dengan spesifikasi kebutuhan pengguna. kemudian beralih ke fase perencanaan, pemodelan, pembuatan, dan penerapan sebelum menyelesaikan dengan dukungan untuk semua perangkat lunak yang dibuat.

Metode *waterfall* pertama kali diperkenalkan oleh Winston Royce awalnya mengusulkan model air terjun pada tahun 1970-an, model inilah yang paling sering digunakan dalam rekayasa perangkat lunak. Saat ini, model *waterfall* merupakan model yang populer untuk pengembangan perangkat lunak. Teknik pertumbuhan ini bekerja dengan cara yang disengaja dan berurutan. Setiap tahap harus menunggu akhir dari tahap sebelumnya dan berjalan secara berurutan, hal itu dikenal sebagai air terjun. Model pengembangan ini mengikuti jalur linier dari tahap perencanaan, tahap pertama pengembangan sistem, hingga tahap pemeliharaan, langkah terakhir pengembangan sistem. Tahap berikutnya tidak dapat mengulang atau kembali ke tahap sebelumnya sebelum tahap sebelumnya selesai (Wahid, 2020).

Tahapan metode *waterfall* yaitu:

1. *Requirement*

Pengembang sistem perlu memahami perangkat lunak yang diharapkan pengguna serta kendala perangkat lunak pada saat ini dalam proses pengembangan. Survei langsung, percakapan, dan wawancara semuanya dapat digunakan untuk mengumpulkan informasi. Mengambil data yang dibutuhkan oleh pengguna, informasi dievaluasi.

2. *Design*

Pada langkah ini, pengembang mengembangkan desain sistem yang membantu dalam menentukan arsitektur sistem secara keseluruhan serta perangkat keras dan persyaratan sistem.

3. *Implementation*

Sistem saat ini sedang dibangun sebagai program terpisah, disebut sebagai unit, yang selanjutnya akan digabungkan. Pengujian unit adalah proses pengembangan dan pengujian fungsional setiap unit yang dikembangkan.

4. *Verification*

Pada tahap ini, sistem diuji dan diverifikasi untuk menentukan apakah sepenuhnya memenuhi persyaratan sistem atau hanya memenuhi sebagian. Pengujian dapat dibagi menjadi tiga kategori: pengujian penerimaan, pengujian sistem, dan pengujian unit. Pengujian unit dilakukan pada modul kode tertentu. Pengujian unit dilakukan untuk memeriksa apakah semua permintaan pelanggan telah dipenuhi.

5. *Maintenance*

Langkah terakhir metode air terjun adalah pemeliharaan. Perangkat lunak yang telah selesai digunakan dan dipelihara. Koreksi kesalahan yang terlewatkan pada tahap sebelumnya adalah bagian dari pemeliharaan (Wahid, 2020).

2.9 *Flowchart*

Flowchart atau sering disebut dengan diagram alir merupakan jenis diagram tertentu yang menunjukkan suatu algoritma atau langkah-langkah instruksi berurutan sistem. Pada dasarnya, *flowchart* digambarkan dengan menggunakan simbol-simbol (Rosaly and Prasetyo, 2019). Setiap simbol mewakili proses yang

berbeda. Sementara itu, koneksi garis digunakan untuk menggambarkan bagaimana menghubungkan satu proses ke proses berikutnya. Setiap rangkaian proses dapat didefinisikan lebih tepat dengan *flowchart*. Selain itu, menggunakan bagan alur ini memudahkan untuk memasukkan prosedur baru. Setelah membuat *flowchart*, tugas programmer adalah mengubah desain logis menjadi kode dalam berbagai bahasa pemrograman yang diterima (Rosaly and Prasetyo, 2019).

Menurut (Rosaly and Prasetyo, 2019) fungsi *flowchart* yaitu:

1. Merancang Proyek Baru

Langkah berikut yang harus diambil pembaca saat merancang proyek adalah memetakan proyek ke bagan alur. Ini dapat membantu pembaca dalam merencanakan urutan tindakan termasuk keputusan kelompok.

2. Mengelola Alur Kerja

Flowchart adalah alat yang paling penting untuk mengelola proses. Fungsi *flowchart* dalam membangun integritas proses dan kapasitasnya untuk menghasilkan keluaran berkualitas tinggi berdasarkan proses.

3. Memodelkan Proses Bisnis

Proses bisnis yang dimaksud mencakup sejumlah tugas, baik sederhana maupun kompleks, selain kaitannya dengan keuntungan. Tujuan dari pemodelan *flowchart* adalah untuk menghasilkan hasil yang dapat diandalkan dan konsisten.

4. Mendokumentasikan Setiap Proses

Dokumentasi proses penting untuk menyelesaikan sebuah proyek. Mempertimbangkan hal ini, diagram alir adalah alat yang luar biasa untuk

mencapai tujuan tersebut. menggunakan diagram alir lebih efektif daripada menggunakan narasi untuk menggambarkan setiap langkah, bukan memetakannya menjadi satu.

5. Mempresentasikan Algoritma

Perancang sistem awalnya memutuskan algoritma proyek menggunakan bahasa spesifikasi dan deskripsi sebelum menuangkan proyek ke dalam bentuk perangkat lunak. Bahasa spesifikasi yang digunakan untuk menggambarkan perilaku sistem disebut bahasa spesifikasi dan deskripsi. *Flowchart* dapat memenuhi persyaratan ini. Hal ini dimaksudkan agar sistem yang akan dibuat dapat dipetakan menggunakan salah satu dari sekian banyak simbol khas yang ditawarkan *flowchart*. Selain itu, diagram alir menyediakan sumber daya yang dapat digunakan untuk masalah.

6. Mengaudit Proses

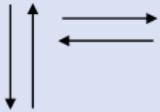
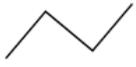
Secara umum, kerusakan yang terjadi pada setiap fase dapat ditemukan dengan menggunakan diagram alir. Dengan memecah setiap tahapan proses menjadi segmen-segmen yang lebih kecil dan kemudian memeriksa bagian mana yang tidak berfungsi atau perlu diperbaiki, diagram alir dapat membantu pembaca menemukan solusi untuk masalah tersebut.

Menurut (Rosaly and Prasetyo, 2019) simbol-simbol dalam *flowchart* yaitu:

1. Simbol Arus (*Flow Direction Symbols*)

Simbol yang masuk dalam kategori ini sering digunakan sebagai penghubung.

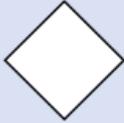
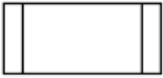
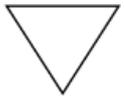
Beberapa simbol cocok dengan deskripsi ini., yaitu:

Simbol	Nama	Fungsi
	Flow Direction Symbol/ Connecting Line	Berfungsi untuk menghubungkan simbol yang satu dengan yang lainnya, menyatakan arus suatu proses
	Communication Link	Berfungsi untuk transmisi data dari satu lokasi ke lokasi lain
	Connector	Digunakan untuk menyatakan sambungan dari proses yang satu ke proses berikutnya di halaman yang sama
	Offline Connector	Digunakan untuk menyatakan sambungan dari proses yang satu ke proses berikutnya di halaman yang berbeda

Gambar 2.1 Simbol Arus *Flowchart*
 Sumber: (Rosaly and Prasetyo, 2019)

2. Simbol Proses (*Processing Symbols*)

Simbol proses digunakan untuk menggambarkan simbol yang terhubung ke sejumlah proses yang sedang berlangsung. Daftar simbol berikut muncul di bagian proses, yaitu:

Simbol	Nama	Fungsi
	Processing	Digunakan untuk menunjukkan pengolahan yang akan dilakukan dalam komputer
	Manual Operation	Digunakan untuk menunjukan pengolahan yang tidak dilakukan oleh komputer
	Decision	Digunakan untuk memilih proses yang akan dilakukan berdasarkan kondisi tertentu
	Predefined Process	Digunakan untuk mempersiapkan penyimpanan yang sedang/akan digunakan dengan memberikan harga awal
	Terminal	Digunakan untuk memulai atau mengakhiri program
	Offline Storage	Berfungsi untuk menunjukkan bahwa data akan disimpan ke media tertentu
	Manual Input Symbol	Digunakan untuk menginputkan data secara manual dengan keyboard

Gambar 2.2 Simbol Proses *Flowchart*
 Sumber: (Rosaly and Prasetyo, 2019)

3. Simbol I/O (*Input-Output*)

Bagian *input-output* berisi simbol-simbol yang berhubungan dengan *input* dan *output*. Beberapa simbol yang digunakan adalah sebagai berikut, yaitu:

Simbol	Nama	Fungsi
	Input / Output	Digunakan untuk menyatakan input dan output tanpa melihat jenisnya.
	Punched Card	Digunakan untuk menyatakan masukan dan keluaran yang berasal dari card
	Disk Storage	Digunakan untuk menyatakan masukan dan keluaran yang berasal dari disk
	Magnetic Tape	Digunakan untuk menyatakan masukan dan keluaran yang berasal dari pita magnetis
	Document	Digunakan untuk menyatakan masukan dan keluaran yang berasal dari dokumen
	Display	Digunakan untuk menyatakan keluaran melalui layar monitor

Gambar 2.3 Simbol *Input-Output Flowchart*
 Sumber: (Rosaly and Prasetyo, 2019)