

BAB 2

TINJAUAN PUSTAKA

2.1 Profil Instansi

Profil instansi akan menjelaskan mengenai sejarah instansi, logo, visi dan misi, dan struktur organisasi di Kecamatan Mandalajati Kota Bandung.

2.1.1 Sejarah Instansi

Mandala diartikan sebagai wilayah kekuasaan yang sebagian masyarakatnya memiliki pengetahuan keagamaan yang berpengaruh terhadap pengikutnya. Sedangkan jati adalah murni atau asli sehingga diartikan sebagai masyarakat yang memiliki kepribadian yang kuat untuk mengabdikan diri kepada nusa bangsa, dan agama. Dengan ciri masyarakatnya yang memiliki ikatan yang kuat untuk kemajuan daerahnya.

Arti kata dari Mandalajati adalah suatu daerah yang masyarakatnya berpengetahuan luhur dan luas, yang diwadahi oleh suatu tempat untuk menggali pengetahuan, yang mana masyarakatnya terdiri dari tokoh-tokoh yang memiliki jati diri kuat serta memiliki pengetahuan dalam bidang keagamaan, kemasyarakatan, dan pemerintahan.

Semua wilayah Kecamatan Mandalajati merupakan Kecamatan Cicadas yang menjadi Kecamatan Antapani (sekarang) dan pada tahun 2006 sesuai dengan peraturan daerah kota Bandung nomor 06 tahun 2006 tentang pemekaran dan pembentukan wilayah kerja kecamatan dan kelurahan di lingkungan pemerintahan kota Bandung di bentuklah kecamatan baru yaitu Kecamatan Mandalajati.

2.1.2 Logo Instansi

Berikut adalah logo dari Kecamatan Mandalajati Kota Bandung dapat dilihat pada gambar dibawah.



Gambar 2. 1 Logo Kecamatan Mandalajati Kota Bandung

2.1.3 Visi dan Misi

1. Visi

Terwujudnya optimalisasi pelayanan dalam rangka mendukung kota Bandung yang unggul, nyaman dan sejahtera.

2. Misi

Terwujudnya optimalisasi pelayanan, dan meningkatkan kinerja pemerintah Kecamatan Mandalajati secara efektif dan akuntabel.

2.1.4 Kondisi Eksisting Kecamatan Mandalajati

Kecamatan Mandalajati merupakan salah satu bagian wilayah timur kota Bandung dengan memiliki luas lahan sebesar 717 Ha.

Secara administratif Kecamatan Mandalajati dibatasi oleh :

1. Bagian Selatan : Kecamatan Antapani Kota Bandung
2. Bagian Utara : Kecamatan Cimenyan Kabupaten Bandung
3. Bagian Timur : Kecamatan Ujungberung Kota Bandung
4. Bagian Barat : Kecamatan Cibeunying Kidul Kota Bandung

Dan dengan pembagian penggunaan areal tanahnya dapat dilihat pada Tabel berikut ini :

Tabel 2. 1 Penggunaan Areal Tanah

No	Penggunaan	Luas (Ha)
1	Tanah Sawah	15 Ha
2	Tanah Kering (Daratan)	380,65 Ha
3	Tanah Basah	0,5 Ha
4	Fasilitas Umum	171,32

2.1.5 Kondisi Geografis

Kawasan perencanaan yang menjadi lingkup kerja Kecamatan Mandalajati dapat dilihat dalam Tabel 2.2 berikut ini :

Tabel 2. 2 Kawasan Kecamatan Mandalajati

Kecamatan	Kelurahan / Desa
Mandalajati	Jatihandap
	Karang Pamulang
	Pasir Impun
	Sindang Jaya

Secara geografis Kecamatan Mandalajati memiliki bentuk wilayah datar / berombak sebesar 35% dari total keseluruhan luas wilayah. Ditinjau dari sudut ketinggian tanah, Kecamatan Mandalajati berada pada ketinggian 500m diatas permukaan laut. Suhu maksimum dan minimum di Kecamatan Mandalajati berkisar 30°, sedangkan dilihat dari segi hujan berkisar 215 mm/th dan jumlah hari dengan curah hujan yang terbanyak sebesar 45 hari.

2.1.6 Administrasi Pemerintahan

Instansi pemerintahan yang berada di wilayah Kecamatan Mandalajati terdiri dari :

1. Instansi Vertikal berjumlah 8 unit, terdiri dari :
 - a. Koramil
 - b. Kapolsek
 - c. Dinas Pendidikan
 - d. KUA

- e. PLKB
 - f. PPL Pertanian
 - g. Puskesmas
 - h. PD. Kebersihan
2. Instansi Otonomi berjumlah 2 unit terdiri dari :
 - a. Puskesmas (5 buah)
 - b. PPL Pertanian
 3. Kelembagaan Kelurahan

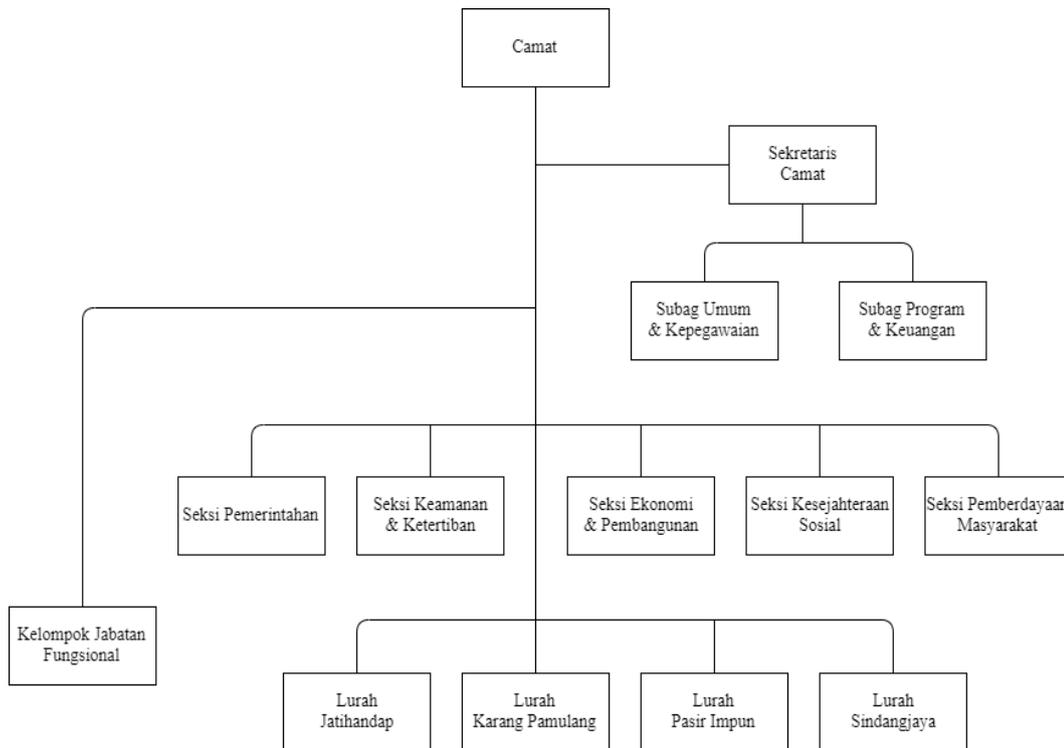
Dalam menjalankan roda pemerintahan, Kecamatan Mandalajati dibagi dalam 4 Kelurahan dengan jumlah 295 RT serta 52 RW seperti di uraikan pada Tabel 2.3 sebagai berikut :

Tabel 2. 3 Kelurahan dan Jumlah RT / RW

No	Kelurahan	Jumlah RT	Jumlah RW
1	Jatihandap	117	17
2	Karangpamulang	71	13
3	Pasir Impun	57	11
4	Sindang Jaya	60	12
	JUMLAH	305	53

2.1.7 Struktur Organisasi

Struktur organisasi merupakan gambaran susunan dari setiap jabatan yang memiliki wewenang dan tanggung jawab dari masing-masing pekerjaan yang ada di lingkungan sekolah. Adapun struktur organisasi di Kecamatan Mandalajati dapat dilihat pada gambar dibawah ini :



Gambar 2. 2 Struktur Organisasi Kecamatan Mandalajati

2.1.8 Deskripsi Jabatan

Kecamatan Mandalajati memiliki susunan jabatan sebagaimana yang tercantum dalam struktur organisasi diatas, dan jabatan tersebut memiliki wewenang masing-masing. Adapun deskripsi jabatan dapat dilihat pada tabel di bawah ini :

Tabel 2. 4 Deskripsi Jabatan

Jabatan	Deskripsi Jabatan
Camat	Membantu kepala daerah dalam melaksanakan kewenangan di wilayah kecamatan
	Memimpin, mengkoordinasikan, mengendalikan serta mengawasi semua kegiatan kecamatan
	Mengusulkan penetapan pegawai kantor dalam jabatan tertentu di lingkungan kantor berdasarkan peraturan perundang-undangan yang berlaku

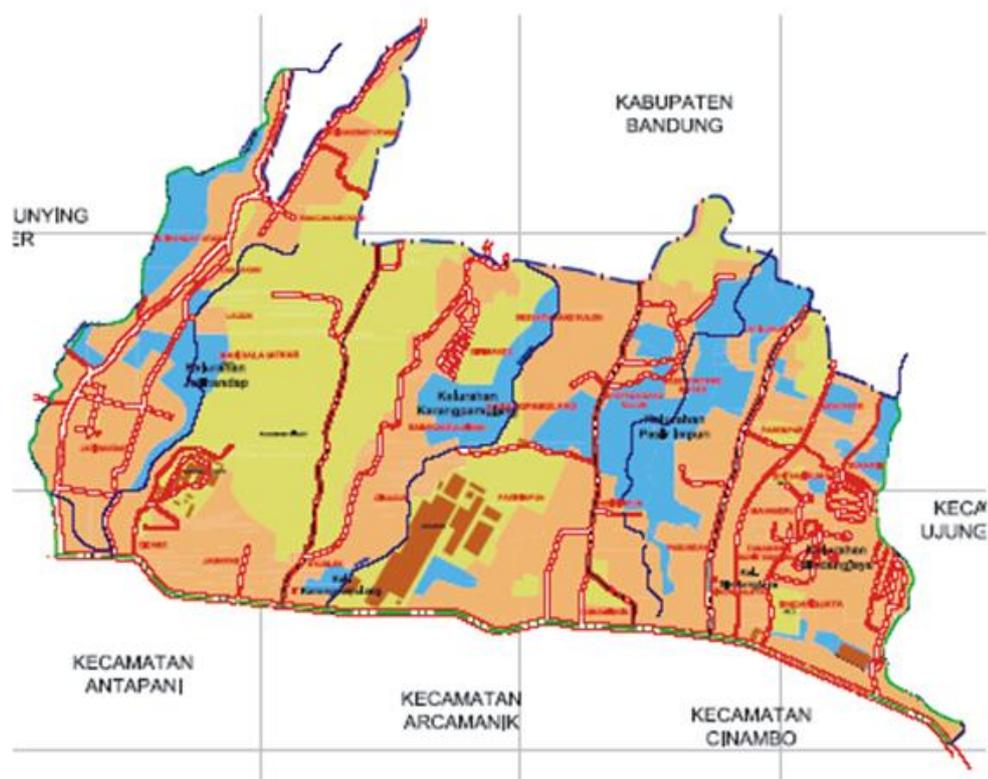
	Menjalin kerjasama dengan semua dinas / instansi untuk kepentingan dan kelancaran pelaksanaan tugas
	Melaksanakan tugas lain yang diberikan pimpinan sesuai dengan bidang tugas
Sekretaris Camat	Penyelenggaraan pengelolaan administrasi perkantoran, administrasi keuangan dan administrasi kepegawaian
	Penyelenggaraan urusan umum dan perlengkapan, keprotokolan dan hubungan masyarakat
	Penyelenggaraan ketatalaksanaan dan kearsipan
	Pelaksanaan koordinasi, pembinaan, pengendalian, evaluasi dan pelaporan pelaksanaan kegiatan kegiatan unit kerja
	Pelaksanaan tugas lain yang diberikan oleh camat sesuai dengan tugas dan fungsinya
Subag Umum & Kepegawaian	Penyusunan rencana penyusunan rencana pengelolaan administrasi dan kepegawaian kecamatan
	Menyusun rencana, program kerja dan anggaran berbasis kinerja berdasarkan tugas pokok dan fungsi kecamatan
	Membagi tugas kepada bawahan sesuai bidang tugas agar tercipta pemerataan tugas
	Memberi petunjuk dan arahan kepada bawahan guna kejelasan pelaksanaan tugas
	Mengawasi pelaksanaan tugas bawahan agar tidak terjadi penyimpangan
	Memeriksa hasil kerja bawahan untuk mengetahui kesulitan dan hambatan serta memberikan jalan keluarnya
	Menilai hasil kerja bawahan secara periodic guna bahan peningkatan kinerja
	Melaksanakan koordinasi guna kelancaran pelaksanaan tugas
	Menyiapkan penyusunan rencana kerja anggaran maupun program kerja tahunan di tingkat kecamatan

Subag Program & Keuangan	Menyiapkan bahan pedoman serta petunjuk teknis di bidang program dan keuangan
	Menyiapkan bahan penyusun rencana strategis, rencana kerja tahunan, serta penetapan kinerja di lingkup kecamatan
	Menyiapkan bahan koordinasi dan konsultasi di bidang perencanaan dan keuangan
	Sebagai pelaksana pengelolaan keuangan
	Pengkordinasian pelaksanaan tugas pemegang kas
	Sebagai coordinator dalam menyusun laporan keuangan dan pelaksanaan kegiatan
Seksi Pemerintahan	Membantu camat dalam menyiapkan bahan perumusan kebijakan, pelaksanaan, evaluasi, dan pelaporan urusan pemerintahan
	Menyusun rencana kerja seksi
	Menyiapkan bahan dan melaksanakan pembinaan pemerintahan
	Mengkordinasi kegiatan administrasi kependudukan
	Melaksanakan pengawasan dan pendataan atas tanah-tanah negara
	Melakukan monitoring, evaluasi dan pelaporan pelaksanaan kegiatan seksi
Seksi Keamanan & Ketertiban	Melaksanakan kordinasi guna kelancaran pelaksanaan tugas
	Melaksanakan pembinaan kesatuan bangsa, organisasi social politik, organisasi kemasyarakatan
	Melakukan pengamanan terhadap pelaksanaan peraturan daerah maupun peraturan lainnya
	Mencegah pengambilan sumber daya alam tanpa izin yang dapat mengganggu serta membahayakan lingkungan hidup
	Melaksanakan penertiban izin mendirikan bangunan

Seksi Ekonomi & Pembangunan	Mengumpulkan dan mengolah data informasi yang berhubungan dengan bidang ekonomi dan pembangunan sarana prasaran fisik
	Menyiapkan bahan fasilitasi dan koordinasi penyelenggaraan pembangunan di wilayah kerjanya
	Menyiapkan bahan kerjasama pelaksanaan pembangunan swadaya masyarakat dan penyelenggaraan kerjasama antar desa
	Melakukan pengendalian, evaluasi dan pelaporan pelaksanaan rencana pembangunan desa
Seksi Kesejahteraan Sosial	Melakukan pembinaan tentang kegiatan pemberdayaan masyarakat
	Memfasilitasi kegiatan keagamaan, organisasi social dan Lembaga swadaya masyarakat
	Melaksanakan penanggulangan masalah social
	Membuat laporan dan evaluasi kegiatan sebagai bahan pertanggungjawaban
Seksi Pemberdayaan Masyarakat	Mendorong partisipasi masyarakat untuk ikut serta dalam pemberdayaan pembangunan lingkup kecamatan melalui forum musyawarah perencanaan pembangunan di kecamatan
	Menyiapkan bahan kordinasi dengan satuan kerja perangkat daerah
	Melaksanakan kordinasi dengan pihak swasta dalam pemeliharaan prasaran dan fasilitas umum
	Melaksanakan pembinaan dan pengawasan terhadap berbagai kegiatan pemberdayaan masyarakat di wilayah kecamatan
	Melaksanakan kordinasi dalam penyelenggaraan pemberdayaan ekonomi masyarakat
Melaksanakan pembinaan dan pengawasan kelembagaan masyarakat , sosial dan budaya	

Lurah	Menyusun program kerja kegiatan di tingkat kelurahan
	Menyusun usulan rencana anggaran belanja kelurahan
	Menyelenggarakan ketatausahaan kelurahan dalam menunjang kinerja organisasi
	Menyelenggarakan pelayanan umum kepada masyarakat sesuai peraturan yang berlaku
	Melaporkan pelaksanaan program dan kegiatan kepada atasan

2.1.9 Peta Kondisi Wilayah Kecamatan Mandalajati



Gambar 2. 3 Peta kondisi Wilayah Kecamatan Mandalajati

2.2 Landasan Teori

Landasan teori bertujuan sebagai acuan atau pedoman agar setiap materi yang digunakan sesuai dengan fakta yang ada, beberapa landasan teori yang akan dibahas diantaranya adalah sebagai berikut :

2.2.1 Pelaporan

Menurut Luther M. Gullick di buku *Papers on the Science of Administration* pelaporan adalah fungsi manajemen dalam menyampaikan hasil kegiatan mengenai hal yang berkaitan dengan tugas dan fungsi-fungsi [2].

2.2.2 Lingkungan

Lingkungan menurut Kamus Besar Bahasa Indonesia merupakan daerah. Arti lingkungan menurut Munadjat Danusaputro merupakan seluruh benda dan daya serta keadaan termasuk yang ada di dalamnya manusia dan segala tingkah perbuatannya yang berada dalam ruang dimana manusia memang berada dan mempengaruhi suatu kelangsungan hidup serta pada kesejahteraan manusia [3].

2.2.3 Sistem

Suatu sistem dapat didefinisikan dalam dua kelompok pendekatan. Pertama pendekatan sistem lebih menekankan pada prosedur. Menurut Jerry Fitz Gerald, “sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul Bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu”. Kedua pendekatan sistem lebih menekankan pada elemen atau komponennya mendefinisikan “sistem merupakan kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu”[4].

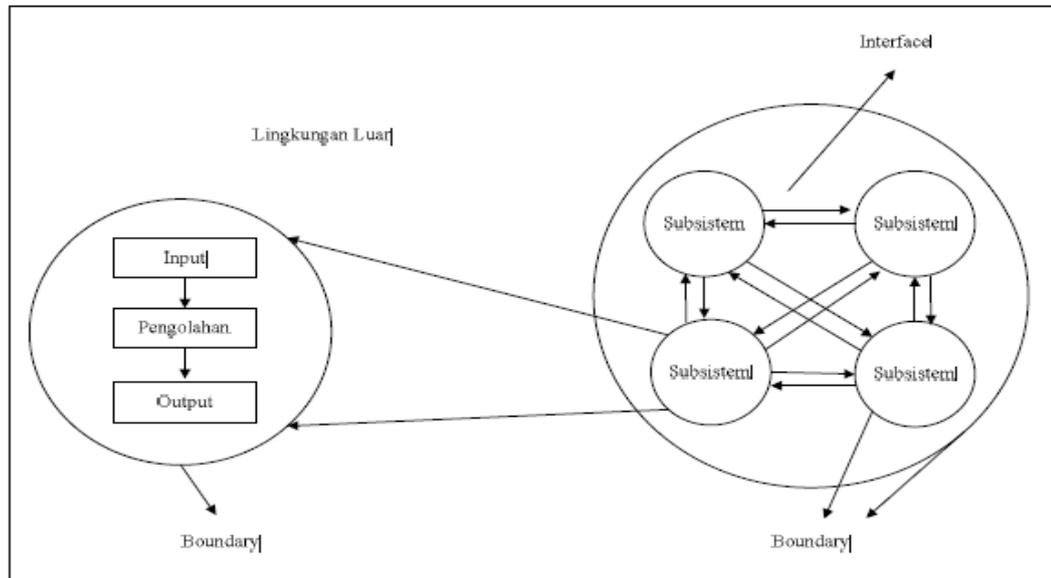
2.2.3.1 Karakteristik Sistem

Suatu sistem mempunyai sifat-sifat tertentu [4], yaitu :

1. Komponen
2. Batas Sistem
3. Lingkungan Luar Sistem
4. Penghubung Sistem
5. Masukan Sistem
6. Keluaran Sistem

7. Pengolah Sistem

8. Sasaran Sistem



Gambar 2. 4 Karakteristik Sistem

2.2.4 Informasi

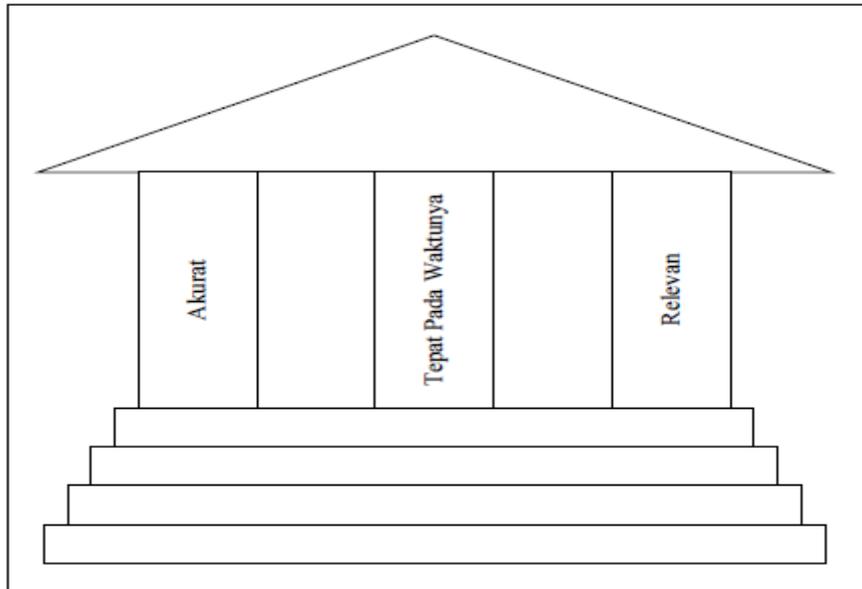
Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya. Sumber dari informasi adalah data, data merupakan bentuk jamak dari bentuk tunggal. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata [4]. Berdasarkan teori tersebut maka informasi dapat disimpulkan sebagai kumpulan dari data yang telah diolah atau diproses untuk menghasilkan suatu arti yang lebih berguna bagi yang menerimanya dan menggambarkan suatu kejadian dan kesatuan nyata untuk kelancaran manajemen untuk pengambilan keputusan.

2.2.4.1 Kualitas Informasi

Kualitas dari suatu informasi tergantung dari tiga hal [4], yaitu :

1. Akurat, berarti informasi harus bebas dari kesalahan-kesalahan dan harus jelas mencerminkan maksudnya.
2. Tepat pada waktunya, berarti informasi yang datang ke penerima tidak boleh terlambat. Informasi yang sudah usang tidak ada nilainya lagi. Karena informasi merupakan landasan dalam pengambilan keputusan.

3. Relevan, berarti informasi tersebut mempunyai manfaat untuk pemakainya.



Gambar 2. 5 Pilar Kualitas Informasi

2.2.5 Internet

Internet merupakan singkatan dari *Interconnecting Networking*. Internet ialah merupakan hubungan antara berbagai jenis komputer dan jaringan di dunia yang berbeda sistem operasi maupun aplikasinya dimana hubungan tersebut memanfaatkan kemajuan komunikasi (telepon dan satelit) yang menggunakan protocol standar dalam berkomunikasi yaitu protocol TCP/IP (*Transmission Control/Internet Protocol*) [5].

2.2.6 Web Server

Web Server adalah perangkat lunak yang berfungsi sebagai penerima permintaan yang dikirimkan melalui browser kemudian memberikan tanggapan permintaan dalam bentuk halaman situs web atau lebih umumnya dalam dokumen HTML. Namun, web server dapat mempunyai dua pengertian berbeda, yaitu sebagai bagian dari perangkat keras (*hardware*) maupun sebagai bagian dari perangkat lunak (*software*) [6].

1. Web Server bertugas menjadi tempat penyimpanan skrip, gambar, maupun konten halaman website. Web Server harus dapat diakses dari

seluruh wilayah yang mempunyai koneksi internet. Ada beberapa persyaratan yang bisa menjadi pertimbangan memilih penyedia layanan server hosting untuk website seperti sever harus cepat, mempunyai kapasitas penyimpanan yang besar, dan memiliki kapasitas RAM yang memadai.

2. Selain tempat penyimpanan, fungsi web server adalah sebagai perangkat lunak yang melayani permintaan dari browser. Ada banyak web server yang saat ini tersedia, salah satu Apache.

2.2.7 Web Service

Web Service adalah suatu sistem perangkat lunak yang dirancang untuk mendukung interoperabilitas dan interaksi antar sistem pada suatu jaringan. *Web Service* digunakan sebagai suatu fasilitas yang disediakan oleh suatu website untuk menyediakan layanan (dalam bentuk informasi) kepada sistem lain, sehingga sistem lain dapat berinteraksi dengan sistem tersebut melalui layanan-layanan (*service*) yang disediakan oleh suatu sistem yang menyediakan *web service*. *Web Service* menyimpan data informasi dalam format XML, sehingga data ini dapat diakses oleh sistem lain walaupun berbeda platform, sistem operasi, maupun bahasa *compiler* [8].

2.2.8 Google Maps

Google maps adalah aplikasi peta online gratis dari google. Google Maps dapat diakses melalui browser web atau melalui perangkat mobile. Anda dapat menggunakan Google Maps untuk mendapatkan arahan yang detail dari suatu lokasi, mencari informasi tentang bisnis lokal, dan banyak lagi. Setiap kali kita pergi ke suatu tempat untuk pertama kalinya, maka kita dapat menggunakan Google Maps untuk petunjuk arah. Aplikasi google maps juga lebih dari sekedar penunjuk jalan. Ada banyak informasi tentang tempat-tempat yang berbeda yang dapat dilihat langsung di Google Maps [8].

2.2.9 Global Positioning System (GPS)

GPS adalah singkatan dari *Global Position System*, yang merupakan sistem navigasi dengan menggunakan teknologi satelit yang dapat menerima sinyal dari satelit. Sistem ini menggunakan satelit yang mengirimkan sinyal gelombang mikro

ke bumi. Sinyal ini diterima oleh alat penerima (*receiver*) di permukaan dimana GPS *receiver* ini akan mengumpulkan informasi dari satelit GPS [9], seperti :

1. Waktu. GPS *receiver* menerima informasi waktu dari jam atom yang mempunyai keakurasian sangat tinggi.
2. Lokasi. GPS memberikan informasi lokasi dalam tiga dimensi yaitu Latitude, Longitude, dan Elevasi.
3. Kecepatan. Ketika berpindah tempat GPS dapat menunjukkan informasi kecepatan berpindah tersebut.
4. Arah perjalanan. GPS dapat menunjukkan arah tujuan.
5. Simpan lokasi. Tempat-tempat yang sudah pernah atau ingin dikunjungi bisa disimpan oleh GPS *receiver*.

2.2.10 Geofence

Geofence adalah merupakan fitur dalam program perangkat lunak yang digunakan bersamaan dengan GPS dalam menentukan batas-batas geografis atau parameter virtual dari suatu peta. Program yang menggunakan *geofence* dapat mengatur suatu triggers yang dapat memberikan informasi atau notifikasi apabila suatu target tertentu masuk atau keluar dari suatu batasan yang telah ditetapkan sebelumnya. Target yang melakukan trigger biasanya merupakan koordinat geografis dari suatu GPS yang dikirim secara reguler ke suatu pengelola data atau server [12].

Secara garis besar, koordinat geografis digunakan untuk mengetahui posisi target dan juga untuk membuat suatu batasan daerah tertentu (*mapping* sebagai pagar virtual (*geofence*) suatu daerah. Sistem akan menentukan posisi target yang dilacak berada di luar atau di dalam wilayah *geofence*. Teknologi ini juga dapat memungkinkan untuk pendekatan kedekatan antara posisi target dengan area *geofence* tertentu.

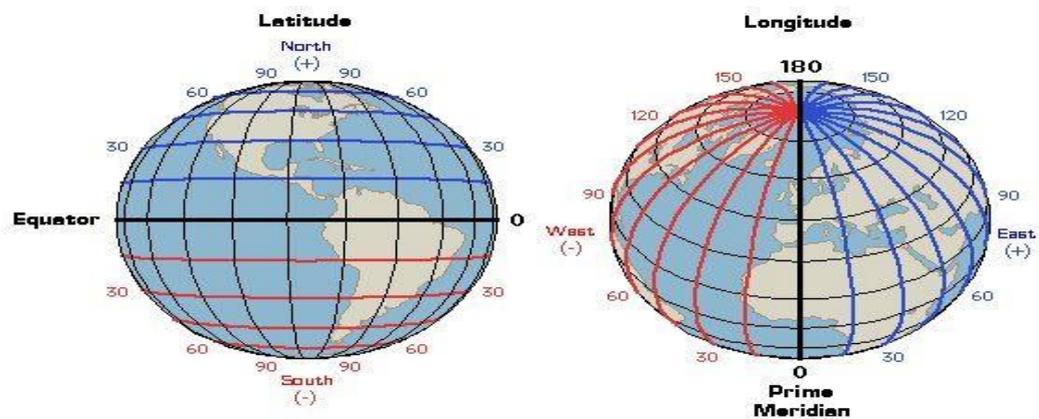
2.2.10.1 Geofence Area

Ukuran wilayah dari suatu *geofence* dapat berkisar dari beberapa puluhan meter sampai dengan beberapa kilometer. Bentuk daerah *geofence* sendiri bisa berbeda-beda, dari bentuk lingkaran, persegi atau persegi panjang atau lebih rumit

seperti suatu poligon. Koordinat dari karakteristik point atau titik yang terbentuk merupakan hal yang penting dalam mendefinisikan parameter *geofence*.

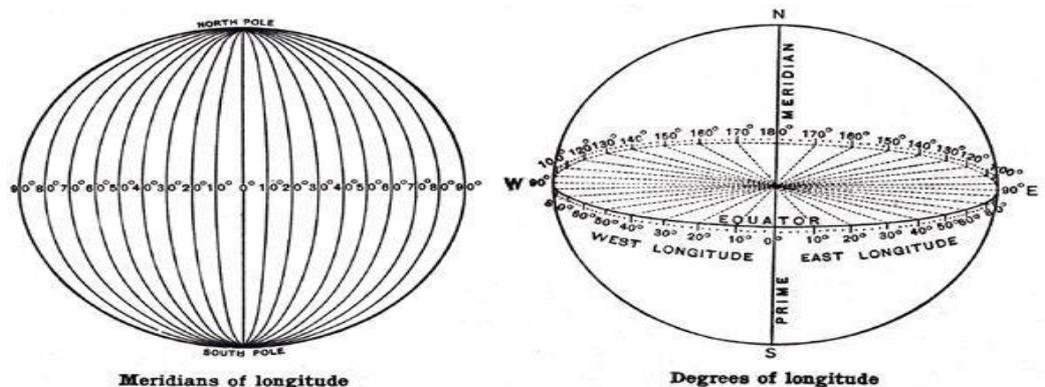
2.2.11 *Latitude dan Longitude*

1. Latitude adalah garis yang melintang dari kutub utara dan kutub selatan. Titik 0 adalah sudut ekuator, tanda + menunjukkan arah ke atas menuju kutub utara, sedangkan tanda minus di kordinat Latitude menuju ke kutub selatan. Titik yang dipakai dari 0 ke 90 derajat arah kutub utara, dan 0 ke -90 derajat ke kutub selatan [11].



Gambar 2. 6 Latitude

2. Longitude adalah garis lintang. Angka dari sudut bundar bumi horizontal. Titik diawali dari 0 ke 180 derajat dan 0 ke -180 ke arah sebaliknya. Titik 0 dimulai dari garis negara inggris. Mengarah ke Indonesia akan menjadi angka positif. Kebalikannya koordinat longitude minus adalah arah kebalikan [11].



Gambar 2. 7 Longitude

2.2.12 Location-Based Service (LBS)

Location Based Service (LBS) adalah layanan informasi yang dapat diakses menggunakan piranti *mobile* melalui jaringan internet dan seluler serta memanfaatkan kemampuan penunjukan lokasi pada piranti *mobile* [12].

2.2.12.1 Komponen Location-Based Service (LBS)

Terdapat empat komponen pendukung utama dalam teknologi *location based service* [13], antara lain :

1. Piranti *Mobile*
2. Jaringan Komunikasi
3. Komponen *Positioning* (Penunjukan posisi/lokasi)
4. Penyedia layanan dan aplikasi
5. Penyedia data dan konten

2.2.13 Android

Dikutip dari buku Android dengan PHP & MySQL karya Akhmad Dharma Kasman [14] Android merupakan sebuah sistem operasi telepon seluler dan komputer tablet layar sentuh (*touch screen*) yang berbasis Linux. Namun seiring perkembangannya, android berubah menjadi platform yang begitu cepat dalam melakukan inovasi. Hal ini tidak lepas dari pengembangan utama dibelakangnya, yaitu Google-lah yang mengakuisisi android, kemudian membuat sebuah platform. Platform android terdiri dari sistem operasi Linux, sebuah GUI (*Graphical User Interface*), sebuah web browser dan aplikasi End-User yang dapat di download dan juga para pengembang bisa dengan leluasa berkarya serta menciptakan aplikasi yang terbaik dan terbuka untuk digunakan oleh berbagai macam perangkat.

2.2.13.1 Arsitektur Aplikasi Berbasis Android

Google mengibaratkan android sebagai sebuah tumpukan software. Setiap lapisan dari tumpukan ini menghimpun beberapa program yang mendukung fungsi-fungsi spesifik dari sistem operasi. Berikut ini susunan dari lapisan-lapisan tersebut jika dilihat dari lapisan dasar hingga lapisan atas [14], lihat gambar dibawah ini.

2.2.13.2 Perkembangan Versi OS *Android*

Keunikan dari nama sistem operasi (OS) android adalah dengan menggunakan nama makanan hidangan penutup (Dessert). Selain itu juga nama-nama OS android memiliki huruf awal berurutan sesuai abjad. Namun juru bicara Google, Randall Sarafa enggan memberi tahu alasannya, Sarafa hanya menyatakan bahwa pemberian nama-nama itu merupakan hasil keputusan internal dan Google memilih tampil sedikit ajaib dalam hal ini.

Tabel 2. 5 Perkembangan Versi OS *Android*

Code Name	Version	Initial Release Date	API Level
No Codename	1.0	September 23, 2008	1
Petit Four (Only Internal User)	1.1	February 9, 2009	2
Cupcake	1.5	April 27, 2009	3
Donut	1.6	September 15, 2009	4
Éclair	2.0 – 2.1	October 26, 2009	5 – 7
Froyo	2.2 – 2.2.3	May 20, 2010	8
Gingerbeard	2.3 – 2.3.7	December 6, 2010	9 – 10
Honeycomb	3.0 – 3.2.6	February 22, 2011	11 – 13
Ice Cream Sandwich	4.0 – 4.0.4	October 31, 2011	14 – 15
Jelly Bean	4.1 – 4.3.1	July 9, 2012	16 – 18
KitKat	4.4 – 4.4.4	October 31, 2013	19 – 20
Lollipop	5.0 – 5.1.1	November 12, 2014	21 – 22
Marshmallow	6.0 – 6.0.1	October 5, 2015	23
Nougat	7.0	August 22, 2016	24
	7.1	October 4, 2016	25
	7.1.1 -	December 5, 2016	25
	7.1.2		
Oreo	8.0	August 21, 2017	26
	8.1	December 5, 2017	27
Pie	9.0	August 6, 2018	28

Android 10	10.0	September 3, 2019	29
------------	------	-------------------	----

2.2.14 Database

Dikutip di buku Simarmata dan Paryudi, ada beberapa pengertian *Database* sebagai berikut :

1. Menurut Stephens dan Plew (2000), adalah mekanisme yang digunakan untuk menyimpan informasi atau data.
2. Menurut Silberschatz, dkk (2002), mendefinisikan basisdata sebagai kumpulan data berisi informasi yang sesuai untuk sebuah perusahaan.
3. Menurut Ramakrishnan dan Gehrke (2003) menyatakan basisdata kumpulan data, umumnya mendeskripsikan aktivitas satu organisasi atau lebih yang berhubungan.
4. Menurut McLeod, dkk (2001), adalah kumpulan seluruh sumber daya berbasis computer milik organisasi.

2.2.14.1 Database Management System (DBMS)

Database Management Sistem (DBMS) adalah paket perangkat lunak yang dirancang untuk mendefinisikan, memanipulasi, mengambil dan mengelola data dalam database. Suatu DBMS umumnya memanipulasi data itu sendiri, format data, nama bidang, struktur catatan dan struktur file. Itu juga mendefinisikan aturan untuk memvalidasi dan memanipulasi data ini. DBMS mengurangi pengguna dari program pembingkai untuk pemeliharaan data. Bahasa Query generasi keempat, seperti SQL, digunakan Bersama dengan paket DBMS untuk berinteraksi dengan database. Beberapa contoh DBMS yaitu :

1. MySQL
2. SQL Server
3. Oracle
4. dBASE
5. FoxPro

2.2.15 Bahasa Pemrograman yang Digunakan

2.2.15.1 PHP

PHP pertama kali ditemukan pada 1995 oleh seorang, *Software Developer* bernama Rasmus Lerdorf. Ide awal PHP adalah ketika itu, Rasmus ingin mengetahui jumlah pengunjung yang membaca resume online-nya. *Script* yang dikembangkan baru dapat melakukan dua pekerjaan yakni merekam informasi dan menampilkan jumlah pengunjung dari suatu *website* dan sampai sekarang kedua tugas tersebut masih tetap populer digunakan oleh dunia web saat ini. Kemudian dari situ banyak orang di milis mendiskusikan script buatan Rasmus Lerdorf, hingga akhirnya Rasmus mulai membuat sebuah *tool/script* bernama *Personal Home Page* (PHP) [15]. Keunggulan PHP antara lain :

1. Gratis
2. *Cross Platform*
3. Mendukung banyak *database*
4. *On The Fly*

2.2.15.2 JAVA

Pada 1991, sekelompok insinyur Sun dipimpin oleh Patrick Naughton dan James Gosling ingin merancang bahasa computer untuk perangkat consumer seperti *cable-tv Box*. Dikarenakan perangkat tersebut tidak memiliki banyak memori, Bahasa harus berukuran kecil dan mengandung kode yang padat. Juga karena manufaktur-manufaktur berbeda memilih *processor* yang berbeda pula, maka Bahasa harus bebas dari manufaktur manapun. Proyek diberi nama kode “Green” [17].

Java telah mengakomodasi hampir seluruh fitur-fitur penting Bahasa-bahasa pemrograman yang ada semenjak perkembangan komputasi modern manusia :

1. Dari SIMULA, Bahasa pada tahun 65-an, Bahasa yang paling mempengaruhi JAVA sekaligus C++. Dari Bahasa ini diadopsi bentuk-bentuk dasar dari pemrograman berorientasi objek.

2. Dari LISP – Bahasa tahun 55-an. Diadopsi fasilitas *garbage collection*, serta kemampuan untuk meniru *generic list processing*, meski fasilitas ini jarang yang memanfaatkannya.
3. Dari Algol – Bahasa pada tahun 60-an, diambil struktur kendali yang dimilikinya.
4. Dari C++, diadopsi sintaks, sebagian semantics dan *exception handling*.
5. Dari Bahasa Ada, diambil *strongly type*, dan *exception handling*.
6. Dari Objective C, diambil fasilitas interface.
7. Dari Bahasa SmallTalk, diambil pendekatan *single root class hierarchie*, dimana *object* adalah satu kesatuan hirarki pewarisan.
8. Dari Bahasa Eiffel, fasilitas *assertion* yang mulai diterapkan di sebagian JDK 1.4.

2.2.16 Flowchart

Flowchart adalah penyajian yang sistematis tentang proses dan logika dari kegiatan penanganan informasi atau penggambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program. *Flowchart* menolong analis dan programmer untuk memecahkan masalah kedalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian.

2.2.17 Unified Model Language (UML)

Unified Modeling Language (UML) adalah Bahasa pemodelan standar pada rekayasa perangkat lunak. Dengan menggunakan UML akan berdampak kepada peningkatan produktifitas dan kualitas serta pengurangan biaya dan waktu, kerumitan arsitektur dalam pengembangan perangkat lunak bisa diatasi dengan menggambarkan cetak biru system tersebut [16].

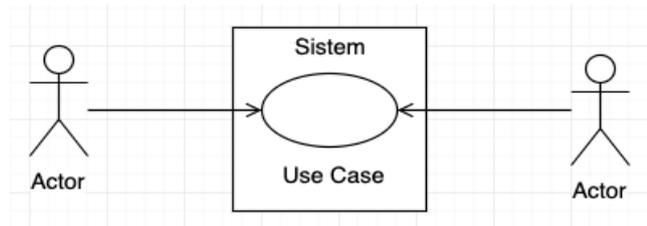
2.2.17.1 Diagram UML

UML menyediakan 4 macam diagram untuk memodelkan aplikasi perangkat lunak berorientasi objek [16], yaitu :

2.2.17.1.1 Use Case Diagram

Use Case Diagram digunakan untuk menangkap aspek dinamis dari system. Secara lebih spesifik use case diagram digunakan untuk mengumpulkan kebutuhan

dari sebuah system baik karena pengaruh internal maupun eksternal. Gambar dibawah ini menunjukkan *Use Case Diagram* dalam UML.

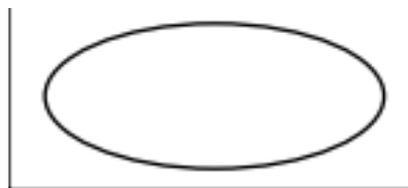


Gambar 2. 8 Use Case Diagram

Berikut ini adalah bagian dari sebuah *use case diagram* :

1. *Use Case*

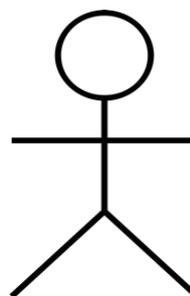
Use Case adalah deskripsi fungsi dari sebuah system dari perpektif pengguna. *Use case* bekerja dengan cara mendeskripsikan tipikal interaksi antar *user* (pengguna) sebuah system dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah system dipakai. Gambar dibawah ini menunjukkan bentuk *Use Case* dalam UML.



Gambar 2. 9 Use Case

2. *Actors*

Actor adalah *abstraction* dari orang dan system yang lain yang mengaktifkan fungsi dari target system. Orang atau system bisa muncul dalam beberapa [eran. Gambar dibawah ini menunjukkan *actor* dalam UML.



Gambar 2. 10 Actors

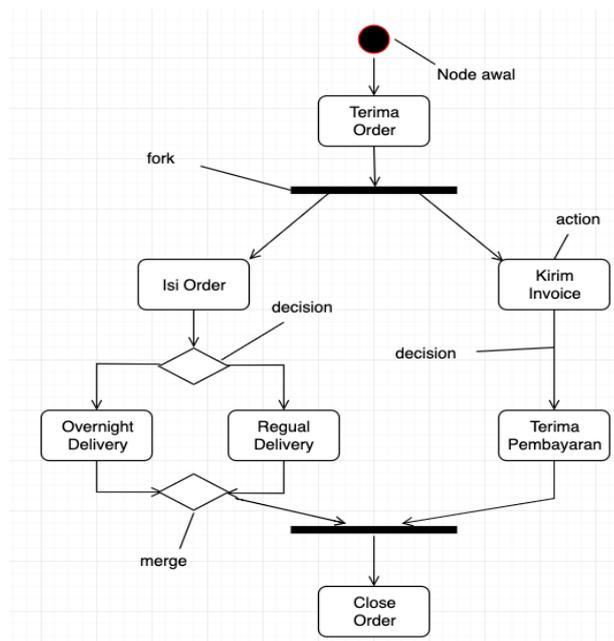
3. Relationship

Relationship adalah hubungan antar *use case* dengan *actor*. *Relationship* dalam use case diagram meliputi :

- a. Asosiasi antar *actor* dan *use case*
- b. Asosiasi antar 2 *use case*
- c. Generalisasi antara 2 *actor*
- d. Generalisasi antar 2 *use case*

2.2.17.1.2 Activity Diagram

Activity Diagram adalah bagian penting dari UML yang menggambarkan aspek dinamis dari system. Logika procedural, proses bisnis dan aliran kerja suatu bisnis bisa dengan mudah dideskripsikan dalam *activity diagram*. Gambar dibawah ini menunjukkan *Activity Diagram* dalam UML.



Gambar 2. 11 Activity Diagram

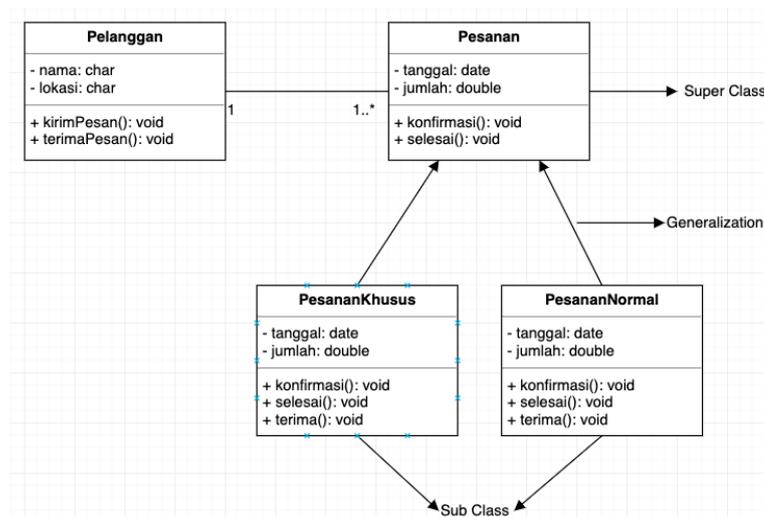
Berikut ini merupakan komponen dalam *activity diagram*, yaitu :

- a. *Activity Node* menggambarkan bentuk notasi dari beberapa proses yang beroperasi dalam control dan nilai data
- b. *Activity Edge* menggambarkan bentuk *edge* yang menghubungkan aliran aksi secara langsung dimana menghubungkan *input* dan *output* dari aksi tersebut

- c. *Initial State* adalah Bentuk lingkaran berisi penuh melambangkan awal dari suatu proses
- d. *Decision* adalah Bentuk wajib dengan suatu *flow* yang masuk beserta dua atau lebih *activity node* yang keluar. *Activity Node* yang keluar ditandai untuk menghasilkan beberapa kondisi
- e. *Fork* adalah Satu bar hitam dengan satu *activity node* yang masuk beserta dua atau lebih *activity node* yang keluar
- f. *Join* adalah Satu bar hitam dengan dua atau lebih *activity node* yang masuk beserta satu *activity node* yang keluar, tercatat pada akhir dari proses secara bersamaan. Semua *actions* yang menuju *join* harus sebelum proses dapat berlanjut.
- g. *Final State* adalah Bentuk lingkaran berisi penuh yang berada di dalam lingkaran kosong menunjukkan akhir dari suatu proses

2.2.17.1.3 Class Diagram

Class Diagram adalah diagram statis. Ini mewakili pandangan statis dari suatu aplikasi. *Class diagram* tidak hanya menggunakan untuk memvisualisasikan, menggambarkan dan mendokumentasikan berbagai aspek sistem tetapi juga membangun kode eksekusi dari aplikasi perangkat lunak. Gambar dibawah ini menunjukkan *class diagram* dalam UML.



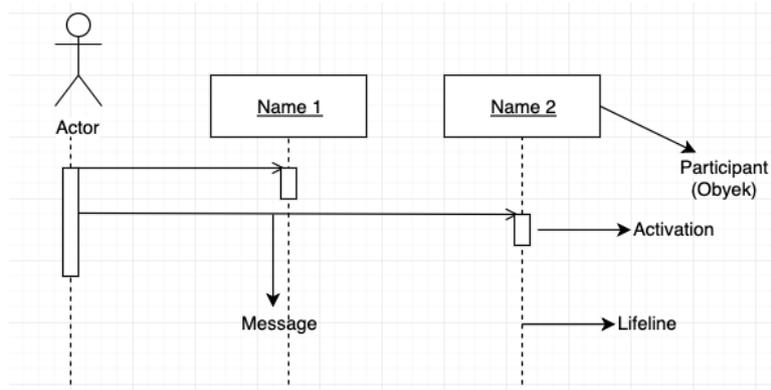
Gambar 2. 12 Class Diagram

Class diagram mempunyai 3 relasi dalam penggunaannya, yaitu :

- a. *Assosiation* adalah sebuah hubungan yang menunjukkan adanya interaksi antar *class*. Hubungan ini dapat ditunjukkan dengan garis dengan mata panah terbuka di ujungnya yang mengindikasikan adanya aliran pesa dalam satu arah.
- b. *Generalization* adalah sebuah hubungan antar *class* yang bersifat dari khusus ke umum
- c. *Constraint* adalah sebuah hubungan yang digunakan dalam system untuk memberi Batasan pada system sehingga didap aspek yang tidak fungsional.

2.2.17.1.4 *Sequence Diagram*

Sequence Diagram digunakan untuk menggambarkan perilaku pada sebuah *scenario*. Diagram ini menunjukkan sejumlah contoh objek-objek di dalam *use case*. Gambar dibawa ini menunjukkan *Sequence Diagram* dalam UML.



Gambar 2. 13 *Sequence Diagram*

Berikut merupakan komponen dalam *sequence diagram* :

- a. *Activation* menjelaskan tentang eksekusi dari fungsi yang dimiliki oleh suatu objek
- b. *Actor* menjelaskan tentang peran yang melakukan serangkaian aksi dalam suatu proses
- c. *Collaboration Boundary* menjelaskan tentang tempat untuk lingkungan percobaan dan digunakan untuk memonitor objek
- d. *Parallel Vertical Lines* menjelaskan tentang suatu garis proses yang menunjukkan pada suatu state

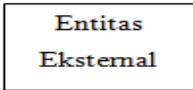
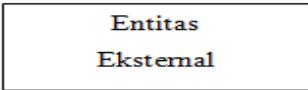
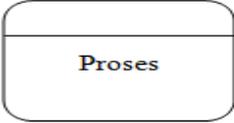
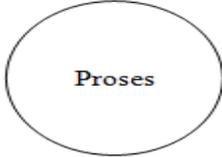
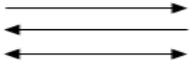
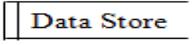
- e. *Processes* menjelaskan tentang tindakan/aksi yang dilakukan oleh aktor dalam suatu waktu
- f. *Window* menjelaskan tentang halaman yang sedang ditampilkan dalam suatu proses
- g. *Loop* menjelaskan tentang model logika yang berpotensi untuk diulang beberapa kali

2.2.18 Data Flow Diagram (DFD)

Data Flow Diagram atau DFD menurut Sukamto dan Shalahudin adalah representasi grafik yang menggambarkan aliran informasi dan transformasi yang diaplikasikan sebagai data yang mengalir dari masukan dan keluaran. DFD ini merupakan alat perancangan sistem yang berorientasi pada alur data dengan konsep dekomposisi dapat digunakan untuk penggambaran analisa maupun rancangan sistem yang mudah dikomunikasikan oleh profesional sistem kepada pemakai maupun pembuat program.

2.2.18.1 Simbol Data Flow Diagram

Data Flow Diagram (DFD) merupakan salah satu alat untuk mengetahui aliran data yang mengalir dalam sistem. DFD menggambarkan beberapa simbol sebagai berikut :

Gane/Sarson	Yourdon/De Marco
	
	
<p>Aliran Data</p> 	<p>Aliran Data</p> 
	

Gambar 2. 14 Simbol Data Flow Diagram

- a. *Process* adalah kegiatan atau pekerjaan yang dilakukan oleh semua orang, mesin, atau komputer dari arus data yang masuk ke proses dan akan dihasilkan arus data yang keluar dari proses. Arus data adalah arus yang mengalir dari proses atau arus yang mengalir dari proses menuju proses lain.
- b. *Data Store* digunakan untuk menunjukkan simpanan dari data yang dapat berupa suatu file atau database di sistem komputer, suatu arsip atau catatan manual
- c. *External Entity* dapat berupa orang, organisasi, departemen didalam organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan input atau menerima output dari sistem.
- d. *Data Flow* menunjukkan arah mengalirnya data. Arus data mengalir menuju proses dan atau meninggalkan data. Arus data mengalir menuju proses dan atau meninggalkan proses. Arus data ini dapat berupa masukan untuk sistem atau hasil dari proses sistem.

2.2.18.2 Tahapan Perencanaan *Data Flow Diagram*

Berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD :

1. Membuat *DFD Level 0* atau sering disebut juga *Context Diagram*
DFD Level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. *DFD Level 0* digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar
2. Membuat *DFD Level 1*
DFD Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan. *DFD Level 1* merupakan hasil *breakdown DFD Level 0* yang sebelumnya sudah dibuat
3. Membuat *DFD Level 2*
Modul-modul pada *DFD Level 1* dapat di *breakdown* menjadi *DFD Level 2*. Modul mana saja yang harus di *breakdown* lebih detail tergantung pada tingkat kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail

maka modul tersebut sudah tidak perlu di *breakdown* lagi. Untuk sebuah sistem, jumlah *DFD Level 2* sama dengan jumlah modul pada *DFD Level 1* yang di *breakdown*.

4. Membuat *DFD Level 3* dan seterusnya

DFD Level 3, 4, 5 dan seterusnya merupakan *breakdown* dari modul pada *DFD Level* di atasnya. *Breakdown* pada level 3, 4, 5 dan seterusnya aturannya sama persis dengan *DFD Level 1* atau *level 2*. Pada suatu diagram *DFD* sebaiknya jumlah modul tidak melebihi dari 20 buah. Jika lebih dari 20 buah modul, diagram akan terlihat rumit dan susah untuk dibaca sehingga menyebabkan sistem yang dikembangkan juga menjadi rumit

