

BAB II

LANDASAN TEORI

2.1 Tinjauan Studi

Adapun penelitian terdahulu yang berhubungan dengan topik peneliti untuk dijadikan sebagai bahan referensi dan pegangan untuk di jadikan pembanding dengan penelitian yang akan dilakukan penulis. Untuk jurnal yang pertama berjudul “Aplikasi Sistem Layanan Pesan Antar Makanan Berbasis Android Pada Kedai Ayam Remuk, Tangerang” yang disusun oleh Embun Fajar Wati yang mengangkat permasalahan tentang pemesanan yang dilakukan melalui telepon dengan menyebutkan pesanan satu per satu. Pemesanan memerlukan waktu yang kurang efisien dan dapat menyebabkan terjadinya kesalahan penginputan menu saat melakukan pemesanan. Kelemahan pada jurnal tersebut tidak ada petunjuk yang terhubung langsung ke lokasi pemesanan. Namun pada penulis akan menghubungkan langsung petunjuk lokasi pemesan jasa.(Wati, 2018) Selanjutnya pada jurnal berjudul “Progam bantu pemesanan jasa perbaikan ac studi kasus: cv. kurniatama” yang dibuat oleh Halim Budi Santoso, Darma Cahyadi, Erick Kurniawan(Halim Budi Santoso, Darma Cahyadi, 2017) kekurangan dari penelitian ini hanya pencatatan pemesanan transaksi servis dan juga pemesanan suku cadang hanya pada cv. Kurniatama tidak menyeluruh, namun pada penulis akan dibuatkan wadah untuk semua penyedia jasa servis jepara supaya lebih efisien karena menggunakan lokasi terdekat dengan pelanggan.

Selanjutnya yaitu jurnal yang disusun oleh Julianto Simatupang, yang di buat berdasarkan permasalahan yang terdapat di transaksi hanya mengandalkan nota transaksi manual, hal tersebut bisa terjadi masalah jika dokumen hilang. Hasil dari sistem ini adalah sistem aplikasi yang mampu menangani pendataan suku cadang, pendataan pelanggan, transaksi penjualan, transaksi layanan jasa, transaksi pembayaran dan pembuatan laporan yang dibutuhkan perusahaan. Pembeda dari jurnal berjudul “Perancangan sistem informasi jasa servis kendaraan dan penjualan suku cadang pada jaya bersama”(Julianto Simatupang,

2019) dengan penelitian yang dilakukan adalah sistem dari jurnal tersebut hanya untuk pendataan data seperti sistem kasir biasa berbeda dengan yang penulis lakukan yang juga mencakup pencarian dan pemesanan jasa secara online dengan lokasi terdekat dengan pelanggan.

Liochita Cleaning merupakan perusahaan yang bergerak di jasa kebersihan yang berlokasi di Semarang. Pada perusahaan ini membuka jasa pencucian karpet dan kebersihan kaca gedung. Pada perusahaan ini data pelanggan dan data transaksi pesanan masih menggunakan kertas sehingga dapat beresiko data tersebut rusak atau hilang selain itu untuk pencatatan data kurang terperinci karena data transaksi dan data pemesan masih menjadi satu. Dengan demikian solusi untuk masalah tersebut adalah dibuat sistem informasi berbasis web dan mobile yang direkdisikan mampu membantu pelanggan dalam melakukan pemesanan. Jika pada jurnal hanya mencakup pada liochita saja namun pada penulis nantinya akan bisa mencakup ke pengguna jasa servis yang ada. (Agung, Kridalukmana, & Windasari, 2016)

Adapun ide lain dari jurnal berjudul “Aplikasi pencarian pariwisata dan tempat oleh – oleh terdekat menggunakan metode haversine berbasis android”. Informasi mengenai pariwisata dan tempat oleh – oleh di Malang merupakan suatu hal yang dibutuhkan, oleh karena itu dibutuhkannya sebuah aplikasi berbasis android yang dapat membantu masyarakat dalam menemukan lokasi pariwisata dan tempat oleh – oleh di Malang. Pada penulis akan sama menggunakan metode haversine untuk menentukan jarak yang terdekat. Tetapi kekurangan pada peneliti aplikasi tersebut belum adanya detail tentang lokasi, hanya menunjukkan arah lokasi kepada pengguna. Namun pada penulis akan menambahkan fitur tentang detail pencarian jasa servis elektronik. (Purnawan, Marisa, & Wijaya, 2018)

2.2 Tinjauan Pustaka

2.2.1 Definisi Sistem

Sistem adalah kumpulan atau himpunan dari unsur, komponen atau variabel-variabel yang terorganisir, saling berinteraksi, saling tergantung satu

sama lain dan terpadu untuk mencapai tujuan tertentu. Unsur dari sistem terdiri dari masukan (input), pengolahan (processing), dan keluaran (output). Pada dasarnya sesuatu dapat disebut sistem apabila memenuhi beberapa syarat, yaitu bila memiliki bagian (sub sistem) yang saling berinteraksi dengan maksud untuk mencapai suatu tujuan tertentu, dan harus memiliki unsur input sebagai penggerak atau pemberi tenaga dimana sistem itu dioperasikan, proses sebagai aktivitas yang mengubah input menjadi output, dan output sebagai hasil operasi. Jadi suatu sistem terdiri dari prosedur sebagai bagian-bagian yang saling berinteraksi dalam rangkaian unsur input, proses, dan output.(Yunita & Devitra, 2017)

2.2.2 Definisi Infomasi

Informasi adalah data yang diproses menjadi sebuah kumpulan yang berarti dan bermanfaat dalam proses pengambilan keputusan saat ini atau kedepannya. Sedangkan menurut McLeod informasi sebagai data yang telah diproses menjadi bentuk yang lebih berarti bagi penerimanya.(Sari & Devitra, 2017)

2.2.3 Sistem Informasi

Menurut Laudon, Sistem Informasi Merupakan “*a set of interrelated components that collect (or retrieve), process, store, and distribute information to support decision making and control in an organization.*” Pengertian diatas memiliki arti bahwa Sistem informasi merupakan satuan komponen yang berhubungan yang mengumpulkan, memproses, menyimpan, dan menyebarkan informasi untuk mendukung dalam pengambilan keputusan dan kendali dalam suatu organisasi. (Loveri, 2018)

2.2.4 Jasa

Jasa umumnya mencerminkan produk tidak berwujud fisik (Intangible) atau sektor industri spesifik, seperti pendidikan, kesehatan, telekomunikasi, transportasi, asuransi, perbankan, perhotelan, konstruksi, perdagangan, rekreasi dan seterusnya.(Lubis, Lie, Butarbutar, & Inrawan, 2017)

2.2.5 Servis

adalah kegiatan mengganti atau memperbaiki sebagian dari peralatan yang rusak agar dapat beroperasi kembali sesuai fungsi dan kemampuannya seperti keadaan sebelum rusak. Peranan repair elektronik untuk memperbaiki sesuatu elektronik yang telah rusak, memperpanjang masa pemakaian pada komponen-komponen, memperbaiki adanya kerusakan serius pada komponen elektronik.(Puspitasari, Atika, & Lestari, 2020)

2.2.6 Elektronik

Elektronik yaitu suatu alat yang dibuat atau di rancang untuk dipergunakan manusia berdasarkan pada prinsip pada sistem elektronika. Alat elektronik sudah menjadi kebutuhan sehari – hari yang digunakan untuk memenuhi kebutuhan hidupnya. Baik untuk kebutuhan bekerja, komunikasi, mencuci, menonton, belajar dan lain – lain.(Puspitasari et al., 2020)

Berikut adalah contoh beberapa elektronik :

- a. TV
- b Kulkas
- c. Ac
- d. Kipas angin
- e. Magicom

2.2.7 Jasa Servis Elektronik

Usaha Jasa Service Elektronik merupakan usaha jasa yang memerlukan keahlian khusus sebagai modal utama, maka sebelum memulai usaha ini, baik itu jadi usaha sampingan maupun usaha pokok kita mesti memiliki dan mempelajari terlebih dahulu hal hal yang berhubungan dengan elektronik. Seperti cara menganalisa dan memperkirakan kerusakan secara cepat, cara membongkar, memperbaiki dan merakit kembali barang elektronik yang rusak dan mengetahui kisaran harga komponen elektronik yang mungkin harus diganti.(Puspitasari et al., 2020)

2.2.8 Android

Android adalah sebuah Sistem Operasi untuk perangkat mobile berbasis Linux yang mencakup Sistem Operasi, Middleware dan aplikasi. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Awalnya, Google Inc (Safaat H-2012) Sedangkan yang dimaksud dengan Android Development Kit atau Android SDK adalah tool API (Application Programming Interface) yang diperlukan untuk mulai mengembangkan aplikasi pada Platform Android menggunakan Bahasa Pemrograman Java. Android merupakan subset perangkat lunak untuk ponsel meliputi Sistem Operasi Middleware dan aplikasi kunci yang di release oleh Google, yang terdiri dari Framework, Virtual Dalvick, Integrated Browser serta media link sebagai pendukungnya.(Arif rizki, 2017)

2.2.9 Android Studio

Android Studio adalah sebuah Integrated Development Environment (IDE) utama Google untuk mengembangkan pada platform Android. Karena Android Studio adalah IDE dari Google, oleh karena itu software ini bisa secara langsung terhubung dengan Google Maps melalui API Key yang dibuat melalui laman yang disiapkan dari Google Maps API untuk mengintegrasikan peta dengan software tersebut kemudian peta secara otomatis ditampilkan di aplikasi yang dibuat. Selain terintegrasi dengan Google Maps, Android Studio juga dapat terintegrasi database SQLite Manager, plugin untuk pengolahan dan penyimpanan informasi yang saling berkaitan untuk kemudian dibuat algoritma dari tiap data yang akan ditampilkan.(Anisa Rachmawati, Arief Laila Nugraha, 2017)

2.2.10 Google Maps API

Google Maps adalah layanan aplikasi peta online yang disediakan oleh Google secara gratis. Layanan peta Google Maps secara resmi dapat diakses melalui situs <http://maps.google.com>. Google Map menawarkan peta dan gambar satelit untuk seluruh dunia. Layanan ini dibuat Interaktif, karena didalamnya peta dapat digeser sesuai keinginan pengguna, mengubah level zoom, serta mengubah

tampilan jenis peta. Fasilitas lain yang disediakan antara lain adalah pencarian lokasi dengan memasukkan kata kunci, yang dimaksud seperti nama tempat, kota, atau jalan, fasilitas lainnya yaitu perhitungan rute perjalanan dari satu tempat ke tempat lainnya Google Map dibuat dengan menggunakan kombinasi dari gambar peta, database, serta obyek-obyek interaktif yang dibuat dengan bahasa pemrograman HTML, Javascript dan AJAX, serta beberapa bahasa pemrograman lainnya. Gambar yang muncul pada peta merupakan hasil komunikasi dengan database pada Web Server Google untuk menampilkan gabungan dari potongan-potongan gambar yang diminta. Sedangkan Google Maps API adalah sebuah layanan (service) yang diberikan oleh Google kepada para pengguna untuk memanfaatkan Google Map dalam mengembangkan aplikasi. Google Maps API menyediakan beberapa fitur untuk memanipulasi peta, dan menambah konten melalui berbagai jenis services yang dimiliki, serta mengizinkan kepada pengguna untuk membangun Aplikasi Enterprise di dalam Website nya. (Arif rizki, 2017)

2.2.11 Database

Menurut Sutabri dalam buku Sistem Informasi Manajemen, Database adalah suatu kumpulan data terhubung yang disimpan secara bersama-sama pada suatu media, tanpa mengatap satu sama lain atau tidak perlu suatu kerangkapan data. (Fitri Ayu and Nia Permatasari, 2018)

Ada beberapa operasi database diantaranya adalah :

1. Pembuatan database baru (create database), merupakan pembuatan lemari arsip yang baru.
2. Penghapusan database (drop database), merupakan penghapusan dari lemari arsip (sekaligus beserta isinya jika ada)
3. Pembuatan file atau tabel (create tabel), merupakan penambahan map arsip baru ke sebuah lemari arsip yang telah ada.
4. Penghapusan file atau tabel (drop table) merupakan penghapusan dari map arsip lama yang ada di sebuah lemari arsip.
5. Penambahan atau pengisian data baru (insert), merupakan penambahan lembaran arsip ke sebuah map arsip.

6. Pengambilan data dari sebuah file atau tabel (retrieve atau search), merupakan pencarian lembaran arsip dari sebuah map arsip.
7. Pengubahan data dari sebuah file atau tabel (update), merupakan perbaikan atau pengeditan dari sebuah lembaran arsip yang ada di sebuah map arsip.
8. Penghapusan data dari sebuah file atau table (delete), merupakan penghapusan sebuah lembaran arsip yang ada di sebuah map arsip.

2.2.12 Metode Haversine Formula

Haversine Formula merupakan persamaan penting dalam sistem navigasi, pada formula haversine ini akan menghasilkan jarak terpendek antara dua titik. Haversine formula ini merupakan penerapan dari konsep trygonometri yang merupakan bagian dari geometri(Purnawan et al., 2018)

Rumus haversine ialah persamaan yang penting dalam navigasi, yang menghasilkan jarak lingkaran besar antara dua titik (latitude dan longitude) pada permukaan bola (bumi) berdasarkan bujur dan lintang. Berikut adalah Rumus Haversine :

$$d = 2 \cdot r \cdot \arcsin\left(\sqrt{\sin^2\left(\frac{\phi_2 - \phi_1}{2}\right) + \cos(\text{lat}1) \times \cos(\text{lat}2) \times \sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right)}\right) \dots\dots\dots(2.1)$$

Keterangan :

r = jari – jari bumi sebesar 6371(km)

ϕ_1 = latitude 1

ϕ_2 =latitude 2

λ_1 = longitude 1

λ_2 = longitude 2

d = jarak (km)

Sudut pada rumus menggunakan satuan radian untuk menggunakan fungsi trygonometri. Rumus haversine digunakan dalam penelitian ini untuk pencarian jasa servis terdekat dari pengguna. Berikut adalah contoh metode haversine dalam perhitungan jarak antara dua titik :

- a. Titik kordinat pertama

$$\text{Latitude 1} = -0,53951879 * \frac{\pi}{180}$$

$$= -0,00942 \text{ Radian}$$

$$\text{Longitude 1} = 123,07590044 * \frac{\pi}{180}$$

$$= 2,14808 \text{ Radian}$$

b. Titik kordinat kedua

$$\text{Latitude 2} = -0,55089201 * \frac{\pi}{180}$$

$$= -0,00961 \text{ Radian}$$

$$\text{Longitude 2} = 123,06661819 * \frac{\pi}{180}$$

$$= 2,147918 \text{ Radian}$$

$$c. d = 2 \cdot r \cdot \arcsin\left(\sqrt{\sin^2\left(\frac{\phi_2 - \phi_1}{2}\right) + \cos(\text{lat}1) \times \cos(\text{lat}2) \times \sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right)}\right)$$

$$= 2 \times 6371 \times 0,000128$$




$$= 1,632$$







Jadi untuk hasil jarak antara koordinat pertama dan kedua adalah 1,63 km yang dimana hasil tersebut di urutkan dan dicari hasil yang memiliki nilai terkecil sebagai lokasi yang terdekat.

2.2.13 Flowchart

Flowchart merupakan penggambaran secara grafik dari langkah – langkah dan urutan prosedur suatu program. Biasanya mempermudah penyelesaian masalah yang khususnya perlu dipelajari dan dievaluasi lebih lanjut. (Burhanuddin, 2019) Adapun simbol – simbol flowchat dapat di lihat pada tabel 2.1 :

Tabel 2.1 Simbol Flowchart

Nama	Simbol	Keterangan
<i>Terminal</i>		Digunakan untuk menyatakan memulai atau mengakhiri program.
<i>Input atau Output</i>		Digunakan untuk meminta masukan yang kemudian nilainya disimpan dalam sebuah variabel.
Proses manual		Digunakan untuk menunjukkan suatu



		pekerjaan yang dilakukan secara manual.
Proses		Digunakan untuk menunjukkan pengolahan yang akan dilakukan dalam komputer.
<i>Decision</i>		Digunakan untuk menentukan langkah selanjutnya dari percabangan yang ada.
<i>Conector</i>		Digunakan untuk penghubung bila flowchart terputus
Simpanan <i>offline</i>		Menunjukkan file non komputer yang diarsip,urut angka (N), huruf (A) dn tanggal (C)
Garis alir		Menunjukkan arus dari proses / dokumen
<i>Preparation</i>		Proses inialisasi / pemberian harga awal

2.2.13 ERD (Entity Relationship Diagram)

Entity Relationship Diagram (Diagram E-R) adalah yang digunakan untuk menggambarkan model Entity Relationship yang berisi komponen-komponen. Himpunan Entitas dan Himpunan Relasi yang masing-masing dilengkapi dengan atribut-atribut yang mempersentasikan seluruh fakta dari dunia nyata yang kita tinjau.(Sukmaindrayana & Sidik, 2017) Sementara seolah-olah teknik diagram atau alat peraga memberikan dasar untuk desain database relasional yang mendasari sistem informasi yang dikembangkan. ERD bersama-sama dengan detail pendukung merupakan model data yang pada gilirannya digunakan sebagai spesifikasi untuk database. Atau pengertian sempitnya adalah sebuah konsep yang mendeskripsikan hubungan antara penyimpanan (database) dan didasarkan pada persepsi dari sebuah dunia nyata yang terdiri dari

sekumpulan objek yaitu disebut sebagai entity dan hubungan atau relasi antar objek- objek tersebut. Adapun simbol – simbol ERD dapat di lihat pada tabel 2.2 :

Tabel 2.2 Simbol - Simbol ERD

Nama	Simbol	Keterangan
Entitas		Suatu objek yang dapat didefinisikan dalam lingkungan pemakai.
Relasi		Menunjukkan adanya hubungan diantara sejumlah entitas yang berbeda.
Atribut		Karakteristik dari entitas/relasi yang merupakan penjelasan detail tentang entitas.
Garis		Hubungan antara entitas dengan atributnya dan himpunan entitas dengan himpunan relasinya.

Dalam pembentukan ERD terdapat 3 komponen yang akan dibentuk yaitu :

1. Entitas

merupakan mengenai basis data yaitu suatu obyek yang dapat dibedakan dari lainnya yang dapat diwujudkan dalam basis data. Contoh : Mahasiswa, Kartu Anggota Perpustakaan (KAP), dan Buku.

2. Hubungan (relasi/relationship)

Suatu hubungan adalah hubungan antara dua jenis entitas dan direpresentasikan sebagai garis lurus yang menghubungkan dua entitas.

Contoh : Mahasiswa mendaftar sebagai anggota perpustakaan (KAP), relasinya adalah mendaftar.

3. Atribut

Memberikan informasi lebih rinci tentang jenis entitas. Atribut memiliki struktur internal berupa tipe data.

a. Derajat Relasi atau Kardinalitas ERD

1. One to One (1:1)

Setiap anggota entitas A hanya boleh berhubungan dengan satu anggota entitas B, begitu pula sebaliknya.

2. One to many (1:M / Many)

Setiap anggota entitas A dapat berhubungan dengan lebih dari satu anggota entitas B tetapi tidak sebaliknya.

3. Many to Many (M:M)

Setiap entitas A dapat berhubungan dengan banyak entitas himpunan entitas B dan demikian pula sebaliknya

- b. Tahapan ERD

1. Menentukan entitas yaitu menentukan peran, kejadian, lokasi, hal nyata dan konsep dimana pengguna untuk menyampaikan data
2. menentukan relasi menentukan hubungan antar pasangan entitas menggunakan matrik relasi
3. Gambar ERD sementara yaitu Entitas di gambarkan Dengan kotak, dan relasi di gambarkan dengan garis
4. Isu Kardinalitas yaitu menentukan jumlah kejadian suatu entitas untuk sebuah kejadian pada entitas yang berhubungan
5. Tentukan kunci utama yaitu menentukan atribut yang mengidentifikasi satu dan hanya satu kejadian pada masing masing
6. Gambar ERD berdasarkan kunci yaitu menghubungkan Relasi many to many
7. Mentukan atribut yaitu mentukan file file yang di perlukan sistem
8. Pemetaan atribut yaitu menghubungkan atribut dengan entitas yang sesuai.

2.2.14 UML (Unified Modeling Language)

UML (Unified Modeling Language) merupakan bahasa dan aturan spesifikasi standar yang telah disepakati dalam rangka membuat dokumentasi, spesifikasi, kalkulasi, visualisasi, proses analisa, gambaran kerja proses, dan rancang bangun dari sebuah sistem perangkat lunak (software). UML sendiri

merupakan penggabungan dari beberapa metode yang kemudian dikembangkan menjadi sebuah metode baru. Metode-metode tersebut yaitu Grady Booch, Ivar Jacobson, Jim Rumbaugh. Dalam perkembangannya, standarisasi dalam metode UML dilakukan dengan standarisasi proses Object Management Group (OMG), sehingga menjadikan UML sebagai bahasa standar dalam permodelan perangkat lunak yang saat ini banyak digunakan oleh berbagai kalangan. (Redi Mulyana, 2017)

UML memiliki komponen atau bagian yang memiliki fungsi masing-masing dalam penggunaan dan kaitannya dengan sistem yang akan dibangun. Berikut adalah beberapa komponen yang ada pada UML:




1. UseCase Diagram





Secara sederhana *usecase* diagram digambarkan sebagai komponen yang memiliki fungsi untuk memodelkan proses kerja atau bisnis dari aplikasi yang akan dibangun. *Usecase* merupakan abstraksi dari proses interaksi antara pengguna dengan fungsi atau tindakan yang dapat dilakukan terhadap sistem. Diagram *usecase* memiliki 3 kegunaan utama yaitu:

- a. Menjelaskan fasilitas atau sistem requirement dari software
- b. Menggambarkan komunikasi atau interaksi pengguna dan sistem
- c. Melakukan serangkaian test dari fungsi sistem secara umum

Adapun simbol – simbol *Use case* dapat di lihat pada tabel 2.3 :

Tabel 2.3 Simbol *Usecase*

Nama	Simbol	Keterangan
<i>Use case</i>		Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan hasil bagi actor.
Aktor		Menspesifikasikan peran pengguna ketika berinteraksi dengan use case.
Asosiasi		Penghubung komunikasi antara actor dan use case.
Ekstensi	<<extend>>	Relasi use case dimana use case

		dapat berdiri sendiri dan tidak bergantung dengan use case yang lain.
<i>Include</i>		Relasi use case dimana use case tidak dapat berdiri sendiri dan bergantung dengan use case yang lain.
Generalisasi		Hubungan antara dua use case dimana yang satu adalah fungsi yang lebih umum.
Note		Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi



2. Activity Diagram




Activity diagram menyediakan analisa dan kemampuan untuk menggambarkan aktivitas proses atau alur kerja ataupun logika pengambilan keputusan dalam sistem. Terdapat beberapa notasi yang digunakan dalam pembuatan activity diagram, diantaranya:

- a. Notasi Activity
- b. Notasi Transition
- c. Notasi Decision
- d. Notasi Synchronization Bars

Adapun simbol – simbol activity diagram dapat di lihat pada tabel 2.4 :

Tabel 2.4 simbol activity diagram


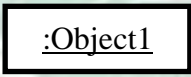



Nama	Simbol	Keterangan
Initial Node		Bagaimana objek di bentuk atau diawali.
Activity		Memperlihatkan bagaimana masing – masing kelas antar muka saling berinteraksi satu sama lain.

Activity Final Node		Bagaimana objek di bentuk atau diakhiri.
Decision		Digunakan untuk menyatakan suatu kondisi dari sistem.
Line Connector		Digunakan untuk menghubungkan satu simbol dengan simbol lain

3. Sequence Diagram

Sequence diagram merupakan gambaran dari keseluruhan tahap yang ada, dan digambarkan secara detail (per tiap tahap) sehingga menghasilkan sesuatu yang sesuai dengan apa yang digambarkan pada use case diagram. Adapun simbol – simbol sequence diagram dapat di lihat pada tabel 2.5 :

Tabel 2.5 sequence diagram

Nama	Simbol	Keterangan
Aktor		Actor juga dapat berkomunikasi dengan object, maka actor juga dapat diurutkan sebagai kolom.
Objek		Object merupakan instance dari sebuah class dan dituliskan tersusun secara horizontal.
<i>Life line</i>		Life line mengindikasikan keberadaan sebuah object dalam basis waktu.
Aktivasi		Activation mengindikasikan sebuah obyek yang akan melakukan sebuah aksi
Pesan		Message mengindikasikan komunikasi antara object-object

4. Class Diagram

Class diagram merupakan deskripsi dari kelompok objek dengan property, operasi (perilaku) dan hubungan (relasi) yang sama. Class diagram berfungsi

untuk menggambarkan dan mengelompokkan struktur dalam suatu kelas atau tingkatan. Class diagram juga berfungsi untuk menjelaskan tipe dari suatu objek yang memiliki hubungan dengan objek lain. Objek yang dimaksud dalam hal ini merupakan nilai yang dimiliki oleh tiap attribute kelas pada entitas. Adapun simbol – simbol class diagram dapat di lihat pada tabel 2.6 :

Tabel 2.6 class diagram

Nama	Simbol	Keterangan
Generalization		Hubungan dimana objek anak (descendent) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (ancestor)
Nary Association		Upaya untuk menghindari asosiasi dengan lebih dari dua objek.
Class		Himpunan dari objek – objek yang berbagi atribut serta operasi yang sama
Collaboration		Deskripsi dari urutan aksi – aksi yang di tampilkan system yang menghasilkan suatu hasil yang terukur bagi suatu aktor
Realization		Operasi yang dilakukan benar – benar oleh suatu objek
Dependency		Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (independent) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
Association		Yang menghubungkan antara objek satu dengan objek lainnya.