BAB 2
TINJAUAN PUSTAKA

2.1 Profil Dinas Pemadam Kebakaran Kota Bandung

Dinas Kebakaran Dan Penanggulangan Bencana Kota Bandung yang beralamat di Jl. Sukabumi No.17, Kacapiring, Batununggal, Kota Bandung, Jawa Barat 40271. Aktivitas utama adalah menanggulangi bencana kebakaran dan bencana lainnya. Pada pembangunan aplikasi pelaporan bencana kebakaran Kota Bandung ini data didapat dari dinas kebakara dan penanggulangan bencana ebagai pendukung penelitian ini.

Pada Zaman Kota Bandung berstatus Gemente, penanggulangan kebakaran di integrasikan dengan pembangunan jalan-jalan utama, dibuat hidran yang sumber airnya dihubungkan dengan pipa air minum. Untuk memeperkuat daya semprot air dibantu dengan pompa beroda disebut " Aangang ". Saat Kota Bandung berubah status menjadi Kota Praja, Walikota Bandung pada waktu itu dijabat oleh Bapak R.Enoch dan dilanjutkan oleh R.Priatna Kusumah menyediakan Mobil Pemadam Kebakaran yang lebih dikenal oleh masyarakat dengan sebutan Branweer, pertama kalinya bermarkas di Jl. Raya Timur atau lebih dikenal dengan Jl. Ahmad Yani No.296 dimana para petugas dan karyawan pemadam kebakaran juga berdomisili di komplek tersebut[1].

Tanggal 2 Oktober 1962, Pemerintah Kota Praja Bandung membentuk Urusan Pemadam Kebakaran (UPK) yang dipimpin olah Bapak Adut bin Enur yang menginduk ke Dinas Teknik Penyehatan (DPT) yang saat ini dikenal dengan PDAM dipimpin oleh Bapak Ir.Achmad. Tahun 1972 Urusan Pemadam Kebakaran berubah status menjadi Barisan Pemadam Kebakaran (BPK) yang menginduk ke Dinas Pekerjaan Umum sampai dengan tahun 1979 dipimpin oleh Bapak A.Surachman, pada tahun yang sama BPK mengalami perubahan lagi menjadi di bawah Sub Direktorat Ketertiban Umum (TIBUM) Tahun 1980 sesuai dengan perda No.19 Tahun 1980 Tentang Stuktur Organisasi dan Tata Kerja Dinas Kebakaran, pengelolaan Kebakaran statusnya berubah lagi menjadi unit tersendiri yaitu Dinas Kebakaran. Pada era reformasi penyelaggaraan Pemerintahan Daerah

dalam Undang-undang No. 22 Tahun 1999, maka pengelolaan penanggulangan kebakaran mengalami lagi perubahan menjadi Dinas Pencegahan dan Penanggulangan Kebakaran Kota Bandung (DPPK Kota Bandung) sebagaimana diatur dalam Perda No.5 Tahun 2001 tanggal 7 Maret 2001.

Dari uraian singkat tersebut di atas, bahwa hadirnya sebuah Kota seperti Kota Bandung pada mulanya belum memperlihatkan suatu keterpaduan / keselarasan yang harmoinis antar berbagai sektor, khususnya pada sektor pengembangan pembangunan perkotaan dimana terlihat bidang penanganan penanggulangan bahaya kebakaran kurang dipacu perkembangannya sehingga pelayanan penanggulangan bahaya kebakaran masih belum memuaskan harapan semua pihak.

Sekretaris Dinas Pencegahan Penanggulangan Kebakaran (DPPK) Kota Bandung, Rachmat Hidayat mengakui jumlah personel yang dimiliki pihaknya saat ini belum ideal. Saat ini, DPPK Kota Bandung memiliki 192 personil. Idealnya, 350 orang personel lapangan.

2.1.1 Logo Dinas Pemadam Kebakaran Kota Bandung

Logo dari Dinas Pemadam Kebakaran Kota Bandung dapat dilihat pada Gambar 2.1.



Gambar 2. 1Logo Dinas Pemadam Kebakaran Kota Bandung

2.1.2 Visi dan Misi

Dinas Pencegahan dan Penanggulangan Kebakaran Kota Bandung mempunyai Visi dan Misi sebagai berikut :

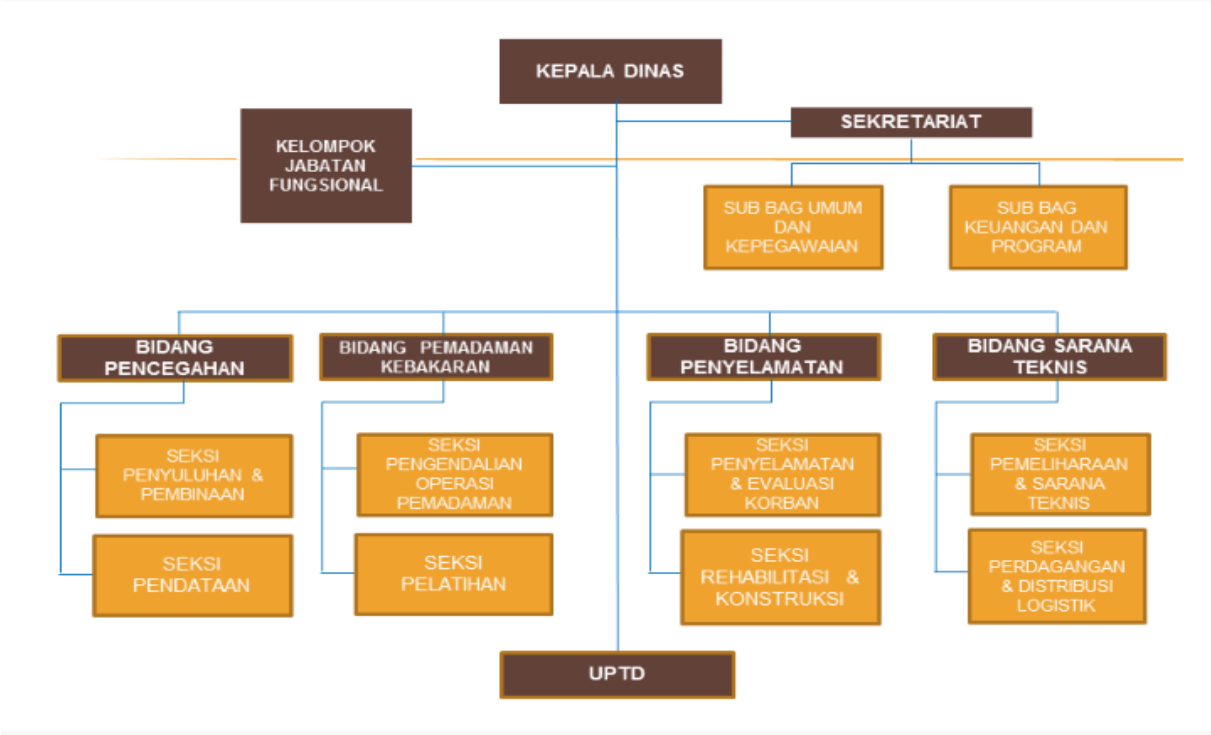
2.1.1.1 Visi Dinas Pemadam Kebakaran Kota Bandung

Mewujudkan dppk sebagai skpd yang responsif dan antisipatif terhadap upaya pencegahan, penanggulangan kebakaran dan bencana.

2.1.1.2 Misi Dinas Pemadam Kebakaran Kota Bandung

a. Meningkatkan intensitas penyuluhan terhadap bahaya kebakaran dan bencana lainnya kepada masyarakat.
b. Meningkatkan peran serta kepedulian masyarakat terhadap bahaya kebakaran.
c. Meningkatkan pengawasan dan pemeriksaan alat pemadam dan proteksi kebakaran baik aktif maupun pasif pada bangunan dan gedung
d. Mengembangkan pos wilayah pemadam kebakaran beserta sarana dan prasarana pencegahan, penanggulangan dan penyelamatan kebakaran dan bencana lainnya guna menunjang pelayanan masyarakat.
e. Terwujudnya waktu tanggap darurat (Response Time) dalam wilayah manajemen kebakaran (WMK).
f. Mengembangkan kualifikasi dan kompetensi kualitas sumber daya aparatur.

2.1.3 Struktur Organisasi



Gambar 2.2 Struktur Organisasi

2.2. Landasan Teori

Landasan teori menjelaskan beberapa teori-teori dan penjelasan yang berkaitan dengan aplikasi atau media yang akan dibangun. Landasan teori yang digunakan dalam penyususnan aplikasi pelaporan bencana kebakaran berbasis *Mobile* Android dengan memanfaatkan teknologi Geotagging, meliputi Profil Dinas Bencana Kebakaran, Pengertian Aplikasi, Pengertian Mobile, Android, Geotagging, GPS, LBS, Google Maps, Pengertian Java, JSON, UML,web service,MySQL,PHP.

2.1.2 Aplikasi

Aplikasi merupakan program yang berisikan perintah-perintah untuk melaksanakan pengolahan data, aplikasi secara umum adalah suatu proses dari cara manual yang ditransformasikan ke komputer dengan membuat sistem atau program agar data diolah lebih berdaya guna secara optimal. Aplikasi (application) juga

adalah software yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya Microsoft Word dan Microsoft Excel.

Dari definisi di atas dapat disimpulkan bahwa aplikasi adalah program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas tertentu untuk pengguna. Aplikasi yang akan dibangun dengan *platform mobile* dan sistem operasinya adalah *android,* dikarenakan saat ini android memiliki *market share* sebesar 87,7% dari seluruh smart phone yang diaktifkan. Berdasarkan dari uraian masalah yang dipaparkan diatas, maka aplikasi menjadi sebuah media yang akan dibangun dengan tujuan untuk memenuhi kebutuhan pelaporan bencana kebakaran.

2.2.1 Android



Gambar 2.3 Logo Android

Android adalah sebuah sistem operasi untuk perangkat mobile berbasis linux yang mencakup sistem operasi, *middleware*, dan aplikasi. Android adalah sistem operasi untuk telepon seluler yang berbasis *Linux*. Android menyediakan platform terbuka bagi para pengembang untuk membuat aplikasi mereka sendiri. Pada awalnya dikembangkan oleh Android Inc, sebuah perusahaan pendatang baru yang membuat perangkat lunak untuk ponsel yang kemudian dibeli oleh Google Inc. Untuk pengembangannya, dibentuklah *Open Handset Alliance* (OHA), konsorsium dari 34 perusahaan perangkat keras, perangkat lunak, dan telekomunikasi termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia [2].

Android 5.0 pertama kali diperkenalkan di bawah codename "Android L" pada 25 Juni 2014 selama presentasi keynote pada konferensi pengembang Google I / O. Di samping Lollipop, presentasi difokuskan pada sejumlah platform Android yang

berorientasi dan teknologi baru, termasuk Android TV, pada platform Android Auto, dpt dipakai pada platform komputasi Android Wear, dan platform pelacakan kesehatan Google Fit.

Android saat ini android memiliki *market share* sebesar 87,7% dari seluruh smart phone yang diaktifkan. Maka dari uraian diatas sistem operasi android digunakan untuk membangun suatu aplikasi pelaporan bencana kebakaran. Android yang digunakan adalah Android versi 5.0 (Lollipop), karena dengan banyaknya kelebihan dan fungsi yang bisa digunakan agar dapat sesuai dan maksimal dalam pembuatan sistem pelaporan bencana kebakaran.

2.2.1.1 Android SDK (Software Development Kit)

Android SDK adalah *tools* API (*Application Programming Interface*) yang diperlukan untuk mulai mengembangkan aplikasi pada *platform* Android menggunakan bahasa pemrograman Java. Android merupakan subset perangkat lunak untuk ponsel yang meliputi sistem operasi, *middleware* dan aplikasi kunci yang di-*release* oleh Google. Saat ini disediakan Android SDK (*Software Development Kit*) sebagai alat bantu dan API untuk mulai mengembangkan aplikasi pada *platform* Android menggunakan bahasa pemrograman Java.

2.2.1.2 ADT (Android Development Tools)

Android Develpoment Tools adalah plugin yang di desain untuk IDE Eclipse yang memberikan kemudahan dalam mengembangkan aplikasi android dengan menggunakan IDE Eclipse. Dengan menggunakan ADT untuk Eclipse akan memudahkan dalam membuat aplikasi *project* Android, membuat GUI aplikasi, dan menambahkan komponen-komponen yang lainnya.

2.2.1.3 Android Life Cycle

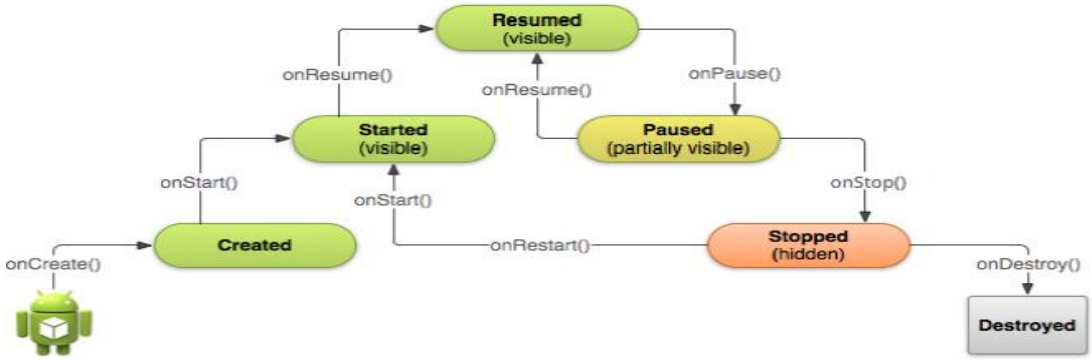
Aplikasi android terdiri dari beberapa fungsi dasar seperti mengedit catatan, memutar file musik, membunyikan alarm, atau membuka kontak telepon. Fungsi-fungsi tersebut dapat diklasifikasikan ke dalam empat komponen android yang berbeda seperti ditunjukkan pada , klasifikasi tersebut berdasarkan kelas- kelas dasar java yang digunakan**.**

Tabel 2. 1 Komponen Aplikasi Android

Functionality	Java Base Class	Examples
Focused thing a user can do	Activity	Edit a note, play a game
Background process	Service	Play music, update weather icon
Receive messages	BroadcastReceiver	Trigger alarm upon event
Store and retrieve data	ContentProvider	Open a phone contact

Setiap aplikasi pasti menggunakan minimal satu dari komponen tersebut, akan tetapi terdapat beberapa komponen yang mengharuskan mencantumkan specified permission sebelum digunakan seperti komponen Service, BroadcastReceiver. ContentProvider .

Android memiliki paradigma pemrograman lain tidak seperti paradigma pemrograman biasa di mana aplikasi yang dijalankan pada fungsi main(), sistem android menjalankan kode dalam method Activity dengan menerapkan metode callback tertentu yang sesuai dengan tahap tertentu dari siklus hidup. Setiap aplikasi yang berjalan dalam sistem operasi android memiliki siklus hidup yang berbeda dengan aplikasi desktop atau web. Hal ini dikarenakan aplikasi mobile memiliki tingkat interupsi proses yang lumayan tinggi seperti ketika handling panggilan masuk aplikasi diharuskan menghentikan proses sementara. Penerapan siklus hidup juga berguna untuk memastikan aplikasi tidak menghabiskan sumber daya baterai pengguna.



Gambar 2. 4 Siklus Hidup Android

Terdapat beberapa state dalam siklus hidup android yang terjadi seperti diilustrasikan pada Gambar 2.4 Siklus Hidup Android ,akan tetapi hanya beberapa dari state tersebut yang menjadi statis diantaranya :

1. Resumed

   Resumed terjadi ketika aplikasi berjalan setelah state paused . State ini akan menjalankan perintah program yang ditulis pada method onResume() (Google Inc,2014).
2. Paused

   Dalam keadaan ini aktivitas yang terjadi dihentikan secara sementara tetapi masih terlihat oleh pengguna karena terdapat proses yang memiliki prioritas lebih tinggi seperti panggilan telepon. Aplikasi tidak dapat menjalankan perintah apapun ataupun menampilkan apapun dalam state ini (Google Inc, 2014).
3. Stopped

Dalam keadaan ini, aplikasi benar-benar tidak ditampilkan dan tidak terlihatoleh pengguna tetapi masih meninggalkan service dibackground (Google Inc, 2014).

State lain seperti Created dan Started bersifat sementara dan sistem dengan cepat menjalankan state berikutnya dengan memanggil metode life cycle callback berikutnya. Artinya, setelah sistem OnCreate() dipanggil, dengan cepat sistem akan memanggil method OnStart(), krmudian diikuti oleh onResume() (Schwarz, Dutson, Stelee, & To, 2013).

2.2.1.4 Arsitektur Android

Secara garis besar Arsitektur Android dapat di jelaskan dan di gambarkan sebagai berikut :

1) *Application* dan *Widgets*

Application dan *Widgets* adalah *layer* dimana *user* berhubungan dengan aplikasi saja, dimana biasanya user men-*download* aplikasi, melakukan instalasi dan menjalankan aplikasi.

2) *Application Frameworks*

Android adalah "*Open Development Platform*" yaitu Android menawarkan kepada pengembang atau member kemampuan untuk membangun aplikasi yang inovatif.

3) *Libraries*

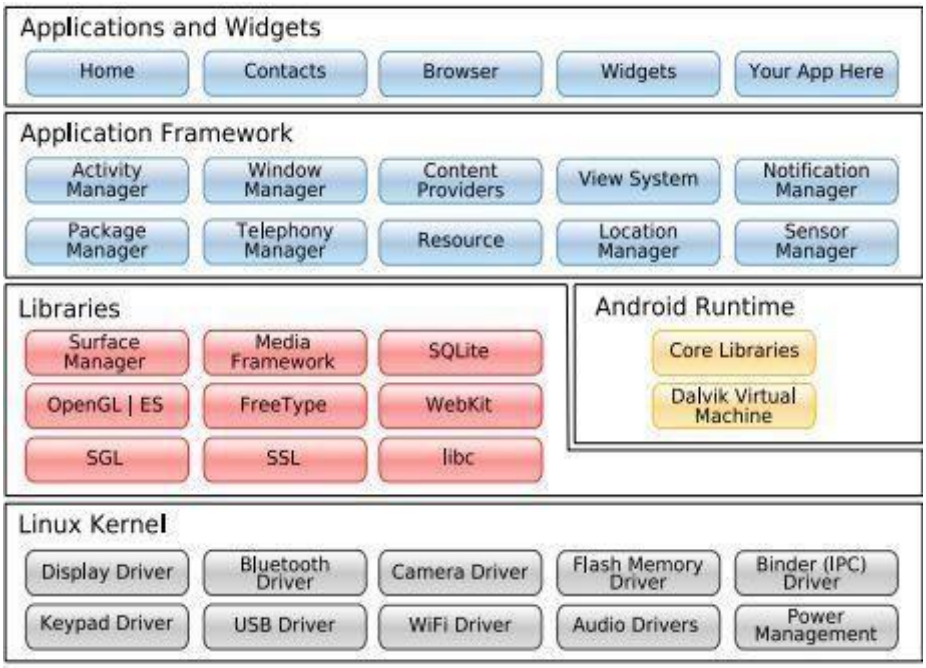
Libraries adalah *layer* dimana fitur-fitur Android berada, biasanya para pembuat aplikasi mengakses *libraries* untuk menjalankan aplikasinya.

4) Android *Runtime*

Layer yang membuat aplikasi Android dapat djalankan dimana dalam prosesnya menggunakan Implementasi Linux.

5) Linux Kernel

Linux Kernel adalah *layer* dimana inti dari *operating system* dari Android itu berada.



Gambar 2. 5 Arsitektur Android

2.2.2 Sistem

Sistem adalah sekelompok sistem yang terintegrasi dengan maksud yang sama utuk mencapai suatu tujuan. Untuk menjadi sebuah sistem, sistem memiliki beberapa sarat diantara sebagai berikut:1. Sistem harus dibentuk untuk menyelesaikan masalah. 2. Elemen sistem harus mempunyai rencana yang ditetapkan. 3. Adanya hubungan diantara elemen sistem. 4. Unsur dasar dari proses lebih penting dari pada elemen sistem. 5. Tujuan organisasi lebih penting dari pada tujuan elemen. Ada beberapa elemen yang membentuk sebuah sistem, yaitu: tujuan, masukan, proses, keluaran, batas, mekanisme pengendalian dan umpan balik serta lingkungan. Berikut penjelasan mengenai elemen-elemen yang membentuk sebuah sistem diantara lain sebagai berikut:

1. Tujuan (Goal)

Tujuan inilah yang menjadi pemotivasi yang mengarahkan sistem. Tanpa tujuan, sistem menjadi tak terarah dan tak terkendali. Tentu saja, tujuan antara satu sistem dengan sistem yang lain berbeda.

2. Masukan (Input)

Masukan(input) sistem yaitu segala sesuatu yang masuk ke dalam sistem dan selanjutnya menjadi bahan yang diproses. Masukan dapat berupa halhal yang berwujud atau tampak secara fisik maupun yang tidak tampak. Contoh masukan yang berwujud adalah bahan mentah, sedangkan contoh yang tidak berwujud adalah informasi, misalnya permintaan jasa pelanggan.

3. Proses

Proses merupakan bagian yang melakukan perubahan atau ransformasi dari masukan menjadi keluaran yang berguna dan lebih bernilai, misalnya berupa informasi dan produk, tetapi juga bisa berupa hal-hal yang tidak berguna, misalnya saja sisa pembuangan atau limbah. Pada pabrik kimia, proses dapat berupa bahan mentah. Pada rumah sakit, proses dapatberupa aktivitas pembedahan pasien.

4. Keluaran (output)

Keluaran (output) merupakan hasil dari pemrosesan. Pada sistemnformasi, keluaran bisa berupa suatu informasi, saran, cetakan laporan, dansebagainya

5. Batas(Boundary)

   Batas sistem adalah pemisah antara sistem dan daerah di luar sistem atau lingkungan. Batas sistem menentukan konfigurasi, ruang lingkup, atau kemampuan sistem. Sebagai contoh, tim sepak bola mempunyai aturan permainan dan keterbatasan kemampuan pemain.

6. Mekanisme Pengendalian dan Umpan Balik

   Mekanisme pengendalian (control mechanism) diwujudkan dengan menggunakan umpan balik (feedback). Umpan balik ini digunakan untuk mengendalikan baik masukan maupun proses. Tujuannya adalah untuk mengatur agar sistem berjalan sesuai dengan tujuan.

7. Lingkungan

   Lingkungan adalah segala sesuatu yang berada diluar sistem. Lingkungan bisa berpengaruh terhadap operasi sistem dalam arti bisa merugikan atau menguntungkan sistem itu sendiri. Lingkungan yang merugikan tentu saja harus ditahan dan dikendalikan agar tidak mengganggu kelangsungan operasi sistem, sedangkan yang menguntungkan tetap harus terus dijaga, karena akan memacu terhadap kelangsungan hidup system

Dari uraian diatas maka sistem adalah seperangkat unsur yang membantu untuk mencapai tujuan dalam membangun suatu media atau aplikasi yang akan kita buat.

2.2.3 Informasi

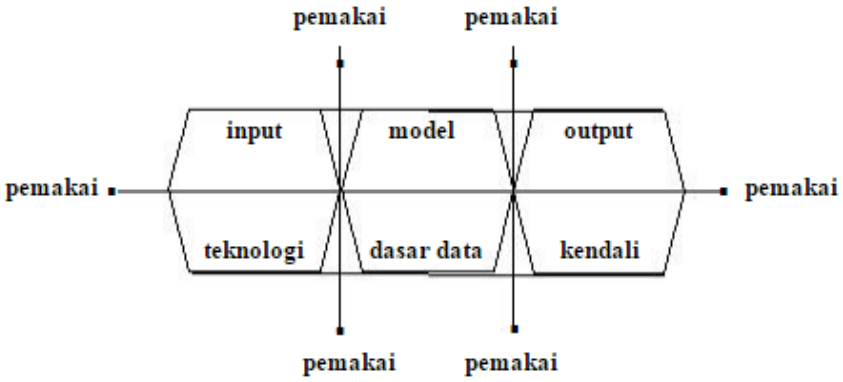
Informasi dapat didefinisikan “Informasi adalah data yang telah diolah menjadi suatu bentuk yang penting bagi si penerima dan mempunyai nilai yang nyata atau yang dapat dirasakan dalam keputusan-keputusan yang sekarang atau keputusan-keputusan yang akan datang. Jadi, secara umum informasi adalah data yang sudah diolah menjadi suatu bentuk lain yang lebih berguna yaitu pengetahuan atau keterangan yang ditujukan bagi penerima agar mendapatkan suatu info yang dapat bermanfaat, baik dimasa sekarang maupun dimasa yang akan datang.

Informasi yang dihasilkan dari aplikasi yang dibuat adalah informasi bencana kebakaran.

2.2.4 Sistem Informasi

Sistem informasi adalah sebuah sistem yang mempunyai fungsi mengumpulkan, memproses, menyimpan, menganalisis, dan menyebarkan informasi untuk tujuan yang spesifik dan berguna . Adapun Komponen-komponen

Sistem informasi terdiri dari komponen-komponen yang disebutnya dengan istilah blok bangunan (building block), yaitu blok masukan (input blocki), blok model (model block), blok keluaran (output block), blok teknologi (technology block), blok basis data (database block) dan blok kendali (controls block). Sebagai suatu sistem, keenam blok tersebut masing-masing saling berinteraksi satu dengan lainnya membentuk suatu kesatuan untuk mencapai sasarannya.



Gambar 2. 6 Blok Sistem Informasi

1. Blok Masukan

   Input mewakili data yang masuk ke dalam sistem informasi. Input disini termasuk metode-metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

2. Blok Model

Blok ini terdiri dari kombinasi prosedur, logika dan model matematika yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok Keluaran

   Produk dari sistem informasi adalah keluaran yang yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok Teknologi

   Teknologi merupakan "kotak alat" (tool-box) dalam sistem informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari 3 bagian utama, yaitu teknisi (humanware atau brainware), perangkat lunak (software) dan perangkat keras (hardware). Teknisi dapat berupa orang-orang yang mengetahui teknologi dan membuatnya dapat beroperasi. Misalnya teknisi adalah operator komputer, pemrogram, operator pengolah data, spesialis telekomunikasi, analis sistem, penyimpanan data dan lain sebagainya.

5. Blok Basis Data

   Basis data (database) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan di dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa, supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanan. Basis data diakses atau dimanipulasi dengan menggunakan perangkat lunak paket yang disebut dengan DBMS (Database management System).

6. Blok Kendali

   Banyak hal yang dapat merusak sistem informasi, seperti halnya bencana alam, api, temperatur, air, debu, kecurangan-kecurangan, kegagalan-kegagalan sistem itu sendiri, kesalahan-kesalahan, ketidak efisienan, sabotase

dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat cepat langsung diatasi.

2.2.5 Global Position System (GPS)

GPS merupakan sebuah infrastruktur satelit yang melayani penempatan posisi dari berbagai macam objek(Speikerman 2004). GPS pertama kali digunakan untuk kepentingan militer, tetapi pada tahun 1980-an pemerintah Amerikat Serikat memutuskan untuk membuat sistem *positioning* secara bebas dan dan tersedia untuk berbagai macam industry di dunia. Menurut (Roth 2004) sistem GPS terdiri atas 3 segmen yaitu[7]:

1. ***User Segment*** yang terdiri atas perangkat bergerak dari pengguna (GPS *receivers).*
2. ***Space Segment*** yang terdiri atas satelit. Setiap satelit mempunyai berat anatra 1.5 sampai 2 ton dan mempunyai energy yang swatantra yang disuplai oleh sel matahari.
3. ***Control segment*** administrasi yang dibutuhkan oleh satelit sebagai koreksi dari internet data satelit (sistem waktu dan orbit).

GPS dapat melakukan perhitungan dan menentukan posisi user dan menampilkan dalam maps. Jika sudah dapat menyimpan posisi user selanjutnya GPS dapat menghitung informasi lain, seperti kecepatan, arah tuju, rute, tujuan perjalanan serta jarak tujuan. GPS ini juga dimanfaatkan untuk membangun sebuah aplikasi pelaporan bencana kebakaran dimana si pelapor mengirimkan sebuah lokasi dan perhitungan waktu yang akurat untuk menentukan informasi lokasi pengguna.

2.2.9.1 Akurasi GPS

Posisi yang ditunjukkan oleh suatu GPS mempunyai faktor kesalahan atau juga disebut tingkat akurasi. Sebagai contoh suatu alat GPS menunjukkan titik koordinat dengan tingkat akurasi 5 meter, itu berarti posisi pengguna bisa berada dalam range radius 5 meter dari titik yang ditunjukkan tersebut. Mengapa tingkat akurasi yang terlihat bisa berubah-ubah? Kadang terlihat 10 meter, 15 meter, atau 5 meter. Ada beberapa hal yang mempengaruhi tingkat akurasi tersebut, antara lain:

1. Kesalahan Ephemeris. Terjadi jika satelit tidak dapat mentransmisikan posisinya di orbit dengan tepat.
2. Keadaan Ionosphere. Ionosphere berada pada jarak sekitar 43-50 mil di atas permukaan bumi.Walaupun GPS receiver berusaha untuk mengkoreksi/memperbaiki faktor keterlambatan yang terjadi tetap saja aktivitas tertentu dari plasma bisa menyebabkan kesalahan perhitungan.
3. Keadaan Troposphere. Troposphere adalah bagian terendah dari atmosfer sampai dengan ketinggian sekitar 11 mil dari permukaan tanah. Variasi pada temperatur, tekanan, dan kelembaban bisa menyebabkan perbedaan kecepatan penerimaan gelombang radio.
4. Kesalahan Waktu. Karena penempatan jam atom pada setiap GPS receiver tidak berjalan sebagaimana mestinya. Kesalahan waktu dari GPS receiver yang tidak presisi dapat menimbulkan ketidakakurasian.
5. Kesalahan Multipath. Terjadi karena sinyal satelit membentur permukaan keras (seperti bangunan atau tebing) sebelum mencapai GPS receiver. Hal tersebut bisa menyebabkan terjadinya delay sehingga perhitungan jarak menjadi tidak akurat.
6. Buruknya Sinyal Satelit. Keadaan langit yang terhalang akan menyebabkan GPS sulit menerima data satelit. Sebuah sinyal satelit yang pada hari tertentu diterima dengan sangat bagus belum tentu pada hari lain bisa diterima dengan kualitas yang sama walaupun user berdiri pada tempat yang sama.

2.2.9.2 Android dan GPS

Dewasa ini, teknologi berkembang dengan pesat. Dulu ponsel hanya sekedar digunakan untuk menelpon dan SMS saja. Sekarang ponsel sudah menjelma menjadi kotak kecil ajaib yang serba bisa. Salah satunya adalah ponsel dengan sistem operasi Android. Dengan standarisasi fitur dan hardware yang dimiliki, menjadikan ponsel Android ponsel canggih yang disukai banyak orang. Tidak lagi canggih karena adanya fitur MMS, radio, atau internet berkecepatan tinggi tapi juga karena ditanamkannya fitur teknologi satelit di dalamnya. Ya, perangkat GPS *receiver* yang dulu besar dan eksklusif, sekarang sudah bisa dimiliki dengan "hanya" membeli sebuah ponsel.

Dengan ponsel berteknologi satelit (GPS), banyak hal yang bisa dilakukan. Ingin melihat di mana posisi user sekarang dalam sebuah peta? Mengambil foto/video yang sudah dilengkapi dengan data koordinat? Ingin tahu jalur olahraga bersepeda yang sudah pernah user lalui? Tidak hanya itu, user juga dapat pergi ke tempat wisata tertentu dengan dipandu gambar dan suara dari sebuah ponsel! Bahkan lebih jauh lagi, GPS dapat digunakan untuk membantu memberikan peringatan dini terhadap terjadinya bencana alam. Sekarang ini banyak sekali pengembang - pengembang aplikasi untuk sistem operasi Android termasuk aplikasi-aplikasi GPS. Yang menyenangkan aplikasi-aplikasi tersebut jenisnya beragam dan jumlahnya pun banyak.

2.2.6 Location *Based service* (LBS)

LBS adalah layanan yang menyediakan informasi mengenai suatu tempat, dapat diakses dengan perangkat *mobile* melalui jaringan selular dan memiliki kemampuan untuk menggunakan posisi geografis dari perangkat *mobile* (Pardede, 2012) [5].

Location Based Service (LBS) atau Layanan Berbasis lokasi adalah layanan informasi yang dapat diakses melalui mobile device dengan menggunakan mobile network yang dilengkapi kemampuan untuk memanfaatkan lokasi dari mobile device tersebut. LBS memberikan kemungkinan komunikasi dan interaksi dua arah. Oleh karena itu pengguna memberitahu penyedia layanan untuk memberi

informasi,dengan referensi posisi pengguna tersebut. Layanan berbasis lokasi dapat digambarkan sebagaisuatu layanan yang berada pada pertemuan tiga teknologi yaitu: Geographic Information System, Internet Service, dan Mobile Devices, hal ini dapat dilihat pada gambar LBS adalah pertemuan dari tiga teknologi. Secara Garis besar jenis Layanan Berbasis Lokasi juga dapat dibagi menjadi dua, yaitu:

1. ***Pull Service***: Layanan diberikan berdasarkan permintaan dari pelanggan akan kebutuhan suatu informasi. Jenis layanan ini dapat dianalogikan seperti menggakses suatu web pada jaringan internet.
2. ***Push Service***: Layanan ini diberikan langsung oleh sevice provider tanpa menunggu permintaan dari pelanggan, tentu saja informasi yang diberikan tetap berkaitan dengan kebutuhan pelanggan.

   Teknologi ini dipakai dalam pembangunan aplikasi mobile pelaporan bencana kebakaran, untuk menentukan lokasi geografis tempat terjadinya bencana kebakaran, dan dapat mengetahui posisi pengguna.

2.2.7 UML

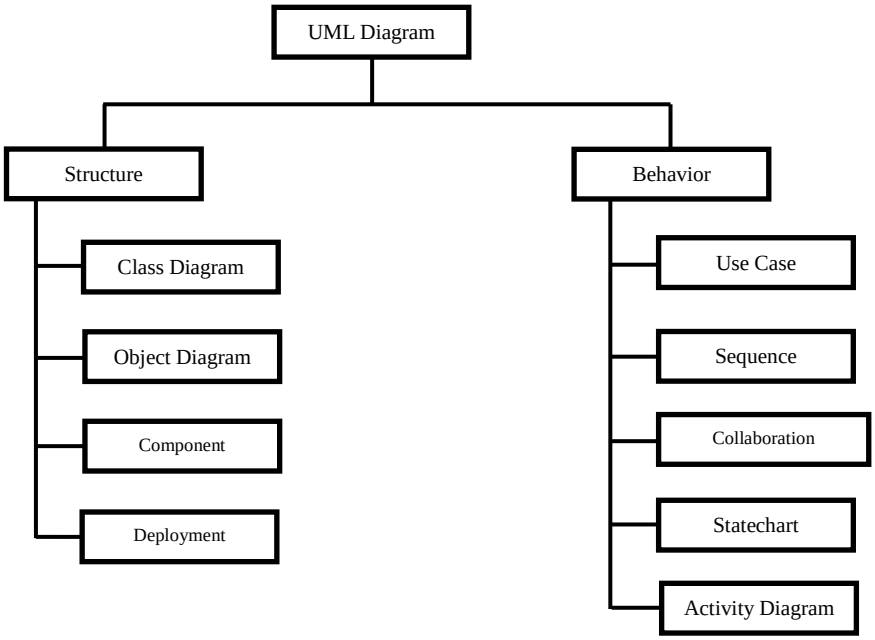
UML adalah alat perancangan sistemyang berorientasi pada objek. UML diagram memiliki tujuan untuk membantu tim pembangunan proyek berkomunikasi, mengeksplorasi potensi desain, dan memvalidasi desain arsitektur perangkat lunak atau pembuat program.

Diagram UML diklasifikasikan menjadi dua kategori; struktur dan perilaku. Diagram struktur untuk memvisualisasikan, menspesifikasikan, membangun dan mendokumentasikan aspek statik dari sistem. Diagram struktur yang akan dibahas dalam penulisan ini adalah class diagram, sedangkan diagram perilaku untuk memvisualisasikan, menspesifikasikan, membangun dan mendokumentasikan aspek dinamis dari sistem. Diagram perilaku yang akan dibahas pada penulisan ini adalah use case diagram, sequence diagram dan *activity diagram*.

UML menjadi metode yang digunakan untuk membangun sistem adalah OOAD(Object Oriented Analysis dan desaign)[13].

2.2.7.1 UML Diagram

Pada UML terdiri dari 13 macam diagram yang dikelompokan dalam 2 kategori. Pembagian kategori dan macam-macam diagram tersebut dapat dilihat pada gambar II.7



Gambar 2. 7 UML *Diagram*

Berikut ini penjelasan singkat dari pembagian kategori tersebut :

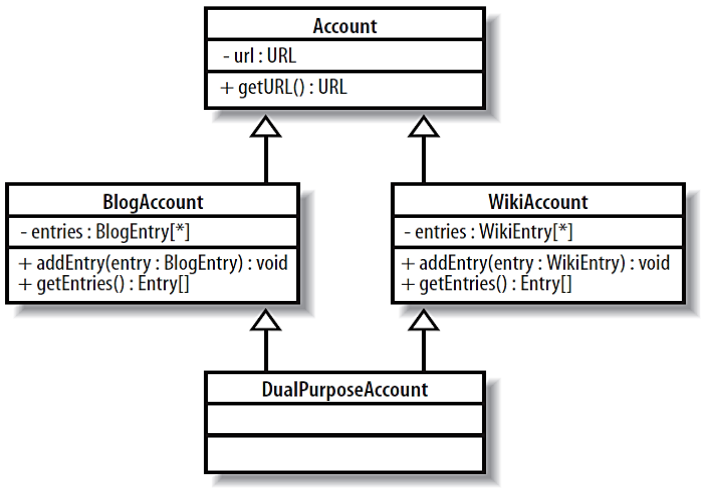
1. *Structure diagrams* yaitu diagram untuk memvisualisasikan, mendokumentasi, menspesifikasikan, dan membangun aspek statis dari sistem yang dimodelkan.
2. *Behavior diagrams* yaitu diagram untuk memvisualisasikan, mendokumentasi, menspesifikasikan, dan membangun aspek dinamis dari sistem yang dimodelkan.

2.2.7.2 Class Diagram

Diagram Kelas menunjukkan sekumpulan kelas, interface dan kolaborasi dan keterhubungan. Diagram kelas ditujukan untuk pandangan statik terhadap sistem. Kelas memiliki apa yang disebut nama, atribut, metode atau operasi.

1. Nama kelas harus unik. Nama akan menjadi identifier di program, seharusnya sedini mungkin dipilih nama yang memenuhi aturan (semua) bahasa pemrograman.
2. Atribut adalah properti bernama di kelas yang mendeskripsikan *range* nilai yang dipunyai instan kelas. Kelas dapat mempunyai sejumlah atribut atau tidak sama sekali.
3. Operasi atau metode adalah implementasi layanan yang dapat diminta pada sembarang objek kelas untuk mempengaruhi perilaku sistem. Kelas dapat mempunyai sejumlah operasi atau tidak sama sekali.

Berikut Contoh *Class Diagram* dapat dilihat pada Gambar II.8.



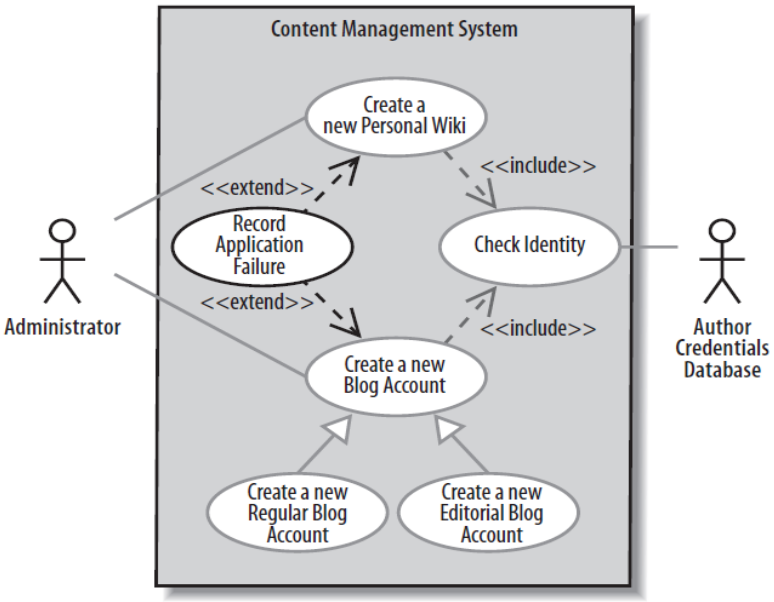
Gambar 2. 8 Contoh *Class Diagram*

2.2.7.3 Use Case Diagram

Use case Diagram merupakan salah satu diagram untuk memodelkan aspek perilakau sistem. Masing-masing diagram *use case* menunjukkan sekumpulan *use case*, aktor, dan hubungannya. Diagram *use case* adalah penting untuk memvisualisasikan, menspesifikasikan, dan mendokumentasikan kebutuhan perilaku sistem. Diagram-diagram *use case* merupakan pusat pemodelan perilaku sistem, subsistem, dan kelas. Diagram *use case* digunakan untuk mendeskripsikan apa yang seharusnya dilakukan oleh sistem. Ada empat hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case* [13].

1. Sistem yaitu sesuatu yang hendak kita bangun.
2. Relasi adalah relasi antara aktor dengan *use case*
3. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
4. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

Berikut Contoh *Use Case Diagram* dapat dilihat pada Gambar II.9.



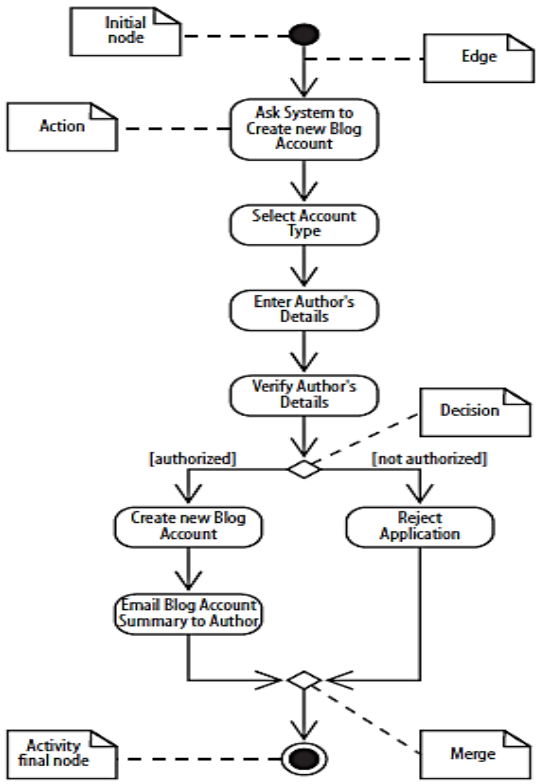
Gambar 2. 9 Contoh *Use Case Diagram*

2.2.7.4 Activity Diagram

Diagram aktivitas adalah diagram flowchart yang diperluas yang menunjukan aliran kendali satu aktivitas ke aktivitas lain di sistem. Diagram aktivitas ini digunakan untuk memodelkan aspek dinamis sistem. Diagram aktivitas mendeskripsikan aksi-aksi dan hasilnya. Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut :

1. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
2. Urutan atau pengelompokkan tampilan dari sistem/*user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
4. Rancangan menu yang ditampilkan pada perangkat lunak.

Berikut Contoh *Activity Diagram* dapat dilihat pada Gambar II.10.



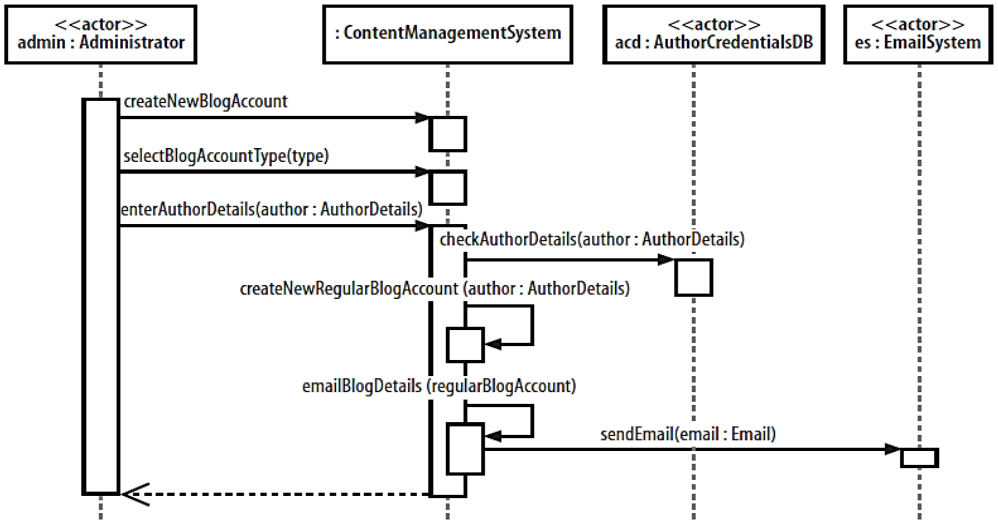
Gambar 2. 10 Contoh *Activity Diagram*

2.2.7.5 Sequence Diagram

Diagram sekuen menunjukan interaksi yang terjadi antar objek. Diagram ini merupakan pandangan dinamis terhadap sistem. Diagram ini menekankan pada sisi basis keberurutan waktu dari pesan-pesan yang terjadi. Diagram sekuen menunjukan objek sebagai garis vertikal dan tiap kejadian sebagai panah horisontal dari objek pengirim ke objek penerima. Waktu berlalu dari atas ke bawah dengan

lama waktu tidak relevan. Diagram ini hanya menunjukan barisan kejadian, bukan perwaktuan nyata. Kecuali untuk sistem waktu nyata yang mengharuskan konstrain barisan terjadi.

Berikut Contoh *sequence diagram* dapat dilihat pada gambar II.11

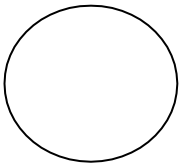
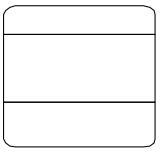
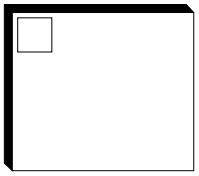
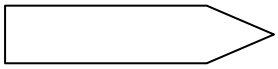


Gambar 2. 11 Contoh *Sequence Diagram*

2.2.8 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) adalah alat pembuatan model yang memungkinkan profesional sistem untuk menggambarkan sistem sebagai suatu jaringan proses fungsional yang dihubungkan satu sama lain dengan alur data, baik secara manual maupun komputerisasi. DFD ini sering disebut juga dengan nama Bubble chart, Bubble diagram, model proses, diagram alur kerja, atau model fungsi.

DFD menggambarkan komponen-komponen sebuah sistem, aliran-aliran data diantara komponen-komponen tersebut, asal, tujuan dan penyimpanan dari data tersebut. Gambar dibawah ini menunjukkan simbol yang digunakan dalam DFD baik dalam versi E.Yourdan dan De Marco maupun versi Chris Gane dan Trish Sarson[14].

	Yourdan	C.Gane dan T.Sarson
Aliran data / Data flow		
Proses / Process		
Simpanan data / Data store		
Kesatuan luar, batas sistem / External entity, boundary		
Aliran phisik / Material flow		

Gambar 2. 12 Komponen-komponen DFD

2.2.13 Pengertian Geotagging

Geotagging adalah gabungan fitur kamera dengan GPS (*Global Positioning Sytem*) guna memberikan informasi secara *relatime* dimana dan bagaimana kondisi sebuah objek. Ponsel yang dilengkapai fasilitas *geotagging* dapat digunakan untuk menghasilkan foto yang menyimpan informasi posisi data GPS, seperti garis lintang dan bujur, ketinggian, bantalan, jarak, akurasi data, dan nama tempat. Banyak ponsel yang dilengkapi dengan fitur GPS dapat menambahkan foto lokasi ke dalam metadata foto, atau biasa disebut dengan data EXIF, secara otomatis sehingga lokasi foto tersebut dapat ditampilkan ke dalam peta[7].

Pada pembangunan aplikasi pelaporan bencana kebakaran ini menggunakan teknologi geotagging, dimana pengguna dapat mengirimkan foto sebuah kejadian

kebakaran yang sedang terjadi dan difoto tersebut akan mengeluarkan sebuah informasi letak dan posisi pengguna.

2.2.14 Google Maps

Google Maps adalah peta online atau membuka peta secara online, dapat dilakukan secara mudah melalui layanan gratis dari *Google*. Bahkan layanan ini menyediakan API (*Application Programming Interface*) yang memungkinkan developer lain untuk memanfaatkan aplikasi ini di aplikasi buatannya. Tampilan *Google Maps* pun dapat dipilih, berdasarkan foto asli atau peta gambar rute saja.

GoogleMaps adalah layanan gratis yang diberikan oleh *Google* dan sangat popular. *GoogleMaps* adalah suatu peta dunia yang dapat kita gunakan untuk melihat suatu daerah seperti pada gambar 2. Dengan kata lain, *GoogleMaps* merupakan suatu peta yang dapat dilihat dengan menggunakan suatu browser. Kita dapat menambahkan fitur *GoogleMaps* dalam *web* yang telah kita buat atau pada blog kita yang berbayar maupun gratis sekalipun dengan *GoogleMaps* API. *GoogleMaps* API adalah suatu library yang berbentuk *JavaScript* [15].

Cara membuat *GoogleMaps* untuk ditampilkan pada suatu web atau blog sangat mudah hanya dengan membutuhkan pengetahuan mengenai HTML serta *JavaScript*, serta koneksi Internet yang sangat stabil. Dengan menggunakan *GoogleMaps* API, kita dapat menghemat waktu dan biaya untuk membangun aplikasi peta digital yang handal, sehingga kita dapat fokus hanya pada data-data yang akan ditampilkan. Dengan kata lain, kita hanya membuat suatu data sedangkan peta yang akan ditampilkan adalah milik *Google* sehingga kita tidak dipusingkan dengan mambuat peta suatu lokasi, bahkan dunia.



Gambar 2. 13 *Google Maps*

Pada GoogleMaps API terdapat 4 jenis pilihan model peta yang disediakan oleh Google, diantaranya adalah:

1. ROADMAP, untuk menampilkan peta biasa 2 dimensi.
2. SATELLITE, untuk menampilkan foto satelit.
3. TERRAIN, untuk menunjukkan relief fisik permukaan bumi dan menunjukkan seberapa tingginya suatu lokasi, contohnya akan menunjukkan gunung dan sungai.
4. HYBRID, akan menunjukkan foto satelit yang diatasnya tergambar pula apa yang tampil pada ROADMAP (jalan dan nama kota).

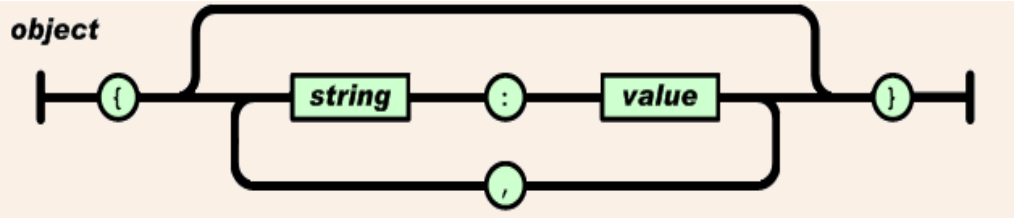
2.2.15 JavaScript Object Notation(JSON)

JSON (*JavaScript Object Notation*) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (generate) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemprograman JavaScript, Standar ECMA-262 Edisi ke-3 - Desember 1999. JSON merupakan format teks yang tidak bergantung pada bahasa pemprograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran-data.JSON terbuat dari dua struktur [9] :

1. Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (object), rekaman (record), struktur (struct), kamus (dictionary), tabel hash (hash table), daftar berkunci (keyed list), atau associative array.
2. Daftar nilai terurutkan (an ordered list of values). Pada kebanyakan bahasa, hal ini dinyatakan sebagai larik (array), vektor (vector), daftar (list), atau urutan (sequence).

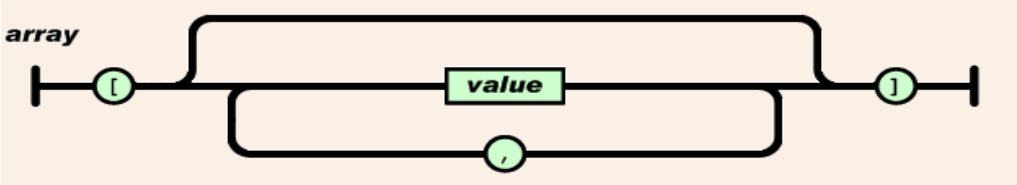
Struktur-struktur data ini disebut sebagai struktur data universal. Pada dasarnya, semua bahasa pemprograman moderen mendukung struktur data ini dalam bentuk yang sama maupun berlainan. Hal ini pantas disebut demikian karena format data mudah dipertukarkan dengan bahasa-bahasa pemprograman yang juga berdasarkan pada struktur data ini. JSON menggunakan bentuk sebagai berikut[16]:

Objek adalah sepasang nama/nilai yang tidak terurutkan. Objek dimulai dengan { (kurung kurawal buka) dan diakhiri dengan } (kurung kurawal tutup). Setiap nama diikuti dengan : (titik dua) dan setiap pasangan nama/nilai dipisahkan oleh , (koma).



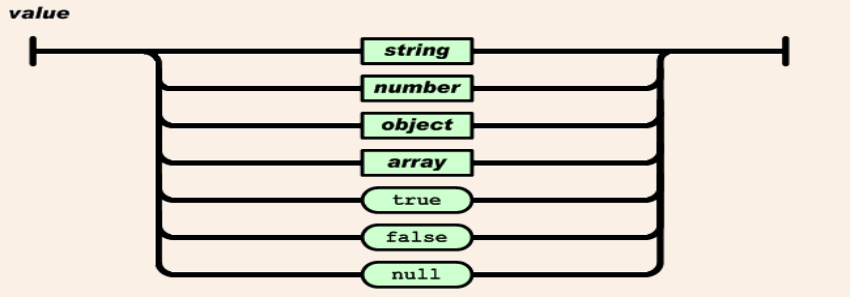
Gambar 2. 14 Objel JSON

Larik adalah kumpulan nilai yang terurutkan. Larik dimulai dengan [(kurung kotak buka) dan diakhiri dengan] (kurung kotak tutup). Setiap nilai dipisahkan oleh , (koma).



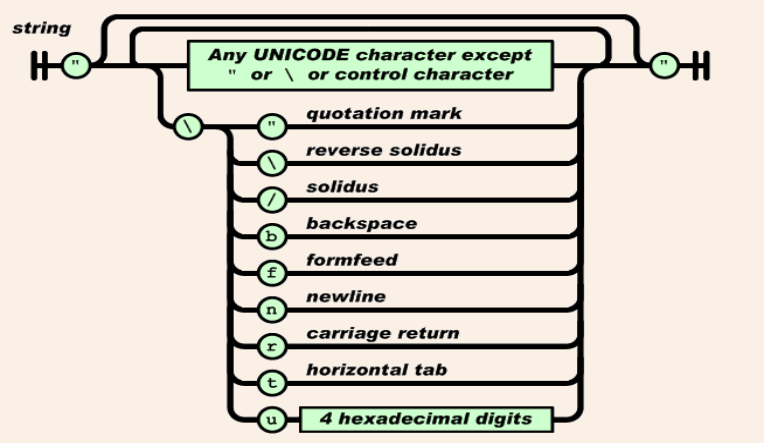
Gambar 2. 15 Array JSON

Nilai (value) dapat berupa sebuah string dalam tanda kutip ganda, atau angka, atau true atau false atau null, atau sebuah objek atau sebuah larik. Struktur- struktur tersebut dapat disusun bertingkat.



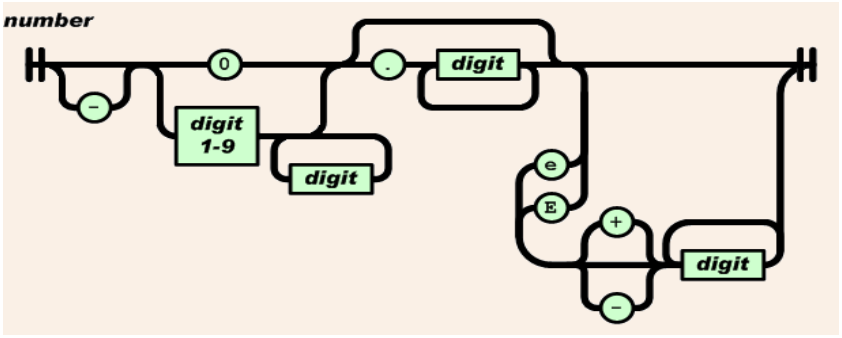
Gambar 2. 16 Value JSON

String adalah kumpulan dari nol atau lebih karakter Unicode, yang dibungkus dengan tanda kutip ganda. Di dalam string dapat digunakan backslash escapes "\" untuk membentuk karakter khusus. Sebuah karakter mewakili karakter tunggal pada string. String sangat mirip dengan string C atau Java.



Gambar 2. 17 String JSON

Angka adalah sangat mirip dengan angka di C atau Java, kecuali format oktal dan heksadesimal tidak digunakan.



Gambar 2. 18 Number JSON

Spasi kosong (whitespace) dapat disisipkan di antara pasangan tanda-tanda tersebut, kecuali beberapa detil encoding yang secara lengkap dipaparkan oleh bahasa pemprograman yang bersangkutan.

2.2.16 Java



Gambar 2. 19 Logo Java

Java adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam. Bahasa ini awalnya dibuat oleh James Gosling saat masih bergabung di Sun Microsystems saat ini merupakan bagian dari Oracle dan dirilis tahun 1995. Bahasa ini banyak mengadopsi sintaksis yang terdapat pada C dan C++ namun dengan sintaksis model objek yang lebih sederhana serta dukungan rutin-rutin aras bawah yang minimal. Aplikasi-aplikasi berbasis java umumnya dikompilasi ke dalam p-code (*bytecode*) dan dapat dijalankan pada berbagai Mesin Virtual Java (JVM). Java merupakan bahasa pemrograman yang bersifat umum/non-spesifik (*general purpose*), dan secara khusus didisain untuk

memanfaatkan dependensi implementasi seminimal mungkin. Karena fungsionalitasnya yang memungkinkan aplikasi java mampu berjalan di beberapa platform sistem operasi yang berbeda, java dikenal pula dengan slogannya, "Tulis sekali, jalankan di mana pun". Java merupakan bahasa pemrograman yang paling populer digunakan, dan secara luas dimanfaatkan dalam membangun berbagai jenis perangkat lunak aplikasi ataupun aplikasi berbasis *web*. Java digunakan untuk membuat aplikasi pelaporan pemadam kebakaran yang sedang dibangun[9].

2.2.17 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 web site 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[3].

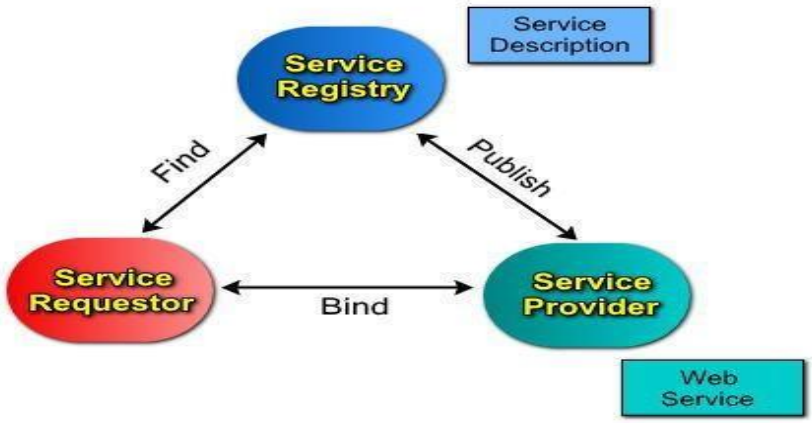
Web service bertujuan untuk meningkatkan kolaborasi antar pemrogram dan perusahaan, yang memungkinkan sebuah fungsi di dalam Web Service dapat dipinjam oleh aplikasi lain tanpa perlu mengetahui detil pemrograman yang terdapat di dalamnya.

Beberapa alasan mengapa digunakannya web service adalah sebagai berikut:

1. Web service dapat digunakan untuk mentransformasikan satu atau beberapa bisnis logic atau class dan objek yang terpisah dalam satu ruang lingkup yang menjadi satu, sehingga tingkat keamanan dapat ditangani dengan baik.
2. Web service memiliki kemudahan dalam proses deployment-nya, karena tidak memerlukan registrasi khusus ke dalam suatu sistem operasi. Web service cukup di-upload ke web server dan siap diakses oleh pihak-pihak yang telah diberikan otorisasi.

3. Web service berjalan di port 80 yang merupakan protokol standar HTTP, dengan demikian web service tidak memerlukan konfigurasi khusus di sisi firewall.

2.2.17.1 Arsitektur Web service



Gambar 2. 20 Arsitektur *Web Service*

Pada gambar diatas, ada tiga komponen yang membuat *web service* berjalan. Ketiga komponen itu adalah:

1. ***Service provide**r*, merupakan pemilik *Web service* yang berfungsi menyediakan kumpulan operasi dari *Web service*.
2. ***Service requesto**r*, merupakan aplikasi yang bertindak sebagai klien dari *Web service* yang mencari dan memulai interaksi terhadap layanan yang disediakan.
3. ***Service registry***, merupakan tempat dimana *Service provider* mempublikasikan layanannya. Pada arsitektur *Web service*, Service registry bersifat optional. Teknologi *web service* memungkinkan kita dapat menghubungkan berbagai jenis *software* yang memiliki *platform* dan sistem operasi yang berbeda.

2.2.17.2 Jenis-jenis *Web Service*

Jenis *web service* dapat dibagi menjadi dua, yaitu REST dan SOAP.

1. *Representational State Transfer* (REST)

REST adalah salah satu jenis *web service* yang menerapkan konsep perpindahan antar *state*. *State* disini dapat digambarkan seperti jika *browser* meminta suatu halaman *web*, maka *server* akan mengirimkan *state* halaman *web* yang sekarang ke *browser*. Bernavigasi melalui *link-link* yang disediakan sama halnya dengan mengganti *state* dari halaman *web*. Begitu pula REST bekerja, dengan bernavigasi melalui *link-link* HTTP untuk melakukan aktivitas tertentu, seakan-akan terjadi perpindahan *state* satu sama lain. Perintah HTTP yang bisa digunakan adalah fungsi *GET, POST, PUT* atau *DELETE*. Balasan yang dikirimkan adalah dalam bentuk XML sederhana tanpa ada protokol pemaketan data, sehingga informasi yang diterima lebih mudah dibaca dan diparsing disisi *Client* [18].

Dalam pengaplikasiannya, REST lebih banyak digunakan untuk *web service* yang berorientasi pada *resource*. Maksud orientasi pada *resource* adalah orientasi yang menyediakan *resource-resource* sebagai layanannya dan bukan kumpulan-kumpulan dari aktivitas yang mengolah *resource* itu. Alasan mengapa REST tidak digunakan dalam skripsi ini karena orientasi pada *resource*nya itu, sedangkan aplikasi lowongan pekerjaan membutuhkan pemanggilan metode yang bisa dikerjakan terhadap kumpulan *resource* data lowongan. Beberapa contoh *web service* yang menggunakan REST adalah: Flickr API (*Application Program Interface*), YouTube API, Amazon API.

2. *Simple Object Access Protokol* (SOAP)

SOAP merupakan protokol standar yang ringan dan ditujukan untuk pertukaran informasi dalam desentralisasi sistem. Protokol ini melakukan pemanggilan metode dalam bahasa XML (*Extensible Markup Language*). Namun, selain itu juga kita dapat menggunakan format pertukaran data lainnya yaitu JSON (*JavaScript Object Notation*). Dengan demikian, SOAP adalah suatu mekanisme yang sederhana untuk melakukan pertukaran struktur dan tipe informasi dalam lingkungan yang tersebar dan terdistribusi menggunakan XML/JSON. Dokumen SOAP yang digunakan

untuk melakukan *request* disebut SOAP *request* sedangkan dokumen SOAP yang diperoleh dari web service disebut dengan SOAP *response* [20].

2.2.18 PHP

Menurut Anhar, ST (2010), “PHP singkatan dari Personal Hypertext Preprocessor yaitu bahasa pemrograman web server-side yang bersifat open source. PHP merupakan script yang terintegrasi dengan HTML dan berada pada server. PHP adalah script yang digunakan untuk membuat halaman website yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh client. Mekanisme ini menyebabkan informasi yang diterima client selalu yang terbaru. Semua script PHP dieksekusi pada server dimana script tersebut dijalankan.

PHP pertama kali di kembangkan pada tahun 1995 oleh Rasmus Lerdorf, namun sekarang di ambil oleh oleh The PHP Group. Pada awalnya PHP adalah singkatan dari Personal Home Page, namun dalam perkembangannya, di ubah menjadi PHP: Hypertext Preprocessor, sebuah kepanjangan rekursif[11].

PHP berjalan pada sisi server sehingga PHP disebut juga sebagai bahasa ServerSideScripting. Artinya bahwa dalam setiap/untuk menjalankan PHP, wajib adanya web server.PHP ini bersifat opensource sehingga dapat dipakai secara cuma-cuma dan mampu lintas platform, yaitu dapat berjalan pada sistem operasi Windows maupun Linux.PHP juga dibangun sebagai modul pada web apache dan sebagai binary yang dapat berjalan sebagai CGI. Pada pembangunan aplikasi pelaporan benacana kebakaran Kota Bandung khususnya halaman backend menggunakan script PHP untuk membuat halaman pelaporan. Berikut contoh skrip dasar PHP bisa dilihat pada gambar 2.21:

```
<html>
<head>
<title>Contoh Kode PHP </title>
</head>
<body>
<?php
echo ("Ini adalah Kode PHP yang diletakan di antara kode HTML");
?>
</body>
</html>
```

Gambar 2. 21 Contoh Skrip PHP

Menurut Vikram Vaswani (2004), PHP memiliki beberapa keunggulan, antara lain[22] :

1. Kinerja

   Script yang ditulis dalam PHP mengeksekusi lebih cepat dibandingkan yang ditulis dalam bahasa script lain.

2. Portabilitas

   PHP tersedia untuk UNIX, Microsoft Windows, Mac OS, dan OS/2.PHP merupakan program portable antar platform. Kemampuan untuk melakukan cross-platform merupakan salah satu keunggulan bagi lingkungan perusahaan yang multiplatform.

3. Kemudahan dalam penggunaan

   PHP adalah bahasa pemograman yang sangat canggih dan dilengkapi dengan 5000 fungsi. Hal ini merupakan salah satu alasan PHP disukai sebagai alat prototyping untuk aplikasi berbasis web.

4. Open Source

   PHP merupakan bahasa opensource dan tersedia secara bebas di web serta dapat digunakan tanpa biaya lisensi.

2.2.19 MySQL

MySQL merupakan *software database* yang paling populer di kalangan para pengembang aplikasi *database*. MySQL juga adalah suatu perangkat lunak yang menganut atau mengimplementasikan model basis data relasional maka MySQL disebut sebagai *Relational Database Management System* (RDBMS). MySQL merupakan *software* RDBMS (atau *server database*) yang dapat mengelola *database* dengan cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak *user* (*multi-user*), dan dapat melakukan suatu proses secara sinkron atau berbarengan (*multi-threaded*) [11].

Saat ini, MySQL banyak digunakan di berbagai kalangan untuk melakukan penyimpanan dan pengolahan data, mulai dari kalangan akademis sampai ke industri, baik industri kecil, menengah, maupun besar. Lisensi MySQL terbagi menjadi dua. Anda dapat menggunakan MySQL sebagai produk *open source* dibawah GNU *General Public License* (gratis) atau dapat membeli lisensi dari versi komersialnya. MySQL versi komersial tentu memiliki nilai lebih atau kemampuan-kemampuan yang tidak disertakan pada versi gratis. Pada kenyataannya, untuk keperluan industri menengah kebawah, versi gratis masih dapat digunakan dengan baik. Beberapa contoh aplikasi yang menggunakan MySQL adalah Joomla, Wordpress, MyBB, phpBB, dan masih banyak lagi.

2.2.20 Metode Pengujian Sistem

Metode pengujian sistem terdiri dari Pengujian *black box* dan *beta* yang dilakukan untuk mengetahui efektifitas dari perangkat lunak (*software*) yang digunakan.

2.2.20.1 Pengujian Black box

Pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Pengujian ini dianalogikan seperti melihat suatu kotak hitam yang hanya bisa dilihat penampilan luarnya saja, tanpa tahu ada apa dibalik bungkus hitamnya. *Black box* testing melakukan evaluasi untuk menemukan kesalahan dalam kategori berikut:

a. Fungsi tidak benar atau hilang.
b. Kesalahan *interface* atau antarmuka.
c. Kesalahan dalam struktur data atau akses *database eksternal*.
d. Kesalahan kinerja atau perilaku dan kesalahan inisialisasi dan terminasi.

2.2.20.2 Skala Likert

Skala likert adalah suatu skala psikometrik yang digunakan dalam kuesioner dan merupakan salah satu teknik yang dapat digunakan dalam evaluasi suatu program atau kebijakan perencanaan. Rensis Likert telah mengembangkan sebuah skala untuk mengukur sikap masyarakat di tahun 1932 yang sekarang terkenal dengan nama skala Likert. Skala Likert ini merupakan skala yang dapat dipergunakan untuk mengukur sikap, pendapat, dan persepsi seseorang atau sekelompok orang mengenai suatu gejala atau fenomena [8].

Dalam skala Likert terdapat dua bentuk pernyataan yaitu pernyataan positif yang berfungsi untuk mengukur sikap positif, dan pernyataan negative yang berfungsi untuk mengukur sikap negative objek. Skor pernyataan positif dimulai dari 1 untuk sangat tidak setuju (TS), 2 untuk tidak setuju (KS), 3 untuk raguragu (CS), 4 untuk setuju (S), dan 5 untuk sangat setuju (SS). Skor pernyataan negative dimulai dari 1 untuk sangat setuju (SS), 2 untuk setuju (S), 3 untuk raguragu (CS), 4 untuk tidak setuju (KS), dan 5 untuk sangat tidak setuju (TS). Beberapa menghilangkan option “Ragu-ragu” dalam instrument untuk memudahkan dalam melihat angket yang responden isikan. Skala Likert digunakan untuk mengukur kesetujuan dan ketidaksetujuan seseorang terhadap sesuatu rencana program, pelaksanaan program ataupun tingkat keberhasilan suatu program.

Kekurangan Skala Likert :

1. Karena ukuran yang digunakan adalah ukuran ordinal, skala Likert hanya dapat mengurutkan individu dalam skala, tetapi tidak dapat membandingkan berapa kali satu individu lebih baik dari individu yang lain.
2. Kadangkala total skor dari individu tidak memberikan arti yang jelas, karena banyak pola respons terhadap beberapa item akan memberikan skor yang sama.

Adanya kelemahan di atas sebenarnya dapat dipikirkan sebagain error dari respons yang terjadi.

Kelebihan Skala Likert :

1. Mudah dibuat dan di terapkan.
2. Skala Likert lebih mudah membuatnya dibanding lain sepertti skala Thurstone.
3. Terdapat kebebasan dalam memasukan pertanyaan- pertanyaan, asalkan sesuai dengan konteks permasalahan yang diteliti.
4. Jawaban suatu item dapat berupa alternative, sehingga informasi mengenai item tersebut diperjelas.
5. Reliabilitas pengukuran bisa diperoleh dengan jumlah item tersebut diperjelas.