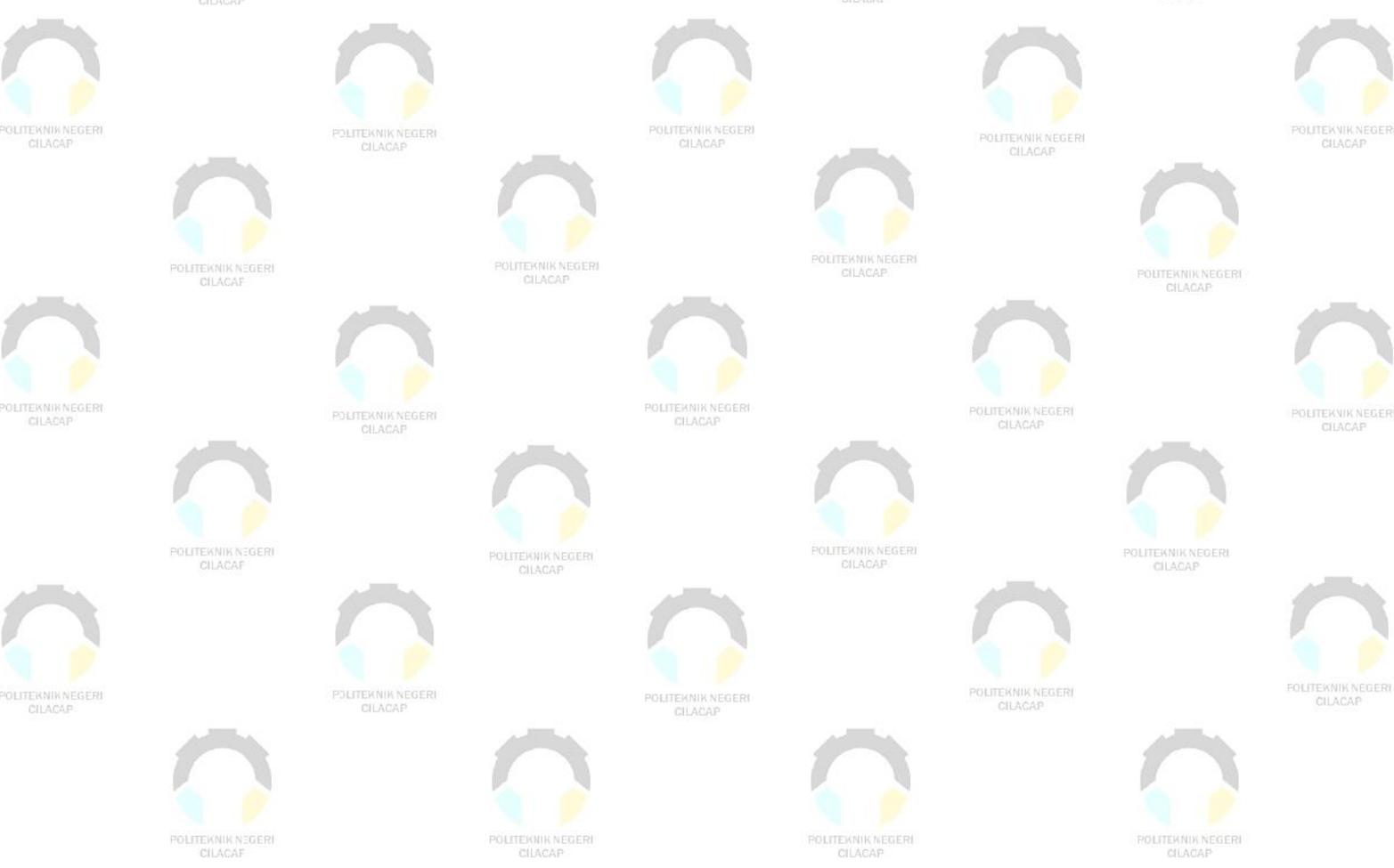




BAB II

LANDASAN TEORI



BAB II LANDASAN TEORI

2.1 Landasan Teori

Penelitian ini memerlukan adanya teori-teori yang menjadi dasar untuk menunjang proses penelitian. Teori yang digunakan dalam penelitian ini sebagai berikut :

2.1.1 Sistem Informasi

Sistem informasi merupakan kumpulan dari beberapa orang yang bekerja sama untuk mencapai tujuan tertentu. Dalam hal ini Sistem Informasi juga mampu mendukung dalam pengambilan keputusan. Pengertian lain juga menyebutkan bahwa Sistem Informasi adalah suatu kombinasi teratur perorangan, *hardware* (perangkat keras), *software* (piranti lunak), jaringan komputer dan komunikasi data dan basis data dalam mengumpulkan, menyebarkan, dan merubah informasi dalam suatu bentuk organisasi[8].

2.1.2 Website

Website (web) merupakan kumpulan dari beberapa halaman yang berisi informasi yang diakses melalui internet pada perangkat yang dapat mengakses internet seperti computer. Web merupakan penyederhanaan dari sebuah istilah yaitu *World Wide Web* yang merupakan bagian dari teknologi internet. *World Wide Web* atau yang biasa disingkat *www*, merupakan sebuah sistem jaringan berbasis *Client-Server* yang mempergunakan protokol HTTP (*Hyperteks Transfer Protocol*) dan TCP/IP (*Transmission Control Protocol / Internet Protocol*) sebagai mediana[9]. Jenis – jenis *website* yaitu sebagai berikut :

a. Website Statis

Website statis merupakan *website* yang isinya tidak diperbaharui secara berkala, sehingga isinya akan selalu sama dari waktu ke waktu. Jenis *website* ini biasa digunakan untuk menampilkan profil dari pemilik *website* seperti profil perusahaan atau organisasi.

b. *Website* Dinamis

Website dinamis merupakan jenis *website* yang isinya diperbaharui secara berkala oleh pengelola *website*. *Website* jenis ini biasa digunakan oleh perusahaan atau perorangan yang aktifitas bisnisnya berkaitan dengan internet contohnya yaitu *web blog* atau *web* berita.

c. *Website* Interaktif

Website interaktif merupakan *website* yang termasuk dalam *website* dinamis yang isi informasinya selalu diperbaharui dari waktu ke waktu oleh pengelola *website* dan pengguna *website*. Sebagai contoh dari *website* interaktif yaitu *website* jejaring sosial seperti facebook dan twitter atau *website marketplace* seperti Bukalapak, Shopee, Tokopedia, dan lain sebagainya.

2.1.3 **Pengelolaan**

Pengelolaan berasal dari kata “kelola” yang berarti mengendalikan, menjalankan, mengurus, atau menangani[10]. Pengelolaan adalah proses melakukan kegiatan untuk membantu merumuskan kebijaksanaan dan tujuan yang memberikan pengawasan pada semua hal yang terlibat dalam pelaksanaan dan pencapaian tujuan[11].

2.1.4 **Surat**

Surat merupakan lembaran kertas tertulis yang didalamnya memuat isi atau maksud tertentu dan digunakan sebagai tanda atau keterangan[12]. Adapaun yang menyatakan bahwa surat adalah informasi tertulis yang digunakan sebagai alat komunikasi tertulis yang dibuat dengan persyaratan surat menyurat. Secara umum surat adalah pernyataan tertulis dalam segala bentuk dan corak yang diatur dan digunakan sebagai sarana komunikasi untuk menyampaikan informasi dari satu pihak kepada pihak lain sesuai dengan ketentuan yang berlaku. Jadi, dapat disimpulkan bahwa surat adalah menyampaikan informasi secara tertulis dari satu pihak ke pihak lain, berupa pemberitahuan, pernyataan, pemikiran, dan sebagainya[13].

Sebagai salah satu contoh jenis surat yaitu surat masuk dan surat keluar.

1. Surat Masuk

Surat masuk adalah surat yang masuk ke dalam suatu instansi/perusahaan atau bagian dalam suatu instansi/perusahaan, baik yang berasal dari instansi/perusahaan lain atau bagian lain pada instansi/perusahaan yang sama[14].

2. Surat Keluar

Surat keluar adalah surat yang sudah lengkap (bertanggal, bernomor, berstampel dan telah ditandatangani oleh pejabat yang berwenang) yang dibuat oleh suatu instansi, kantor atau lembaga untuk ditujukan/dikirim kepada instansi, kantor atau lembaga lain[14].

2.1.5 Disposisi Surat

Disposisi surat adalah tata cara atau prosedur pengelolaan surat dibuat dan disepakati oleh pemimpin perusahaan yang memiliki hak dalam mengurus segala persetujuan surat menyurat pada perusahaan yang dipimpinnya[15].

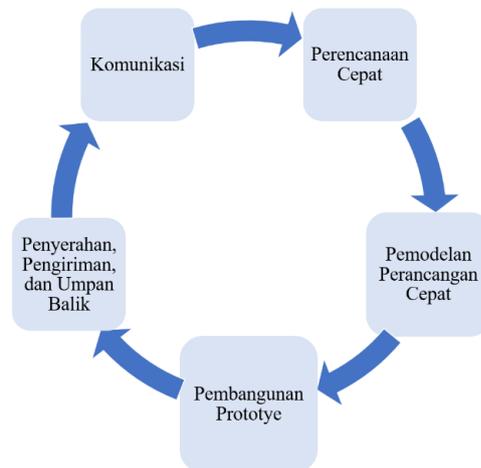
2.1.6 Penomoran Surat

Penomoran surat adalah serangkaian kode yang dimasukkan dalam surat, dan memiliki makna tertentu. Keberadaan nomor surat sangatlah penting dalam pembuatan surat resmi karena dengan nomor surat tersebut menjadi identitas yang membedakan antara satu surat dengan surat yang lainnya. Nomor surat adalah tanda berupa angka atau gabungan angka huruf yang dicantumkan pada surat keluar dari suatu organisasi, instansi, atau badan usaha, sesuai dengan nomor berkas atau buku agenda surat keluar yang berhubungan dengan surat tertentu[16].

2.1.7 Metode Prototype

Metode *prototype* merupakan sebuah metode pengembangan sistem yang menggunakan *prototype* untuk menggambarkan sistem agar pengguna atau pemilik sistem dapat memiliki gambaran tentang pengembangan sistem yang akan

dilakukan. *Prototype* adalah cara yang baik untuk mendapatkan umpan balik tentang sistem yang diusulkan serta dapat menjelaskan bagaimana sistem tersebut tersedia untuk memenuhi kebutuhan pengguna[17].



Gambar 2. 1 Metode *Prototype* menurut Pressman(2015)

Berikut adalah tahap-tahap dalam metode *prototype*[7]:

1. Komunikasi (*Communication*)
Komunikasi dilakukan untuk memperoleh data secara lengkap mengenai identifikasi masalah dan kebutuhan dari pengguna.
2. Perencanaan Cepat (*Quick Plan*)
Tahapan mulai melakukan perencanaan strategis secara cepat dengan memberikan solusi berdasarkan analisis kebutuhan yang digunakan untuk membangun *prototype* yang sesuai dengan kebutuhan pengguna.
3. Pemodelan Perancangan Cepat (*Modeling Quick Design*)
Tahapan ini merupakan tahapan membuat rancangan sistem menggunakan UML (*Unified Modeling Language*) berupa *usecase*, *activity diagram*, *sequence diagram*, *class diagram*, serta *flowchart* sesuai dengan kebutuhan sistem ini.
4. Pembangunan Prototype (*Construction of Prototype*)
Pengerjaan pembuatan *prototype* mulai dilaksanakan. Pengerjaan dilakukan sesuai dengan data yang didapat dari hasil Analisa kebutuhan agar *prototype* yang dibuat benar-benar sesuai dengan kebutuhan pengguna.

5. Penyerahan, Pengiriman, dan Umpan Balik (*Deployment Delivery & Feedback*)

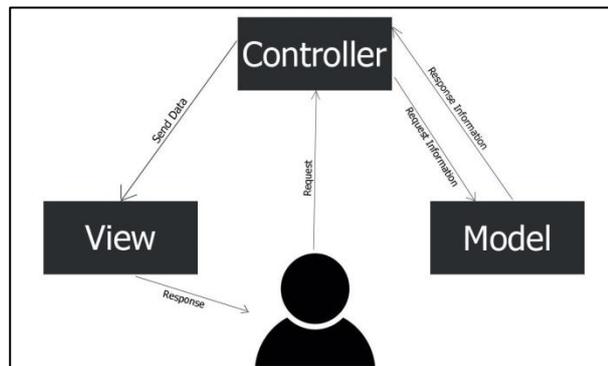
Tahap ini merupakan tahap dimana sistem diserahkan kepada pengguna yang nantinya pengguna akan melakukan pengujian terhadap sistem. Tahap ini dibutuhkan untuk mendapatkan umpan balik atau *feedback* dari pengguna.

2.1.8 Framework Laravel

Laravel adalah *framework open source* PHP berbasis web gratis yang dibuat oleh Taylor Otwell dan ditujukan untuk pengembangan aplikasi web mengikuti *model model-view-controller* (MVC). Laravel merupakan pengembangan *website* berbasis MVP yang ditulis dalam PHP yang dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, serta untuk meningkatkan pengalaman bekerja dengan aplikasi yang menyediakan sintaks yang ekspresif, jelas, dan menghemat waktu[18].

MVC merupakan sebuah pendekatan perangkat lunak yang memisahkan aplikasi logika dari presentasi. MVC memisahkan aplikasi berdasarkan komponen-komponen aplikasi, seperti manipulasi data, *controller*, dan *user interface*[19].

- a. Model mewakili struktur data. Biasanya model berisi fungsi-fungsi yang membantu seseorang dalam pengelolaan basis data, seperti memasukkan data ke basis data, pembaruan data, dan lain-lain.
- b. View adalah bagian yang mengatur tampilan ke pengguna. Bisa dikatakan berupa halaman web.
- c. Controller merupakan bagian yang menjembatani model dan view. Controller berisi perintah-perintah yang berfungsi untuk memproses suatu data dan mengirimkannya ke halaman web.



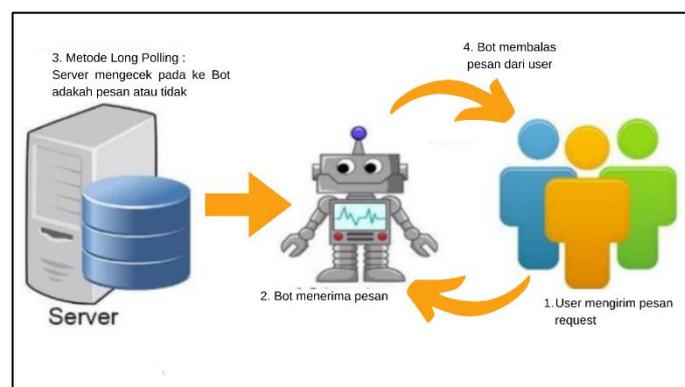
Gambar 2. 2 MVC Model

2.1.9 Telegram

Telegram adalah aplikasi pengiriman pesan berupa teks maupun media (gambar, video, dan file) yang berfokus pada kecepatan, keamanan, sederhana, dan dapat diunduh secara gratis. Telegram dapat digunakan di semua perangkat dalam satu akun dalam waktu yang sama[20]. Telegram mempunyai dua jenis API untuk developer yaitu :

1. Bot API

Bot API memungkinkan developer untuk menghubungkan Bot dengan sistem Telegram. Telegram BOT adalah sebuah akun khusus yang tidak memerlukan nomor telepon tambahan dalam pengaturannya. Akun ini berfungsi sebagai antarmuka untuk tempat berjalannya kode pada suatu server. Berikut merupakan gambaran alur pengiriman Bot Telegram[21]:



Gambar 2. 3 Alur Pengiriman Pesan Bot Telegram

2. Telegram API

Telegram API memungkinkan developer untuk membangun sendiri Telegram *clients* yang diinginkan. Telegram API terbuka 100% untuk semua developer yang ingin membuat aplikasi dengan *platform* Telegram.

2.1.10 PHP

PHP (*hypertext Preprocessor*) adalah sebuah bahasa pemrograman yang berbentuk scripting. PHP merupakan script untuk pemrograman script web server side, script yang membuat dokumen HTML, dokumen HTML yang dihasilkan dari suatu aplikasi bukan dokumen HTML yang dibuat dengan menggunakan editor teks atau editor HTML. Perancangan sistem pakar ini menggunakan bahasa pemrograman PHP dan MySQL sebagai databasenya. MySQL biasanya dipadukan dengan menggunakan program aplikasi PHP, karena dengan menggunakan kedua program tersebut di atas telah terbukti akan kehandalan dalam menangani permintaan data[22].

2.1.11 MySql

MySQL (*My Structured Query Language*) merupakan *database engine* atau *server database* yang mendukung bahasa *database* SQL sebagai bahasa interaktif dalam mengelola data. MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau DBMS (*Database Management System*) yang *multithread* dan *multi-user*[23].

1. *Structural Query Language* (SQL)

SQL (*Structural Query Language*) merupakan Bahasa yang digunakan untuk mengelola data pada RDBMS. Secara umum perintah SQL dibagi ke dalam 3 (tiga) yaitu[24] :

a. *Data Definition Language* (DDL)

Data Definition Language merupakan perintah yang digunakan untuk membuat, mengubah, dan menghapus objek-objek database seperti tabel, view, dan lainnya. Perintah yang termasuk ke dalam kelompok DDL yaitu:

1. **Create**, perintah yang digunakan untuk membuat objek baru.
2. **Alter**, perintah yang digunakan untuk memodifikasi objek.
3. **Drop**, perintah yang digunakan untuk menghapus objek.

b. *Data Manipulation Language* (DML)

Data Manipulation Language merupakan perintah yang digunakan untuk melakukan *query* dan manipulasi data. Perintah yang termasuk ke dalam kelompok DML yaitu:

1. **Insert**, perintah yang digunakan untuk menambahkan data pada *database*.
2. **Update**, perintah yang digunakan untuk mengubah data pada *database*.
3. **Delete**, perintah yang digunakan untuk menghapus data pada *database*.
4. **Select**, perintah yang digunakan untuk menampilkan data/isi tabel pada *database*.

c. *Data Control Language* (DCL)

Data Control Language merupakan perintah yang digunakan untuk mengatur otorisasi (hak akses) terhadap objek-objek *database*. Perintah yang termasuk ke dalam DCL yaitu: *GRANT*, *REVOKE*, dan *DENY*.

2. *Database Management System* (DBMS)

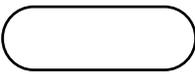
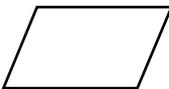
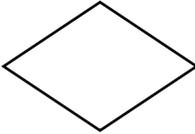
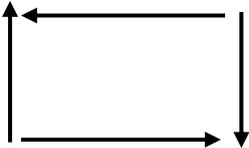
Database Management System merupakan sebuah perangkat lunak yang digunakan untuk mengelola dan mengatur data dalam suatu basis data. Basis data merupakan kumpulan data yang terstruktur, tersimpan, dan dapat diakses dengan mudah[25].

2.1.12 *Flowchart* Sistem

Flowchart adalah penggambaran secara grafik mengenai langkah-langkah dan urutan prosedur dari suatu program. *Flowchart* sistem merupakan suatu urutan proses dalam sistem dengan menunjukkan alat dari media input, output serta jenis media yang digunakan untuk penyimpanannya dalam proses pengolahan data. *Flowchart* digambarkan dengan menggunakan simbol-simbol standar yang

mewakili suatu proses tertentu[26]. Berikut merupakan simbol-simbol yang digunakan dalam penggambaran *flowchart* :

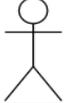
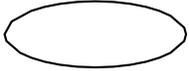
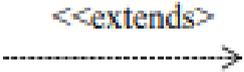
Tabel 2. 1 Simbol Simbol *Flowchart*

No	Simbol	Nama	Fungsi
1.		<i>Terminal</i>	Memulai atau mengakhiri program
2.		<i>Processing</i>	Menunjukkan proses perhitungan atau proses pengolahan data yang dilakukan oleh komputer
3.		<i>Input / Output</i>	Menyatakan informasi data <i>input</i> atau <i>output</i>
4.		<i>Decision</i>	Memilih proses yang akan dilakukan selanjutnya berdasarkan kondisi tertentu
5.		<i>Manual Operation</i>	Menunjukkan proses perhitungan atau proses pengolahan data yang tidak dilakukan oleh komputer
6.		<i>Manual Input</i>	Menginputkan data manual dengan <i>keyboard</i>
7.		<i>Document</i>	Menyatakan masukan dan keluaran yang berasal dari dokumen
8.		<i>Storage Data</i>	Menunjukkan penyimpanan data baik file maupun database dalam sebuah sistem atau proses.
9.		<i>Flow</i>	Menghubungkan simbol yang satu dengan yang lainnya, menyatakan arus suatu proses

2.1.13 Use Case Diagram

Diagram *use case* merupakan model untuk perilaku dari sistem informasi yang akan dibangun. *Use case* digunakan untuk mengetahui fungsi apa yang ada dalam sistem informasi dan siapa yang memiliki kewenangan untuk menggunakan fungsi tersebut. Berikut ini simbol-simbol yang digunakan pada *use case* diagram[27].

Tabel 2. 2 Simbol Simbol *Use Case Diagram*

No	Simbol	Nama	Fungsi
1.		<i>Aktor</i>	Mewakili peran orang atau alat ketika berkomunikasi dengan <i>use case</i>
2.		<i>Use Case</i>	Abstraksi dan interaksi antara sistem dan aktor
3.		<i>Association</i>	Abstraksi dari penghubung antara aktor dengan <i>use case</i>
4.		<i>Generalisasi</i>	Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i>
5.		<i>Extend</i>	Menunjukkan bahwa suatu <i>use case</i> merupakan tambahan fungsional dari <i>use case</i> lainnya jika suatu kondisi terpenuhi
6.		<i>Include</i>	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya

2.1.14 ERD (*Entity Relationship Diagram*)

Entity Relationship Diagram (ERD) adalah diagram berbentuk notasi grafis yang berada dalam pembuatan database yang menghubungkan antara data satu dengan yang lain. Fungsi ERD adalah sebagai alat bantu dalam pembuatan database dan memberikan gambaran bagaimana kerja database yang akan dibuat[28].

Terdapat 3 elemen dasar pada ERD, yaitu[28]:

1. Entitas

Entitas merupakan objek yang akan menjadi perhatian dalam suatu database. Entitas dapat berupa manusia, tempat, benda, atau kondisi mengenai data yang dibutuhkan. Simbol dari entitas berbentuk persegi panjang.

2. Atribut

Atribut merupakan informasi yang terdapat dalam entitas. Sebuah entitas harus memiliki primary key sebagai ciri khas entitas dan atribut deskriptif.

Atribut biasanya terletak dalam tabel entitas atau dapat juga terpisah dari tabel. Simbol dari atribut berbentuk elips.

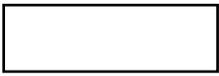
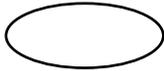
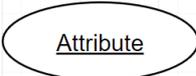
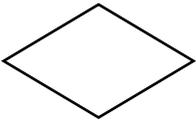
3. Relasi

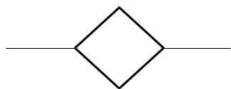
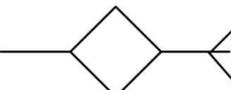
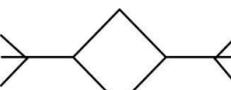
Relasi di dalam ERD merupakan hubungan antara dua atau lebih entitas. Simbol dari relasi berbentuk belah ketupat. Relasi yang dapat dimiliki oleh ERD ada beberapa macam, yaitu[28]:

- a. *One to One* Satu, anggota entitas dapat berelasi dengan satu anggota entitas lain.
- b. *One to Many* Satu, anggota entitas dapat berelasi dengan beberapa anggota entitas lain.
- c. *Many to Many*, Beberapa anggota entitas dapat berelasi dengan beberapa anggota entitas lain.

Simbol-simbol yang digunakan pada pembuatan ERD sebagai berikut[29] :

Tabel 2. 3 Simbol-Simbol ERD

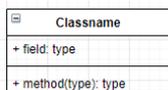
No	Simbol	Nama	Fungsi
1.		Entitas	Sesuatu di dalam sistem atau diluar sistem yang saling berinteraksi berupa data yang akan disimpan.
2.		Atribut	Elemen dari setiap entitas.
3.		<i>Key</i> Atribut	Elemen entitas yang menunjukkan kunci utama (Primary Key).
4.		<i>Weak Key</i> Atribut	Elemen entitas yang menunjukkan kunci yang memiliki hubungan dengan kunci utama pada entitas lain (<i>Foreign Key</i>).
3.		<i>Relationship</i>	Hubungan antara entitas satu dengan entitas lainnya.
4.		<i>Link</i>	Menghubungkan satu entitas dengan entitas lainnya.

5.		Relasi <i>One to One</i> (1 : 1)	Relasi yang menunjukkan bahwa suatu entitas memiliki hubungan paling banyak satu dengan entitas lainnya.
6.		Relasi <i>One to Many</i> (1 : N)	Relasi yang menunjukkan bahwa suatu entitas dengan entitas lainnya memiliki hubungan satu banding banyak atau sebaliknya.
7.		<i>Relasi Many to Many</i> (N : N)	Relasi yang menunjukkan bahwa suatu entitas dengan entitas lainnya memiliki hubungan banyak banding banyak atau lebih dari satu.

2.1.15 Class Diagram

Class diagram adalah salah satu pemodelan UML yang berfungsi untuk membuat sebuah *logical models* dari sebuah sistem. *Class diagram* menunjukkan bagaimana skema dari arsitektur sebuah sistem yang sedang dirancang. *Class diagram* digambarkan dengan *class* yang berisi atribut dan *method* dan dihubungkan dengan sebuah garis yang disebut asosiasi[30]. Simbol-simbol yang digunakan pada pembuatan *class diagram* adalah sebagai berikut :

Tabel 2. 4 Simbol-Simbol Class Diagram

No	Simbol	Nama	Fungsi
1.		<i>Generalization</i>	Hubungan Dimana objek anak berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk.
2.		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
3.		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terstruktur bagi suatu aktor.

4.	←-----	<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
5.	----->	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri akan mempengaruhi elemen yang bergantung padanya (elemen tidak mandiri).
6.	_____	<i>Association</i>	Garis yang berfungsi untuk menghubungkan antara objek satu dengan objek lainnya.