BAB II
DASAR TEORI

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1. Tinjauan Pustaka

Penelitian sebelumnya dilakukan oleh Devista Andra, Akni Widiyastuti, dan Didi Susianto dengan judul "Pengembangan Sistem Informasi Rekam Medis Berbasis *Website* Pada Puskesmas Gisting" pada tahun 2019. Dalam penelitiannya penulis menggunakan bahasa pemograman PHP dan database MySQL. Sistem yang dibuat hanya bias diakses oleh admin saja[1].

Penelitian lainnya juga dilakukan oleh Oiyana Caesera Suibto dengan judul "Perancangan Sistem Informasi Rekam Medis Di RS Keluarga Husada Batam Berbasis *Website*". Dalam penelitiannya penulis menggunakan bahasa pemograman PHP dan database MySQL. Sistem yang dibuat bias diakses oleh admin dan juga pasien[2].

Penelitian berikutnya juga dilakukan oleh Aditya Pamungkas dengan judul "Rancang Bangun Sistem Informasi Pelayanan Kesehatan Untuk Rekam Medis rawat Jalan Pada Klinik Pratama Afiyah Medika Berbasis Desktop". Dalam penelitiannya penulis menggunakan bahasa pemograman java dan database MySQL. Sistem yang dibuat bisa diakses oleh admin, dokter, dan apoteker[3].

Penelitian ini penulis melakukan penelitian dengan judul " Sistem Informasi pencatatan Data Rekam Medis Berbasis *Website* Di Klinik Bidan Turachayati". Dalam penelitiannya penulis menggunakan bahasa pemograman PHP dan database MySQL. Sistem ini nantinya bisa diakses oleh dokter, admin maupun pasien.

Tabel 2. 1 Perbandingan Sistem

No	Penulis(tahun)	Judul	Tujuan	metode	proses	hasil
1.	Devista Andra , Akni Widiyastuti, Didi Susianto (2019) Dian Puspita Sari (2022)	Pengembangan Sistem Informasi Rekam Medis Berbasis *Website* Pada Puskesmas Gisting Sistem Informasi Pencatatan Data Rekam Medis Berbasis *Website* Di Klinik Bidan Turachayati	Dapat mempermudah petugas pada waktu pengolahan data rekam medis pasien dan pencarian dat rekam medis pasien saat datanya diperlukan Mampu membuat sistem informasi pencatatan data rekam medis berbasis *website* di	Metode Sebelumnya: *Extreme Programming.* *prototype*	Sistem ini meayani pemeriksaan secara *offline* saja dimana Petugas yang mencatat data rekam medis pasien didalam sebuah komputer dan disimpan dalam sebuah *database* Sistem ini melayani	Adanya sistem yang dapat membantu admin mengelola data pasien dan rekam medis pasien Adanya sistem yang membantu admin dalam mengelola data pasien , membantu dokter dalam mengelola data rekam medis pasien, serta dapat membantu pasien dalam melakukan konsultasi dan pemeriksaan secara online dengan notifikasi menggunakan *email gateway.*

			Klinik Bidan Turachayati yang Diharapkan dapat membantu dokter mengelola data rekam medis pasien yang datang di Klinik Bidan Turachayatiserta membantu admin pada saat pencarian data rekam medis pasien saat datanya diperlukan Dan mempermudah pasien dalam melakukan konsultasi dan		pemeriksaan secara *online* danjuga offline, dimana nanti data rekam medisnya akan disimpan oleh dokter dalam sebuah database. Sistem ini juga melayanikonsultasi pasien secara *online.*	

			pemeriksaan *online*			
2.	Oiyana Caesera Suibto(2019) Dian Puspita Sari (2022)	Perancangan Sistem Informasi Rekam Medis Di Rs Keluarga Husada Batam Berbasis *Website* Sistem Informasi Pencatatan Data Rekam Medis Berbasis *Website* Di Klinik Bidan Turachayati	Dapat membangun sistem informasi rekam medis yang dapat mengelola data riwayat pasien dan laporan bulanan Mampu membuat sistem informasi pencatatan data rekam medis berbasis *website* di Klinik Bidan	*Waterfall.* *prototype*	Sistem ini melayani pemeriksaan secara *offline* dimana Petugas yang mencatat data rekam medis pasien didalam sebuah komputer dan disimpan dalam sebuah database. Sistem ini juga dapat	Adanya sistem yang membantu admin dalam mengelola rekam medis dan laporan bulanan pasien rawat inap Adanya sistem yang membantu admin dalam mengelola data pasien , membantu dokter dalam mengelola data rekam medis pasien, serta dapat membantu pasien dalam melakukan konsultasi dan pemeriksaan secara online dengan notifikasi

			Turachayati yang Diharapkan dapat membantu dokter mengelola data rekam medis pasien yang datang di Klinik Bidan Turachayatiser ta membantu admin pada saat pencarian data rekam medis pasien saat datanya diperlukan Dan mempermudah pasien dalam melakukan konsultasi dan pemeriksaan		mengelola data laporan bulanan pasien rawat inap. Sistem ini melayani pemeriksaan secara online danjuga *offline,* dimana nanti data rekam medisnya akan disimpan oleh dokter dalam sebuah database. Sistem ini juga melayani konsultasi	menggunakan *email gateway.*

			online		pasien secara *online.*	
3.	Aditya Pamungkas (2020) Dian Puspita Sari (2022)	Rancang Bangun Sistem Informasi Pelayanan Kesehatan Untuk Rekam Medis Rawat Jalan Pada Klinik Pratama Afiyah Medika Berbasis Desktop Sistem Informasi Pencatatan Data Rekam Medis Berbasis *Website* Di Klinik Bidan Turachayati	Dapat membangun sistem informasi klinik yang diharapkan dapat meningkatkan efisiensi dalam melakukan kegiatan pelayanan kesehatan serta mempermudah dalam pembuatan laporan secara cepat dan akurat	*Waterfall.* *prototype*	Sistem ini melayani pemeriksaan secara *offline* dimana Petugas yang mencatat data rekam medis pasien didalam sebuah komputer dan disimpan dalam sebuah database. Sistem ini megelola data obat	Adanya sistem pelayanan Kesehatan untuk rekam medis klinik yang dapat mempermudah pengelolaan data seperti data pasien dan rekam medis pasien. Adanya sistem yang membantu admin dalam mengelola data pasien , membantu dokter dalam mengelola data rekam medis pasien, serta dapat membantu pasien dalam melakukan konsultasi dan pemeriksaan secara online dengan notifikasi

			mampu membuat sistem informasi pencatatan data rekam medis berbasis *website* di Klinik Bidan Turachayati yang Diharapkan dapat membantu dokter mengelola data rekam medis pasien yang datang di Klinik Bidan Turachayatiser ta membantu admin pada saat pencarian		oleh apoteker serta dapat mengelola data laporan bulanan pasien dan rekam medis pasien. Sistem ini melayani pemeriksaan secara *online* danjuga *offline,* dimana nanti data rekam medisnya akan disimpan oleh dokter dalam sebuah *database*.	menggunakan *email gateway.*

			data rekam medis pasien saat datanya diperlukan Dan mempermudah pasien dalam melakukan konsultasi dan pemeriksaan *online*		Sistem ini juga melayani konsultasi pasien secara *online.*	

2.2. Landasan Teori

2.2.1. Sistem Informasi

Sistem Informasi merupakan sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu meyajikan informasi[4].

Secara garis besar sistem Informasi merupakan alat yang dibuat manusia untuk memudahkan proses pengolahan data atau pemrosesan data. Sistem informasi memiliki beberapa komponen yang berfungsi mendukung jalannya sistem [4]. Berikut adalah beberapa komponen dalam sistem informasi:

a. *Hardware* dan *software* yang berfungsi sebagai mesin
b. *People* dan *procedures* yang merupakan manusia dan tatacara menggunakan mesin
c. Data merupakan jembatan penghubung antara manusia dan mesin agar terjadi suatu proses pengolahan data

2.2.2. *Website*

a. Pengertian

Website adalah fasilitas hiperteks untuk menampilkan data berupa teks, gambar, suara, animasi dan data multimedia lainnya yang diantara data tersebut saling berhubungan satu sama lain[5].

b. Jenis-jenis *website*

1) Web Statis

Web statis merupakan *web* yang berisi atau menampilkan informasi-informasi yang bersifat tetap.Pengunjung tidak dapat berinteraksi seperti mengisi form, berkomentar atau semacamnya. Contoh dari *web statis* ini adalah *web* tentang profil sebuah perusahaan.

2) Web Dinamis

Web dinamis adalah *web* yang menampilkan informasi serta dapat berinteraksi dengan *user* yang sifatnya *dinamis*. Contoh dari

web dinamis yaitu *web* tentang toko *online,* portal berita, atau jeraring social seperti facebook dan yang lainnya

2.2.3. Rekam Medis

Menurut UU nomor 29 tahun 2004 tentang praktik kedokteran, dijelaskab bahwa setiap dokter atau dokter gigi dalam menjalankan praktik wajib membuat rekam medis.

Pengertian rekam mdis juga telah dijelaskan pada Pasal 46 Ayat (1) dalam UU praktik kedokteran, bahwa rekam medis adalah berkas yang berisi catatan dokumen tentang identitas pasien, pemeriksaan, pengobatan, tindakan, dan pelayanan lain yang telah diberikan kepada pasien. Adapun kegunaan dari rekam medis yaitu[6]:

a. Sebagai alat komunikasi antara dokter dengan tenaga kesehatan lainnya yang ikut ambil bagian dalam memberikan pelayanan Kesehatan
b. Sebagai dasar untuk merencanakan pengobatan/perawatan yang harus diberikan kepada seorang pasien
c. Sebagai bukti tertulis atas segala tindakan pelayanan , perkembangan penyakit dan pengobatan selama pasien berkunjung/dirawat di Rumah sakit

2.2.4. PHP *Mailer*

PHP mailer merupakan fungsi yang digunakan untuk megirim email. Fungsi ini awalnya hanya dapat menjalankan fungsinya sesama *localhost*, tetapi dapat ditambahkan kemampuan PHP agar dapat menjalan fungsinya untuk sesama antar *server*. PHP *mailer* dapat menjalankan fungsinya sebagai pengirim email jika ditambahkan dengan *Simple Mail Tranfer Protocol* (SMTP). SMTP adalah sauatu protokol yang diperlukan untuk mengirim dan menerima *email*. Karena itu dibutuhkan SMTP sebagai layanan untuk mengirim *email*. Layanan tersebut dapat digunakan untuk keperluan seperti memverifikasi *email*[7].

2.2.5. Rekayasa *Web*

Rekayasa *web* merupakan sebuah aplikasi yang menggunakan pendekatan sistematis, disiplin, dan terukur untuk mengembangan, operasi, dan pemeliharaan aplikasi berbasis *web*. Rekayasa *web* adalah subdisiplin dari rekayasa perangkat lunak yang membantu menyediakan

metodologi untuk merancang, mengembangkan, memelihara, dan melibatkan aplikasi web[8].

2.2.6. Rekayasa Perangkat Lunak

Rekayasa perangkat Lunak merupakan pembangunan yang menggunakan prinsip atau konsep rekayasa dengan tujuan menghasilkan perangkat lunak yang bernilai ekonomi yang dipercaya dan bekerja secara efisien menggunakan mesin[9].

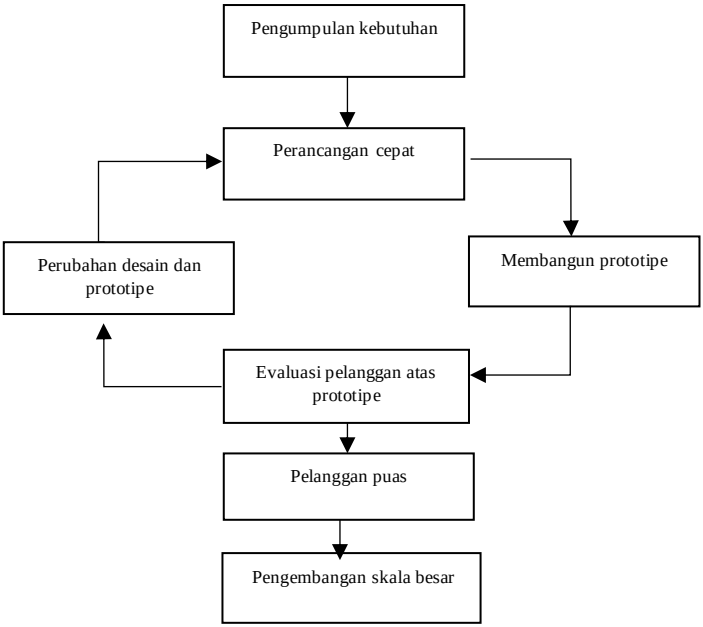
2.2.7. Metode Pengujian Sistem

Metode pengujian sistem yang digunakan adalag metode *black box testing*. *Black box testing* merupakan metode pengjuian sistem perangkat lunak dari segi spesifikasifungsional tanpa menguji desain atau kode program. Pengujin dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai denggan spesifikasi yang dibutuhkan[9].

Metode *blackbox testing* merupakan salah satu metode yang mudah digunakan karena hanya memerlukan batas bawah dan batas atas dari data yang diharapkan, Estimasi banyaknya data uji dapat dihitung melalui banyaknya field data entri yang akan diuji, aturan entri yang harus dipenuhi serta kasus batas atas dan batas bawah yang memenuhi. Dan dengan metode ini dapat diketahui jika fungsionalitas masih dapat menerima masukan data yang tidak diharapkan maka menyebabkan data yang disimpan kurang valid. Solusi praktis peningkatan akurasi perlu dilakukan segera guna memperbaiki celah *error* yang telah ditemukan[10].

2.2.8. *Prototype*

Prototype merupakan sebuah metode pengembangan sistem yang mengedepankan kebutuhan dan kemauman klien. *Prototype* tersebut akan dievaluasi oleh klien sehingga dapat menyaring kebutuhan perancangan sistem yang diinginkan. Berikut adalah langkah-langkah metode *prototype*[11]*:*



Gambar 2. 1 Alur Metode *Prototype*

Penjelasan tahapan *prototype*

1) Pengumpulan kebutuhan

   Bagian ini merupakan proses engumpulan informasi-informasi yang dibutuhkan sehingga nantinya sistem yang dibuat sesuai dengan kebutuhan *user*.

2) Melakukan Perancangan
   Setelah mengumpulkan informasi tentang kebutuhan, selanjutnya adalah merancang desain sebuah sistem berdasarkan informasi yang diperoleh.
3) Membangun Sebuah *Prototype*
   Membangun *prototype* dimaksudkan agar user dapat melihat gambaran secara utut sistem yang akan dibuat .
4) Evaluasi Pelanggan Atas *Prototype*

Setelah *Prototype* selesai dibuat maka prototype tersebut akan di evaluasi oleh *user* .

5) Perubahan Rancangan *Prototype*
   Jika *prototype* yang telah dibangun belum memenuhi kebutuhan user atau tidak sesuai dengan apa yang diinginkan user maka rancangan prototype tersebut akan diubah.
6) Apalabila *user* puas dengan rancangan *prototype* tersebut maka pengembangan akan dilanjutkan ke skala yang lebih besar.

2.2.9. Pemograman Berorientasi Objek

Pemograman berorientasi objek adalah suatu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek yang berisi data dan operasi yang diberlakukan terhadapnya. Bahasa pemograman yang mendukung pemograman berorientasi onjek yaitu antaranya *smalltalk, iffel*, C++, PHP, dan juga Java[9].

Pemogramanan berorientasi objek banyak dipilih karena Pemogramanan berorientasi objek mempunyai kelebihan yaitu multiplatform, adalah sistem operasi yang dapat di jalankan secara bersamaan, dan juga menghemat pemakaian fungsi karena semua fungsi ditulis dalam satu file class. Berikut ini aalah beberapa konsep dasar yang harus dipahami tentang metodologi berorientasi objek[9]:

a. Kelas (*Class*)
   Kelas merupakan berkas atau file yang berisi kode program, dimana kode program tersebut mrnampung isi program yang akan dijalankan.
b. Objek (*Object*)
   Secara teknis objek merupakan sebuah elemen pada saat runtime yang akan diciptakan, dimanipulasi, dan dihancurkan saat eksekusi sehingga objek akan ada pada saat program dieksekusi, jika masih dalam bentuk kode disebut sebagai kelas.
c. Metode (*Method*)
   Metode merupakan sebuah metode atau fungsi untuk memanipulasi objek dalam kelas. Metode didalam kelas boleh lebih dari satu. Contoh metode bias berupa read, write, move atau lainnya
d. Atribut (*Attribute*)

Atribut merupakan variable global dalam sebuah kelas. Atribut bias berupa nilai atau elemen-elemen data yang dimiliki objek dalam kelas. Atribut dimiliki secara individual oleh sebuah objek, misalnya nama, berat, dan sebagainya.

e. Abstraksi (*abstraction*)
Prinsip untuk mempresentasikan dunia nyata yang kompleks menjadi satu bentuk model yang sederhana dengan mengabaikan aspek-aspek lain yang tidak sesuai dengan permasalahan.

f. Enkapsulasi (*Encapsulation*)
Pembungkusan attribute data dan layanan (opersi-operasi) yang dipunyai objek untuk menyembunyikan implementasi dan objek sehingg objek lain tidak mengetahui cara kerjanya

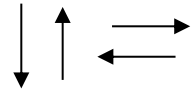
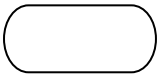
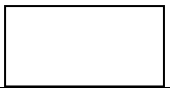
g. Pewarisan (*Inheritance*)
Mekanisme yang memungkinkan suatu objek mewarisi sebagian atas seluruh devinisi dan objek lain sebagai bagian dari dirinya.

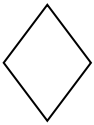
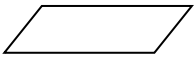
2.2.10. Struktur Data

1) Flowchart

Flowchart merupakan bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah. *Flowchart* merupakan cara penyajian dari suatu algoritma[4].

Tabel 2. 2 Simbol *Flowchart*

No	Simbol	Deskripsi
1.	 Flow Direction Simbol	Merupakan sebuah simbol untuk menyatakan jalannya arus suatu proses.
2.	 Terminator Simbol	Merupakan simbol untuk menyatakan permulaan atau akhir suatu program.
3.		Merupakan simbol Untuk menyatakan sambungan dari suatu proses ke proses lainnya

	Processing Simbol	dalam halaman atau lembar yang berbeda.
4.	Manual Operation Simbol	Merupakan simbol untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh computer.
5.	 Decision Simbol	Merupakan simbol untuk menunjukan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, ya/tidak.
6.	 Input-Output Simbol	Merupakan simbol untuk menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya
7.	Document Simbol	Merupakan simbol untuk mencetak laporan ke printer

2) UML

Unified Modeling Language atau biasa *disebut UML merupakan* Bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks *pendukung*. *UML meliputi use case diagram, sequence diagram* dan *class diagram*[9].

a. Use Case Diagram

Use case digram merupakan pemodelan untuk kelakuan sistem informasi yang akan dibuat. *Use case* mendiskripsikan sebuah interaksi antara satu atau lebih actor dengan sistem informasi yang akan dibuat .

Berdasarkan deskripsi diatas *use case* merupakan pemodelan yang menggambarkan interaksi aktor dengan sistem. Berikut adalah beberapa simbol *user case diagram:*

Tabel 2. 3 Simbol *Use Case Diagram*

No	Simbol	Nama simbol dan keterangan
1.		*Use case*, merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau faktor. *Use case* biasanya dinyatakan dengan kata kerja.
2.		Aktor, merupakan orang yang merinteraksi dengan sistem informasi yang akan dibuat.
3.		*Collacboration*, interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
4.		*Note*, Elemen fisik yang eksis saat aplikasidijalankan dan mencerminkan suatu sumber daya komputasi.

5.		Sistem, menfesifikasikan paket yang menampilkan sistem secara terbatas.
6.		*Assosiation*, komunikai atau penghubung antara actor dan use case.
7.		*Generalization*, Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah *use case* .
8.		Include, relasi *use case* tambahan ke sebuah *use case*, dimana *use case* yang ditambahkan memerlukan use case lainuntuk menjalankan fungsinya.
9.		*Dependency*, hubungan dimana perubahan yang terjadi pada suatu elemen mandiri akan mempengaruhi elemen yang bergantung pada elemen yang tidak mandiri.
10		*Extend*, relasi *use case* tambahan ke sebuah *use case* dimana *use case* yang ditambahkan dapat berdiri

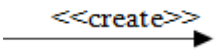
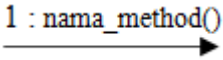
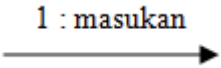
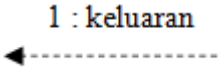
		sendiri walau tanpa *use case* tambahan itu.

b. Sequence Diagram

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup *objek* . Banyaknya *diagram sequence* yang harus Digambar adalah minimal sebanyak pendefisian use case yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada *diagram sequence* sehingga semakin banyak *use case* yang didefinisikan maka *diagram sequence* yang dibuat juga semakin banyak. Berikut adalah beberapa simbol *user sequence diagram*

Tabel 2. 4 Simbol *Sequence Diagram*

No	Simbol	Nana simbol dan keterangan
1.		Aktor, orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem infornasi yang akan dibuat itu sendiri.
2.		*Lifeline*, menyatakan kehidupan suatu objek.
3.		*Objek*, menyetakan *objek* yang berinteraksi dengan pesan.

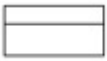
4.		Waktu aktif, menyatakan *objek* dalam keadaan aktif dan berinteraksi.
5.		Pesan tipe *create*, menyatakan suatu *objek* membuat *objek* yang lain, arah panah mengarah pada *objek* yang akan dibuat.
6.		Pesan tipe *call*, menyatakan suatu objek memanggil oprasi atau metode yang ada pada *objek* lain atau dirinya sendiri.
7.		Pesan tipe *send*, menyatakan bahwa suattu *objek* mengirimkan data atau masukan atau informasi ke *objek* lainnya, arah panah mengarah ke *objek* yang dikirim.
8.		Pesan tipe *return*, menyatakan bahwa suatu *objek* yang tlah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke *objek* tertentu, arah panah mengarah pada *objek* yang menerima kembalian.

c. Class Diagram

Class diagram menggambarkan sruktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Diagram kelas dibuat agar pembuat program menbuat kelas-kelas sesuai

rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron.

Tabel 2. 5 Simbol *Class Diagram*

No	Simbol	Nama simbol dan keterangan
1.	Association	Merupakan relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan *multiplicity.*
2.	 Class	Merupakan kelas pada struktur sistem.
3.	Dependency	Merupakan relasi antar kelas dengan makna kebergantungan antar kelas.
4.	Directed association	Merupakan relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan *multiplicity.*
5.	generalization	Merupakan relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus).
6.		Merupakan relasi antar kelas dengan makna semua bagian (*whole-part*).

	Aggregation	

2.2.11. Basis Data

Basis data merupakan sistem terkomputerisai yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan.

DBMS *(Database Management System)* atau sering disebut sebagai sistem manajemen basis data dalah suatu sistem aplikasi yang digunakan untuk menyimpan, mengelola dan menampilkan data.

SQL *(Structured Query Language)* adalah Bahasa yang dihunakan untuk mengelola data pad*a RDBMS*[9].

Berikut adalah beberapa contoh pengaksesan data *pada DBMS* dengan *SQL*

1) Memasukan data
   Insert digunakan untuk memasukan data dalam sebuah tabel.

```
INSERT into nama Tabel (field1, field2, field3,)
VALUES ('nilai 1','nilai 2','nilai 3'.)
```

2) Mengubah Data
   Update digunakan untuk mengubah data yang ada pada tabel.

```
UPDATE nama Tabel set field1='nilai baru field1', field2='nilai baru field2', field3='nilai baru field3',... [kondisi where]
```

3) Menghapus data
   Delete digunakan untuk meghapus data yang ada pada tabel.

```
DELETE from nama Tabel [kondisi where]
```

4) Menampilkan

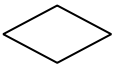
Select digunakan untuk memanggil atau menampilkan data yang ada pada tabel.

```
SELECT *|field1,field2,field3,... from nama_tabel [kondisi]
```

2.2.12. ERD

ERD*(Entity Relationship Diagram*) merupakan suatu pemodelan awal basis datarelasional. ERD memliki beberapa aliran seperti notasi Chen (dikembangkan oleh Peter Chen), Barker (dikembangkan oleh Richard Baeker, Ian Palmer, Harry Ellis), notasi Crow's Foot, dan beberapa notasi lain. Namun yang banyak digunakan adalah notasi dari Chen. Beriut adalah symbol-simbol yang digunakan pada ERD dengan notasi Chen[9].

Tabel 2. 6 *Simbol ERD*

No	Simbol	Deskripsi
1.	Entity	*Entity* atau entitas merupakan data inti yang akan disimpan. Entitas merupakan bakal tabel pada basis data yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi computer.
2.	Atribut	*Atribut* merupaan field atau kolom data yang butuh disimpan dalam suatu entitas.
3.	Relasi 	Relasi menghubungkan antar entitas, biasanya diawali dengan kata kerja.
4.	Alur	Garis sebagai penghubung antara himpunan, relasi, dan himpunan entitas dengan atributnya.

ERD memiliki hubungan relasi atau biasa disebut kardinalitas. Kardinalitas menjelaskan jumlah keterhubungan satu *entity* dengan *entity* lainnya. Berikut adalah macam-macam kardinalitas.

Tabel 2. 7 *Simbol Kardinalitas ERD*

Simbol	Deskripsi
Binary (Relasi one to one)	*Binery* menunjukan hubungan satu relasi menghubungkan dua buah entitas.
Ternary (*Relasi one to many)*	*Ternary* menunjukan hubungan satu relasi menghubungkan tiga buah entitas.
N-ary (Relasi many to many)	*N-ary* menunjukan hubungan satu relasi menghubungkan banyak entitas.