

## **BAB 2**

### **TINJAUAN PUSTAKA**

#### **2.1 PMI**

PMI merupakan sebuah organisasi perhimpunan nasional di Indonesia yang bergerak dibidang kemanusiaan dengan slogan setetes darah anda, nyawa bagi sesama. PMI selalu mempunyai tujuh prinsip dasar Gerakan Internasional Palang Merah dan Bulan Sabit Merah yaitu kemanusiaan, kesamaan, kesukarelaan, kemandirian, kesatuan, kenetralan, dan kesemestaan. Palang merah indonesia tidak memihak golongan politik, ras, suku ataupun agama tertentu. Palang Merah Indonesia dalam pelaksanaanya juga tidak melakukan pembedaan tetapi mengutamakan korban yang paling membutuhkan pertolongan segera untuk keselamatan jiwanya.

Sejarah berdirinya Palang Merah Indonesia sebetulnya sudah dimulai sebelum Perang Dunia II, tepatnya 12 Oktober 1873. Pemerintahan kolonial Belanda mendirikan Palang Merah Indoensia dengan nama *Nederlandsche Roode Kruis Afdeeling Indie* (NERKAI) yang kemudian dibubarkan pada saat pendudukan jepang. Palang Merah Indonesia sendiri diawali pada tahun 1932. Kegiatan tersebut dipelopori Dr R C L Senduk dan Dr Bahder Djohan dengan membuat rancangan pembentukan PMI. Rancangan tersebut mendapat dukungan luas tertuma dari kalangan terpelajar Indonesia, dan diajukan ke dalam sidang konferensi Narkai pada tahun 1940, akan tetapi ditolak. Proses pembentukan PMI dimulai 3 September 1945 saat itu Presiden Soekarno memerintahkan Dr Boentaran (Menkes RI Kabinet I) agar membantu suatu badan Palang Merah Nasional [6].

Dalam PMI sendiri yang bergerak dalam hal kemanusiaan, salah satunya seperti donor darah. Donor darah itu sendiri dilakukan oleh PMI untuk membantu orang-orang yang sedang membutuhkan darah, bagi PMI orang-orang yang sudah mendonorkan darahnya secara sukarela, diharapkan darah yang sudah didonorkan dapat bermanfaat bagi orang-orang yang membutuhkan.

#### **2.2 Donor Darah**

Donor darah berarti melakukan kegiatan kemanusiaan dalam hal menyumbangkan darah yang kita miliki terhadap orang lain melalui tindakan

penyadapan darah. Definisi donor darah secara umum ialah kegiatan menyumbangkan darah secara sukarela dilakukan untuk banyak orang yang membutuhkan transfusi darah. Dalam dunia medis, donor darah berarti menyumbangkan darah kepada orang lain dengan tujuan untuk menyelamatkan jiwa orang yang membutuhkan [6].

Dalam Peraturan Pemerintah Republik Indonesia Nomor 18 Tahun 1980 yang berisi tentang Transfusi Darah. Bahwa usaha transfusi darah merupakan bagian dari tugas pemerintah di bidang pelayanan kesehatan rakyat dan merupakan suatu bentuk pertolongan yang sangat berharga kepada umat manusia, berdasarkan ilmu pengetahuan kedokteran, satu-satunya sumber darah yang aman untuk keperluan transfusi darah adalah darah manusia. Transfusi darah merupakan suatu tindakan memasukkan darah atau komponennya ke dalam sistim pembuluh darah seseorang. Komponen darah yang biasa ditransfusikan kedalam tubuh seseorang adalah sel darah merah, trombosit, plasma.

Alasan transfusi darah dan penggunaannya adalah:

1. Untuk memperbaiki anemia (kadar hemoglobin yang rendah).
2. Untuk mengganti kehilangan darah karena terjadi pendarahan pada operasi atau kecelakaan.
3. Untuk mengganti kandungan tertentu dari darah, misalnya dalam faktor-faktor pembekuan.
4. Meningkatkan kemampuan darah dalam mengangkut oksigen.
5. Memperbaiki volume darah tubuh.
6. Memperbaiki kesehatan.

### **2.2.1 Golongan Darah**

Golongan darah adalah ilmu pengklasifikasian darah dari suatu kelompok berdasarkan ada atau tidaknya zat antigen warisan pada permukaan membran sel darah merah. Hal ini disebabkan karena adanya perbedaan jenis karbohidrat dan protein pada permukaan membran sel darah merah tersebut [7]. Dengan sepengetahuan banyak orang golongan darah yang paling penting adalah Rhesus.

Rhesus ialah jenis penggolongan golongan darah pada setiap manusia atau yang lebih dikenal dengan sistem A, B, AB, dan O. Rhesus sendiri memiliki dua

jenis golongan yaitu Rhesus positif (Rh+) dan Rhesus negatif (Rh-) berdasarkan ada tidaknya antigen rhesus pada dinding sel darah merah seseorang. Rh+ dalam darahnya memiliki antigen rhesus yang ditunjukkan dengan reaksi positif atau dijumpai adanya gumpalan sel darah merah pada waktu dilakukan tes dengan antibodi Rh. Sedangkan Rh- dalam darahnya tidak memiliki antigen Rhesus yang menunjukkan reaksi negatif atau tidak dijumpai penggumpalan saat dilakukan tes dengan antobodi Rh. Dalam penulissannya, jenis penggolongan rhesus ini digabungkan dengan penggolongan ABO yaitu berupa A+ dan A-, B+ dan B-, O+ dan O-, serta AB+ dan AB- [8].

Dalam dunia kedokteran golongan Rh- merupakan golongan darah yang termasuk langka. Langkanya golongan darah ini disebabkan karena sifar alelnya yang resesif. Sehingga Rh- baru akan muncul apabila alel resesif bertemu dengan alel resesif. Sebanyak 85% penduduk dunia memiliki Rh+, dan hanya 15% yang memiliki Rh-.

Golongan Darah	Komponen													
	Rhesus +							Rhesus -						
	WB	TC	PRC	FFP	AHF	BC	LP	WB	TC	PRC	FFP	AHF	BC	LP
A	0	4	305	210	35	0	0	0	0	0	0	0	0	0
B	0	3	1249	147	34	0	0	0	0	0	0	0	0	0
AB	0	1	279	197	42	0	0	0	0	0	0	0	0	0
O	0	0	1620	222	62	0	0	0	0	0	0	0	0	0

**Gambar 2.1 Stok Darah di PMI Kota Bandung Berdasarkan Golongan Darah dan Rhesus [9]**

Dalam gambar diatas ialah contoh dari golongan darah dan rhesus yang berada di PMI Kota Bandung

### 2.2.2 Prosedur Donor Darah

Didalam kegiatan donor darah terdapat prosedur meliputi selesi donor, pengambilan darah, pemeriksaan serologi, pengolahan komponen darah, penyimpanan darah dan pengiriman darah [10]

#### 1) Seleksi darah

pada seleksi donor darah bertujuan untuk menjamin kesehatan dan keselamatan pendonor agar tidak ada kesalahan yang berakibat fatal dalam proses donor darah.

a. Petugas

Petugas yang menyeleksi awal para pendonor adalah yang mempunyai kompetensi dalam hal seleksi donor. Untuk pemeriksaan kesehatan pendonor harus dilakukan oleh seorang dokter.

b. Metoda

Pada setiap donor darah harus terlebih dahulu mendapatkan informasi-informasi sebagai berikut:

- a) Penyumbang darah atau disebut sebagai pendonor (donor) diseleksi kesehatannya dengan dilakukan pemeriksaan denyut nadi, tekanan darah, suhu tubuh dan contoh darah untuk mengetahui adanya penyakit *anemia*.
- b) Menggali informasi tentang keadaan donor dengan menanyakan apakah pernah atau sedang menderita keadaan tertentu yang menyebabkan darah mereka tidak memenuhi syarat untuk donor. Keadaan tersebut adalah hepatitis, penyakit jantung, kanker, asma yang berat, malaria, kelainan pendarahan, HIV, dan kemungkinan terjangkit oleh virus HIV. tekanan darah tinggi, tekanan darah rendah, anemia atau pemakaian obat tertentu, yang bisa menyebabkan tidak terpenuhinya syarat untuk menyumbangkan darah.
- c) Pengisian daftar isian donor.
- d) Penandatanganan persetujuan tindakan medis
- e) Pemeriksaan pendahuluan terdiri dari penimbangan berat badan, HB, golongan darah dan pemeriksaan fisik oleh dokter.

Persyaratan donor darah:

1) Keadaan Umum

Calon donor tidak sedang sakit, tidak dalam pengaruh obat-obatan (narkotika) dan alkohol serta tidak menderita penyakit kronis yang menular.

2) Umur Donor

Pendonor diharuskan berumur antara 17 hingga 60 tahun, kecuali atas pertimbangan dari dokter. Donor yang berumur 60 tahun dapat menyumbangkan darahnya sampai dengan umur 65 tahun. Donor yang batu pertama kali tidak diperbolehkan pada umur 60 tahun.

### 3) Berat Badan (BB)

Donor dengan BB minimal 45 kg dapat menyumbangkan darahnya sebanyak 350 ml, ditambah sejumlah darah untuk pemeriksaan yang jumlahnya tidak lebih dari 30 ml. Donor dengan berat badan 50 kg atau lebih dapat menyumbangkan darahnya maksimal sebanyak 450ml tetapi tidak melebihi 15% dari perkiraan volume darah calon donor ditambah sejumlah darah untuk pemeriksaan yang jumlahnya tidak lebih dari 30 ml.

4) Suhu tubuh calon donor tidak lebih dari 37°C.

5) Denyut nadi teratur berkisar antara 60 hingga 100 x/menit.

6) Tekanan darah sistolik antara 100 hingga 160 mmHg dan diastolik antara 60 hingga 100 mmHg.

7) Kadar hemoglobin calon donor  $\geq 12,5$  g/dl. Penetapan kadar tersebut dilakukan minimal dengan metode CuSO<sub>4</sub>(BJ 1.053).

8) Bagi wanita yang selesai haid, 6 bulan setelah melahirkan dan 3 bulan setelah berhenti menyusui dapat menyumbangkan darahnya.

9) Jarak penyumbangan darah lengkap tidak kurang dari 8 minggu. Maksimal 5 kali setahun. Penyumbangan darah lengkap dapat dilakukan minimal 48 jam setelah menjalani plasma tromboferesis. Jarak penyumbang komponen darah trombosit minimal 1 bulan dengan jumlah trombosit  $>150.000/\mu\text{l}$ , maksimal 6 kali setahun untuk laki-laki dan 4 kali untuk perempuan.

## 2.3 Android

Android adalah sebuah sistem operasi telepon seluler dan komputer tablet berbasis linux. Android merupakan sistem operasi yang digunakan di smartphone yang fungsinya sama seperti sistem operasi Symbian, iOS dan Blackberry OS [11]. Android tidak terikat terhadap satu merek smartphone saja. Beberapa vendor terkenal sudah memakai sistem operasi android untuk kebutuhan produknya. Android dibuat dengan basis kernel Linux yang telah dimodifikasi, dan untuk setiap versinya diberi kode nama seperti:

1) Éclair (2.0, 2.1)

- 2) Frozen Yogurt (2.2)
- 3) Gingerbread (2.3)
- 4) Honeycomb (3.0, 3.1, 3.2)
- 5) Ice Cream Sandwich (4.0)
- 6) Jelly Bean (4.1, 4.2 dan 4.3)
- 7) Kitkat (4.4 hingga 4.4.4)
- 8) Lollipop (5.0 hingga 5.1.1)
- 9) Marshmallow (6.0)
- 10) Nougat (7.0), Oreo (8.0)

Pemasaran android pada tahun 2012 sekitar 630 juta smartphone akan terjual diseluruh dunia, dimana pada tahun itu diperkirakan sebanyak 49,2% diantaranya akan menggunakan OS Android pada smartphone.

Awalnya Perusahaan Google Inc, membeli Android Inc yang merupakan pendatang baru yang membuat piranti lunak untuk smartphone. Para pendiri Android Inc, bekerja pada Google untuk memulai membangun platform Android secara lebih intensif, diantaranya Andy Rubin, Rich Miner, Nick Sears, dan Chris White. Pada tanggal 12 November 2007 Google bersama *Open Handset Alliance* (OHA) merilis Google *Android Software Development Kit* (SDK). Paket SDK yang dirilis untuk mengembangkan aplikasi yaitu sistem operasi, middleware, dan aplikasi utama untuk perangkat mobile utama perangkat mobile. Dengan dirilisnya SDK tersebut membuka kesempatan kepada pengembang untuk mengembangkan sebuah aplikasi berbasis android [12].

Android merupakan perangkat bergerak pada sistem operasi untuk telepon seluler yang berbasis linux. Android mempunyai API yang disediakan menawarkan akses ke hardware, maupun data-data ponsel sekalipun, atau data sistem sendiri untuk memudahkan pihak developer mengembangkan aplikasi untuk android. Bahkan developer ataupun pengguna dapat menghapus aplikasi inti dari mengantikannya dengan aplikasi pihak ketiga.

Menurut Emarketer yang dilansir oleh Tech in Asia pengguna smartphone android di Indonesia berjumlah 103 juta pengguna, dengan banyaknya pengguna tersebut dan pada hasil kuisisioner dengan 113 koresponden, 97,3% pengguna

android. Maka dari itu penelitian ini ditujukan dengan penggunaan sistem operasi yang berbasis android [13]

### 2.3.1 Software Development Kit (SDK)

Android-SDK merupakan *tools* bagi para programmer yang ingin mengembangkan aplikasi berbasis google android. Android SDK mencakup seperangkat alat pengembangan yang komprehensif. Android SDK terdiri dari *debugger*, *libraries*, *handset emulator*, dokumentasi, contoh kode, dan *tutorial*. Saat ini Android sudah mendukung arsitektur x86 pada Linux (distribusi Linux apapun untuk *desktop* modern), Mac OS X 10.4.8 atau lebih, Windows XP atau Vista. Persyaratan mencakup JDK, Apache Ant dan Python 2.2 atau yang lebih baru. IDE yang didukung secara resmi adalah Eclipse 3.2 atau lebih dengan menggunakan *plugin Android Development Tools (ADT)*, dengan ini pengembang dapat menggunakan teks editor untuk mengedit file Java dan XML serta menggunakan peralatan *command line* untuk menciptakan, membangun, melakukan *debug* aplikasi Android dan pengendalian perangkat Android misalkan *reboot*, menginstal paket perangkat lunak dengan jarak jauh [14] .

Pada tanggal 18 Agustus 2008, Android SDK 0.9 *beta* dirilis. Rilis ini menyediakan API yang diperbarui dan diperluas, perbaikan pada alat-alat pengembangan dan desain terbaru untuk layar awal. Petunjuk untuk *upgrade* SDK sudah tersedia pada rilis sebelumnya. Pada tanggal 23 September 2008, Android 1.0 SDK telah dirilis. Pada tanggal 9 Maret 2009, Google merilis versi 1.1 untuk telepon seluler Android. Rilis terbaru tersebut termasuk dukungan untuk pencarian dengan suara, harga aplikasi, perbaikan jam *alarm*, perbaikan pengiriman gmail, perbaikan surat pemberitahuan dan peta.

Pada pertengahan Mei 2009, Google merilis versi 1.5 (Cupcake) pada sistem operasi Android dan SDK. Pembaruan ini termasuk banyak fitur baru seperti perekaman video, dukungan untuk *bluetooth*, sistem *keyboard* pada layar dan pengenalan suara. Rilis ini juga membuka *AppWidget framework* kepada para pengembang yang memungkinkan orang untuk membuat *widget* sendiri pada halaman *home*. Pada September 2009 versi 1.6 (Donut) dirilis yang menampilkan hasil pencarian yang lebih baik dan penggunaan indikator baterai [14].

Aplikasi Android dipaketkan ke dalam format .apk dan disimpan pada *folder* /data/app. Pengguna dapat menjalankan perintah adb root untuk mengakses *folder* tersebut karena *root* memiliki izin untuk mengakses folder tersebut.

### 2.3.2 Java

Java adalah bahasa pemrograman yang multi platform dan multi device. Sekali menuliskan sebuah program dengan menggunakan Java, dapat menjalankannya hampir di semua komputer dan perangkat lain yang support Java, dengan sedikit perubahan atau tanpa perubahan sama sekali dalam kodenya. Aplikasi dengan berbasis Java ini dikompulasikan ke dalam p-code dan bisa dijalankan dengan Java Virtual Machine. Fungsionalitas dari Java ini dapat berjalan dengan platform sistem operasi yang berbeda karena sifatnya yang umum dan non-spesifik.

Sejarah dari perkembangan bahasa pemrograman Java terlahir dari The Green Project, yang berjalan selama 18 bulan, dari awal tahun 1991 hingga musim panas 1992. Proyek tersebut belum menggunakan versi yang dinamakan Oak. Proyek ini dimotori oleh Patrick Naughton, Mike Sheridan, dan James Gosling, beserta sembilan pemrogram lainnya dari Sun Microsystems. Salah satu hasil proyek ini adalah maskot Duke yang dibuat oleh Joe Palrang.

Nama Oak, diambil dari pohon oak yang tumbuh di depan jendela ruangan kerja "Bapak Java", James Gosling. Nama Oak ini tidak dipakai untuk versi release Java karena sebuah perangkat lunak lain sudah terdaftar dengan merek dagang tersebut, sehingga diambil nama penggantinya menjadi "Java". Nama ini diambil dari kopi murni yang digiling langsung dari biji (kopi tubruk) kesukaan Gosling. Konon kopi ini berasal dari Pulau Jawa. Jadi nama bahasa pemrograman Java tidak lain berasal dari kata Jawa (bahasa Inggris untuk Jawa adalah Java). [15]

### 2.3.4 Flutter

Flutter adalah kerangka kerja seluler UI Google untuk membuat antarmuka asli berkualitas tinggi di iOS dan Android dalam waktu singkat. Flutter berfungsi dengan kode yang ada, digunakan oleh pengembang dan organisasi di seluruh



dunia, dan sumber terbuka [16]. Framework Flutter tersebut mempunyai beberapa keunggulan seperti:

- a) Dioptimalkan, mesin rendering 2D mobile-first dengan dukungan yang sangat baik untuk teks.
- b) Kerangka gaya reaksi modern.
- c) Kumpulan widget yang kaya untuk Android dan iOS.
- d) API untuk tes unit dan integrasi.
- e) Interop dan plugin API untuk terhubung ke sistem dan SDK pihak ketiga.
- f) Headless test runner untuk menjalankan tes pada Windows, Linux, dan Mac.
- g) Alat baris perintah untuk membuat, membuat, menguji, dan menyusun aplikasi.

#### **2.4 GPS (*Global Positioning System*)**

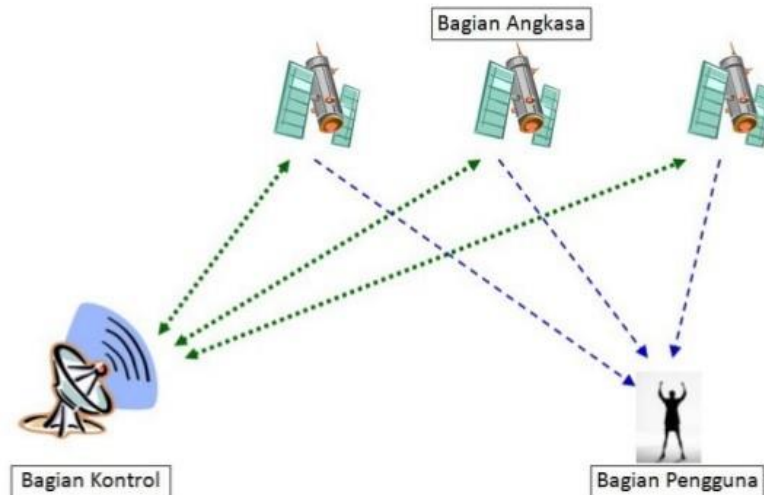
*Global Positioning System* atau yang lebih dikenal dengan GPS, hampir setiap pengguna smartphone sering memakai teknologi ini. GPS itu sendiri merupakan navigasi sistem berbasis satelit yang dikembangkan oleh Departemen Pertahanan Amerika yang didukung oleh 27 jaringan satelit. GPS terdiri dari 3 segmen yaitu Segmen angkasa, Segmen kontrol atau pengendali, dan Segmen pengguna.

Segmen angkasa terdiri dari 24 satelit yang beroperasi dalam 6 orbit pada ketinggian 20.200 km dan inklinasi 55 derajat dengan periode 12 jam. Satelit tersebut memutar orbitnya sehingga minimal ada 6 satelit yang dapat dipantau pada titik manapun di bumi ini. Satelit tersebut mengirimkan posisi dan waktu kepada pengguna seluruh dunia.

Dalam segmen kontrol atau pengendali bagian ini untuk mengontrol setiap satelit yang ada diluar angkasa. Setiap satelit dapat berada sedikit diluar orbit, sehingga bagian ini melacak orbit satelit, lokasi, ketinggian, dan kecepatan. Sinyal-sinyal dari satelit diterima oleh bagian kontrol, dikoreksi dan dikirimkan kembali ke satelit. Koreksi data lokasi yang tepat dari satelit ini disebut dengan data ephemeris yang nantinya akan dikirimkan kepada alat navigasi kita.

Bagian pengguna bagian ini terdiri atas navigasi yang digunakan. Satelit akan memancarkan data almanak dan ephemeris yang akan diterima oleh alat navigasi secara teratur. Data almanak berisikan perkiraan lokasi satelit yang

dipancarkan terus menerus oleh satelit. Data ephemeris dipancarkan oleh satelit dan valid untuk sekitar 4-6 jam. Untuk menunjukkan koordinat sebuah titik, alat navigasi memerlukan paling sedikit sinyal dari 3 buah satelit. Untuk menunjukkan data ketinggian sebuah titik, diperlukan tambahan sinyal dari 1 buah satelit lagi [17].



**Gambar 2.2 Cara Kerja GPS**

Terdapat 2 metode dalam penggunaan teknologi GPS antara lain:

a. Metode Absolut

Metode ini juga dikenal dengan point positioning atau menentukan posisi berdasarkan satu pesawat penerima (receiver) saja. Ketelitian posisi dalam beberapa meter (tidak berketelitian tinggi) dan umumnya hanya diperuntukkan bagi keperluan navigasi. Karakteristik penentuan posisi dengan cara absolut ini adalah sebagai berikut:

1. Posisi ditentukan dalam sistem WGS 84 (terhadap pusat bumi).
2. Prinsip penentuan posisi adalah perpotongan ke belakang dengan jarak beberapa satelit sekaligus.
3. Hanya memerlukan satu receiver GPS.
4. Titik yang ditentukan posisinya bisa diam atau bergerak.
5. Ketelitaian posisi berkisar antara 5 sampai 10 meter.

Aplikasi utama untuk keperluan navigasi, metode posisi absolut ini umumnya menggunakan data pseudorange dan metode ini tidak dimaksudkan untuk aplikasi-aplikasi yang menuntut ketelitian posisi yang tinggi [18].

#### b. Metode Relatif

Metode ini sering disebut differential positioning yakni menentukan posisi suatu titik relatif terhadap titik lain yang telah diketahui koordinatnya, pengukuran dilakukan secara bersamaan pada dua titik dalam selang waktu tertentu. Selanjutnya dari data hasil pengamatan diproses/dihitung akan didapat perbedaan koordinat kartesian 3 dimensi atau disebut juga dengan baseline antar titik yang diukur. Metode ini menghasilkan posisi berketelitian tinggi (umumnya kurang dari 1 meter) dan diaplikasikan untuk keperluan survei geodetik atau pemetaan yang memerlukan ketelitian tinggi. Karakteristik umum dari metode penentuan posisi ini adalah sebagai berikut:

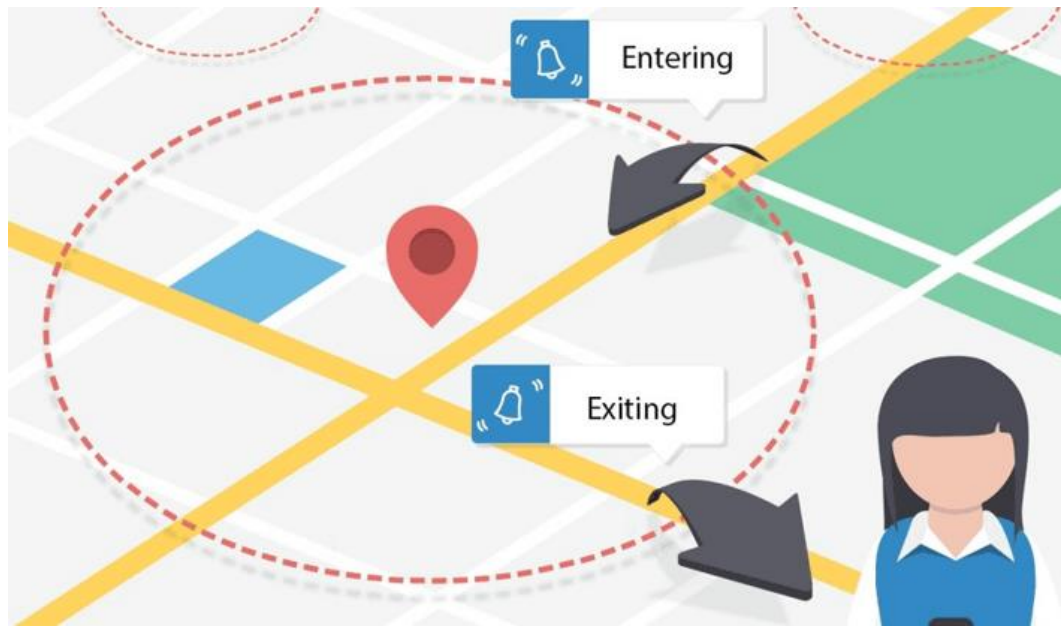
1. Memerlukan minimal 2 receiver, satu ditempatkan pada titik yang telah diketahui koordinatnya.
2. Posisi titik ditentukan relatif terhadap titik yang diketahui.
3. Konsep dasar adaah differencing process dapat mengeliminasi atau mereduksi pengaruh dari beberapa kesalahan dan bias.
4. Bisa menggunakan data pseudorange atau fase.
5. Ketelitian posisi yang diperoleh bervariasi dari tingkat milimeter hingga desimeter.
6. Aplikasi utama digunakan untuk survey geologi, pengukuran batas, geodesi dan navigasi berketelitian tinggi.

#### 2.4.1 Geofencing

Geofencing sebagai fitur yang ditawarkan dari sebuah software atau program yang memanfaatkan komponen GPS guna menentukan batasan geografi secara virtual. Program geofence memungkinkan administrator yang menggunakan fitur tersebut dapat menentukan titik-titik virtual yang menyampaikan sinyal

apabila perangkat yang dilengkapi GPS tersebut melewati batas pagar virtual. Baik masuk maupun keluar dari garis pagar yang telah ditentukan. Fitur aplikasi geofencing ini dapat melibatkan layanan lain seperti Google Earth, untuk menentukan garis pagar virtual, atau ditentukan berupa koordinat longitude dan latitude.

Penggunaannya dalam android saat ini sebagai aplikasi yang memungkinkan aplikasi berinteraksi dengan dunia fisik dan ideal untuk meningkatkan keterlibatan pengguna. Geofence itu sendiri didalam penggunaannya pada android menggabungkan posisi pengguna dengan parameter geofence untuk mengetahui lokasi pengguna berada di dalam atau diluar pagar virtual, bahkan jika pengguna keluar atau memasuki area tersebut. Geofence itu sendiri merupakan bagian dari *API Location Google*.



**Gambar 2.3 Geofencing [19]**

Untuk menggunakan teknologi geofencing ini, administrator atau pengembang harus terlebih dahulu menetapkan batas virtual di sekitar lokasi tertentu dalam perangkat lunak yang dilengkapi GPS atau RFID. Ini bisa sesederhana lingkaran yang digambar 100 kaki di sekitar lokasi di Google Maps, sebagaimana ditentukan menggunakan API saat mengembangkan aplikasi seluler. Geofencing virtual ini kemudian akan memicu respons ketika perangkat

resmi masuk atau keluar dari area tersebut, seperti yang ditentukan oleh administrator atau pengembang.

Geofence juga dapat diatur oleh pengguna akhir menggunakan kemampuan geofencing di aplikasi seluler mereka. Aplikasi ini, seperti Peningat iOS, memungkinkan Anda memilih alamat atau lokasi tempat Anda ingin memicu peringatan tertentu atau notifikasi push. Ini disebut perintah "*if this, then that*", di mana aplikasi diprogram untuk memicu aksi berdasarkan tindakan lain. [20]

#### 2.4.1.1 Metode Haversine

Dalam metode ini untuk perhitungan radius notifikasi yang nantinya akan digunakan sebagai jarak yang ditentukan untuk mendapatkan notifikasi dari user lain. Haversine Formula merupakan metode untuk mengetahui jarak antar dua titik dengan memperhitungkan bahwa bumi bukanlah sebuah bidang datar namun sebuah bidang yang memiliki derajat kelengkungan. Metode Haversine Formula menghitung jarak antara 2 titik dengan berdasarkan panjang garis lurus antara 2 titik pada garis bujur dan lintang [21].

Berikut bentuk Rumus Haversine Formula, dapat dilihat pada tabel 2.1:

**Tabel 2.1 Rumus Haversine Formula**

$$\Delta lat = lat2 - lat1$$

$$\Delta long = long2 - long1$$

$$a = \sin\left(\frac{\Delta lat}{2}\right)^2 + \cos(rad\ lat1) * \cos(rad\ lat2) * \sin\left(\frac{\Delta long}{2}\right)^2$$

$$c = 2\ asin(a)$$

$$d = R * c$$

Keterangan :

$R$  = jari-jari bumi sebesar 6371(km)

$\Delta lat$  = besaran perubahan latitude

$\Delta long$  = besaran perubahan longitude

$c$  = kalkulasi perpotongan sumbu

$d$  = jarak (km)

1 derajat = 0.0174532925 rad

### 2.4.2 Geolocation

*Geolocation* adalah fasilitas HTML 5 yang memungkinkan website mengetahui lokasi pengakses. Ini bisa diintegrasikan dengan Google API nantinya, seperti Google Maps, atau yang lainnya untuk membuat aplikasi berbasis lokasi. *Geolocation* mengacu pada identifikasi lokasi geografis dari pengguna atau perangkat melalui berbagai mekanisme collection data. *Geolocation* kebanyakan menggunakan alamat jaringan internal routing atau perangkat GPS untuk menentukan lokasi. Geolocation merupakan perangkat spesifik API.

Cara kerja dari *Geolocation* ketika smartphone menyalakan fitur GPS, komunikasi antar smartphone dan BTS (Base Transmission System) akan selalu berkomunikasi, dari komunikasi tersebut smartphone akan mengirimkan data-data, baik berupa aspasial maupun spasial, kemudian data tersebut yang diterima BTS akan dikirimkan ke server map, lalu kemudian data tersebut diolah dan ditampilkan dalam bentuk peta atau longitude dan latitude sesuai GPS yang ada pada perangkat. [22]

### 2.4.3 LBS (*Location Based Services*)

Penggunaan fitur LBS ini semakin meningkat. Salah satunya adalah aplikasi yang memanfaatkan fungsi dari lokasi dengan menggunakan GPS atau yang biasa disebut LBS. *Location Based Services* adalah layanan berbasis lokasi atau istilah umum yang sering digunakan untuk menggambarkan teknologi yang digunakan untuk menemukan lokasi perangkat yang pengguna gunakan. Layanan ini menggunakan teknologi GPS dan *Cell Based Location* dari vendor Google. Selain itu LBS tersebut terdiri dari beberapa komponen seperti *mobile devices*, *communication network*, *position component*, serta *service and content provider*. Karena LBS sangat bergantung pada lokasi pengguna ponsel, tujuan utama dari sistem penyedia layanan adalah menentukan di mana pengguna berada. Ada banyak teknik untuk mencapai ini.

*mobile devices* merupakan komponen yang sangat penting, piranti *mobile devices* tersebut diantaranya adalah smartphone, PDA, dan lainnya yang dapat berfungsi sebagai alat navigasi atau seperti halnya alat navigasi berbasis GPS.

Komponen *communication network* ini berupa jaringan telekomunikasi bergerak yang memindahkan data pengguna dari perangkat ke penyedia layanan. *Position component* yang dimaksud ialah posisi pengguna harus ditentukan, posisi ini didapatkan dengan jaringan telekomunikasi atau dengan GPS. Sedangkan *service and content provider* adalah penyedia layanan yang menyediakan layanan berbeda ke pengguna seperti pencarian rute, kalkulasi posisi, dan lainnya [23].

Untuk menentukan lokasi pengguna ponsel, satu metode melibatkan penggunaan jaringan telepon seluler. Misalnya, ID sel saat ini dapat digunakan untuk mengidentifikasi base transceiver station (BTS) yang digunakan ponsel untuk berkomunikasi. Setelah itu ditentukan, satu-satunya yang tersisa adalah untuk menentukan lokasi BTS. Sistem lain menggunakan satelit GPS, metode ini jauh lebih akurat daripada yang disebutkan sebelumnya dan sekarang menjadi lebih mudah oleh beberapa smartphone yang sudah memiliki penerima GPS *built-in*. Metode umum lainnya adalah penggunaan beacon pemandu jarak pendek. Perangkat semacam itu biasanya menggunakan teknologi WiFi atau Bluetooth dan ideal untuk aplikasi LBS dalam ruangan [23].

Layanan ini dapat diklasifikasikan menjadi dua jenis yaitu Push dan Pull. Dalam jenis layanan Push, pengguna menerima informasi dari penyedia layanan tanpa memintanya pada saat itu, meskipun pengguna mungkin awalnya berlangganan ke layanan pada waktu sebelumnya. Dalam jenis Pull, pengguna harus secara aktif meminta informasi.

Platform pengembangan perangkat lunak, terutama yang digunakan untuk membuat aplikasi seluler, seperti J2ME dan Android, memiliki API khusus yang mendukung LBS.

## 2.5 Firebase

Firebase pertama kali didirikan pada tahun 2011 oleh Andrew Lee dan James Tamplin. Produk yang pertama kali dikembangkan ialah Realtime Database, dimana developer dapat menyimpan dan melakukan sinkronisasi data ke banyak user. Kemudian berkembang menjadi layanan penyedia pengembangan aplikasi. Pada Oktober 2014, perusahaan tersebut diakuisisi oleh Google, berbagai fitur terus dikembangkan hingga diperkenalkan pada Mei 2016 di Google I/O. Firebase itu

sendiri adalah BaaS atau *Backend as a Service* yang saat ini dimiliki oleh Google. Firebase ini merupakan solusi yang ditawarkan oleh Google untuk mempermudah pekerjaan dalam membuat aplikasi berbasis mobile. Dengan adanya Firebase, membuat aplikasi bisa fokus mengembangkan aplikasi tanpa harus memberikan upaya yang besar untuk urusan backend [24].

Beberapa fitur yang dimiliki oleh Firebase adalah sebagai berikut :

- 1) Firebase Analytics.
- 2) Firebase Cloud Messaging dan Notifications.
- 3) Firebase Authentication.
- 4) Firebase Remote Config.
- 5) Firebase Real Time Database.
- 6) Firebase Crash Reporting.

### **2.5.1 Firebase Cloud Messaging dan Notifications**

Firebase Cloud Messaging (FCM) adalah pengiriman pesan lintas platform yang memungkinkan pengguna mengirimkan pesan dengan terpercaya tanpa biaya. Dengan FCM, pengguna dapat memberi tahu aplikasi klien bahwa email baru atau data lainnya tersedia untuk disinkronkan. Pengguna dapat mengirimkan pesan notifikasi untuk mendorong interaksi kembali dan retensi pengguna. Untuk kasus penggunaan seperti messagin instan, pesan dapat mentransfer payload hingga 4 KB ke aplikasi klien.

Implementasi FCM mencakup 2 komponen utama untuk mengirim dan menerima pesan yaitu

1. Lingkungan terpercata seperti Cloud Functions for Firebase atau server aplikasi yang akan digunakan untuk membuat, menargetkan dan mengirim pesan.
  2. Aplikasi klien iOS, Android, atau Web (Javascript) yang menerima pesan.
- Artsitektur dari penggunaan FCM seperti di Gambar 2.3 ini.





**Gambar 2.4 Firebase Cloud Messaging [25]**

Pesan notifikasi bisa dikirimkan ke pengguna melalui Notifications Console GUI dan HTTP request seperti pada gambar diatas.

### 2.5.2 Firebase Authentication

Firebase Authentication adalah sebuah service untuk memudahkan pembangunan sistem autentikasi yang aman, sekaligus meningkatkan pengalaman login dan pengalaman aktivasi bagi pengguna akhir. Fungsi ini menyediakan solusi identitas end-to-end, mendukung login akun email/sandi, autentikasi telepon, Google, Twitter, Facebook, Github dan yang lainnya [26]. Service ini bertujuan untuk memudahkan pengguna dalam melakukan registrasi dan login. Nantinya pengguna dapat menggunakan akun Google yang telah terdaftar, Twitter, Facebook, Github dan yang lainnya. Firebase ini dibuat oleh tim yang mengembangkan login dengan Google, pengelola sandi Chrome dan smart lock. Keamanan firebase menerapkan keahlian internal Google dalam mengelola salah satu database akun terbesar didunia.

Untuk membuat pengguna login ke aplikasi, dapatkan kredensial autentikasi dari pengguna terlebih dahulu. Kredensial ini dapat berupa alamat email dan sandi pengguna atau token OAuth dari penyedia identitas tergabung. Kemudian, teruskan kredensial ini ke Firebase Authentication SDK. Layanan backend kami selanjutnya akan memverifikasi kredensial tersebut dan menampilkan respons ke klien.

Setelah berhasil login, pengguna dapat mengakses informasi profil dasar pengguna dan admin dapat mengontrol akses pengguna ke data yang disimpan di produk Firebase lainnya. pengguna juga dapat menggunakan token autentikasi yang disediakan untuk memverifikasi identitas pengguna di layanan backend.

#### **2.5.4 Firebase Realtime Database**

Firebase Realtime Database adalah database yang di host dalam cloud. Data disimpan sebagai JSON dan disinkronkan secara realtime ke setiap klien yang terhubung. Ketika pengguna membuat aplikasi lintas platform dengan SDK Android, iOS, dan JavaScript, semua klien akan berbagai sebuah *Instance Realtime Database* dan menerima *update* data terbaru secara otomatis [27].

Firebase Realtime Database memungkinkan Anda untuk membuat aplikasi kolaboratif dan kaya fitur dengan menyediakan akses yang aman ke database, langsung dari kode sisi klien. Data disimpan di drive lokal. Bahkan saat offline sekalipun, peristiwa realtime terus berlangsung, sehingga pengguna akhir akan merasakan pengalaman yang responsif. Ketika koneksi perangkat pulih kembali, Realtime Database akan menyinkronkan perubahan data lokal dengan update jarak jauh yang terjadi selama klien offline, sehingga setiap perbedaan akan otomatis digabungkan.

Realtime Database menyediakan bahasa aturan berbasis ekspresi yang fleksibel, atau disebut juga Aturan Keamanan Firebase Realtime Database, untuk menentukan metode strukturisasi data dan kapan data dapat dibaca atau ditulis. Ketika diintegrasikan dengan Firebase Authentication, developer dapat menentukan siapa yang memiliki akses ke data tertentu dan bagaimana mereka dapat mengaksesnya. [27]

Realtime Database adalah database NoSQL, sehingga memiliki pengoptimalan dan fungsionalitas yang berbeda dengan database terkait. API Realtime Database dirancang agar hanya mengizinkan operasi yang dapat dijalankan dengan cepat. Hal ini memungkinkan Anda untuk membangun pengalaman realtime yang luar biasa dan dapat melayani jutaan pengguna tanpa mengorbankan kemampuan respons. Oleh karena itu, perlu dipikirkan bagaimana

pengguna mengakses data, kemudian buat struktur data sesuai dengan kebutuhan tersebut.

## **2.6 JSON (*JavaScript Object Notation*)**

JSON merupakan struktur data yang universal, dalam artian bisa digunakan dalam berbagai bahasa pemrograman. Hampir semua bahasa pemrograman mendukung penuh JSON dalam berbagai format. Hal ini memungkinkan format data yang dapat dipertukarkan menggunakan bahasa pemrograman juga menggunakan dasar dari struktur JSON. JSON merupakan format yang ringan untuk memasukan data kedalam sebuah variabel dalam format teks yang sepenuhnya independen tetapi menggunakan konvensi yang familiar dengan bahasa pemrograman, kelebihan ini lah yang membuat json menjadi sebuah bahasa data interchange yang ideal [28].

## **2.7 Push Notification**

Push notification ialah jenis pesan yang dikirim dari aplikasi ke perangkat, paling sering digunakan oleh aplikasi mobile untuk memberikan informasi yang berkaitan dengan penggunaannya. Push Notification muncul sebagai pesan pemberitahuan pada home screen perangkat. Pengguna tidak perlu membuka atau menjalankan aplikasi untuk menerima pemberitahuan dari aplikasi. Push Notification tersedia untuk semua sistem operasi mulai dari iOS, Android, dan banyak lagi yang menggunakan Push Notification tersebut [29].

## **2.8 PHP**

PHP adalah Bahasa server side scripting yang menyatu dengan HTML untuk membuat halaman web yang dinamis. Dengan demikian kode program yang ditulis dalam PHP tidak akan terlihat oleh user sehingga keamanan halaman web lebih terjamin. PHP dirancang untuk membuat halaman web yang dinamis, yaitu halaman web yang dapat membentuk suatu tampilan berdasarkan permintaan terkini, seperti menampilkan isi basis data ke halaman web.

PHP juga dapat berjalan pada berbagai web server seperti IIS (*Internet Information Server*), PWS (*Personal Web Server*), Apache, Xitami. PHP juga mampu berjalan di banyak sistem operasi yang beredar saat ini, diantaranya Sistem

Operasi Microsoft Windows (semua versi), Linux, Mac Os, Solaris. PHP dapat dibangun sebagai modul web server Apache dan sebagai binary yang dapat berjalan sebagai CGI (*Common Gateway Interface*).

PHP dapat mengirim HTTP header, dapat mengatur cookies , mengatur authentication dan redirect user. Salah satu keunggulan yang dimiliki PHP adalah kemampuannya untuk melakukan koneksi ke berbagai macam software sistem manajemen basis data atau Database Management Sistem (DBMS), sehingga dapat menciptakan suatu halaman web dinamis. PHP mempunyai koneksitas yang baik dengan beberapa DBMS seperti Oracle, Sybase, mSQL, MySQL, Microsoft SQL Server, Solid, PostgreSQL. Hampir seluruh aplikasi berbasis web dapat dibuat dengan PHP. Namun kekuatan utama adalah konektivitas basis data dengan web. Dengan kemampuan ini kita akan mempunyai suatu sistem basis data yang dapat diakses [30].

## **2.9 API**

API (Application Programming Interface) adalah sebuah teknologi untuk memfasilitasi pertukaran informasi atau data antara dua atau lebih aplikasi perangkat lunak. API sebagai antarmuka virtual antara dua fungsi perangkat lunak yang saling bekerja sama, seperti antara sebuah word processcor dan spreadsheet. Sebuah API tersedia untuk sistem windowing, sistem file, sistem basis data dan sistem jaringan.

Perkembangan teknologi API berawal dari penciptaan sebuah subrutin sederhana yang menyediakan kemampuan interoeperability dan system modifiability untuk mendukung pertukaran data antar beberapa aplikasi. Subrutin tersebut hanya mampu melaksanakan perhitungan matematis sederhana hingga terbentuk sebuah API pustaka perhitungan yang hampir selalu ada pada setiap bahasa pemrograman. Dari sebuah subrutin sederhana tersebut mulai bermunculan gagasan-gagasan bagaimana API harus dikembangkan khususnya sejalan dengan berkembangnya paradigma pemrograman berorientasi objek yang mengakibatkan sekumpulan subrutin sejenis dapat dikumpulkan menjadi sebuah kelas pembungkus subrutin-subrutin tersebut [31].

## 2.10 UML

UML adalah bahasa untuk menspesifikasi, memvisualisasi, membangun dan mendokumentasikan *artifacts* (bagian dari informasi yang digunakan untuk dihasilkan oleh proses pembuatan perangkat lunak, *artifact* tersebut dapat berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak, seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya. Selain itu UML adalah bahasa pemodelan yang menggunakan konsep orientasi object. UML dibuat oleh Grady Booch, James Rumbaugh, dan Ivar Jacobson di bawah bendera Rational Software Corps. UML menyediakan notasi-notasi yang membantu memodelkan sistem dari berbagai perspektif. UML tidak hanya digunakan dalam pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan. [32]

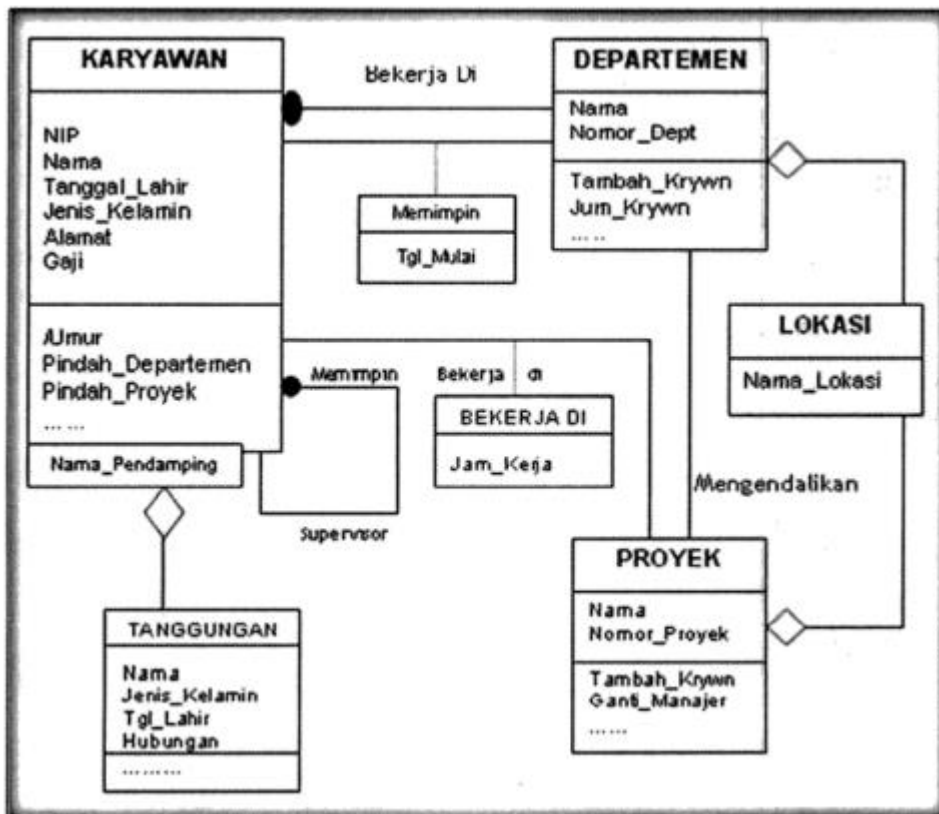
### 2.10.1 Activity Diagram

Menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktivitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktivitas lainnya seperti use case atau interaksi. Activity Diagram atau Diagram aktivitas adalah bentuk visual dari alur kerja yang berisi aktivitas dan tindakan, yang juga dapat berisi pilihan, atau pengulangan. Dalam Unified Modeling Language (UML), diagram aktivitas dibuat untuk menjelaskan aktivitas komputer maupun alur aktivitas dalam organisasi. Selain itu diagram aktivitas juga menggambarkan alur kontrol secara garis besar.

Diagram aktivitas memiliki komponen dengan bentuk tertentu, dihubungkan dengan tanda panah. Panah tersebut mengarahkan urutan aktivitas yang terjadi, dari awal sampai akhir. Yang perlu diperhatikan yaitu diagram aktivitas bukan menggambarkan aktivitas sistem yang dilakukan aktor, tetapi menggambarkan aktivitas yang dapat dilakukan oleh sistem. Activity diagram pada beberapa waktu dianggap sama seperti flowchart (diagram alur), namun meskipun diagram terlihat seperti sebuah diagram alur, tetapi sebenarnya berbeda. Activity diagram menunjukkan aliran yang berbeda seperti paralel, bercabang, bersamaan dan tunggal [33].

### 2.10.2 Class Diagram

Class diagram merupakan salah satu diagram utama dari UML untuk menggambarkan class atau blueprint object pada sebuah sistem. Analisis pembentukan class diagram merupakan aktivitas inti yang sangat mempengaruhi arsitektur piranti lunak yang dirancang hingga ke tahap pengkodean. Class diagram mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis. Class diagram juga menunjukkan properti dan operasi sebuah class dan batasan-batasan yang terdapat dalam hubungan-hubungan objek tersebut. UML menggunakan istilah fitur sebagai istilah umum yang meliputi properti dan operasi sebuah class [34]. Sebagai contoh untuk *class diagram* yang menggambarkan sebuah struktural dari sistem seperti pada gambar 2.5



Gambar 2.5 Contoh Class Diagram [35]

Pada gambar diatas menyatakan bahwa setiap kelas mempunyai keterkaitan, asosiasi diwakili oleh garis yang saling terhubung antar kelas tanpa tanda pada setiap ujung garis menyatakan kedua kelas terdapat multiplisitas pada sebuah relasi saling membutuhkan, kelas yang saling terkait dengan sebuah tanda panah hitam pada ujung garis disebut komposisi yaitu kelas tersebut tidak bisa berdiri sendiri,

agregasi dinyatakan dengan garis pada kedua kelas dengan tanda panah putih (kosong) menyatakan relasi antar kelas, garis putus putus menyatakan . Berikut penjelasan dari gambar 2.5:

1. Kelas Departemen membutuhkan kelas Karyawan dengan simbol garis dengan tanda panah hitam mengarah ke kelas karyawan.
2. Kelas Karyawan dan kelas Proyek saling membutuhkan, jika salah satu tidak ada maka kelas yang lain tidak dapat berdiri.
3. Kelas Tanggungan membutuhkan kelas karyawan, jika kelas karyawan tidak ada maka kelas Tanggungan tidak dapat digunakan.
4. Kelas Proyek dan kelas Departemen saling membutuhkan, jika salah satu tidak ada maka kelas tersebut tidak dapat digunakan.
5. Kelas lokasi membutuhkan kelas Departemen dan kelas Proyek, jika kelas Departemen dan kelas Proyek tidak ada maka kelas lokasi tidak dapat digunakan.
6. Kelas karyawan dan kelas Departemen menghasilkan kelas baru yaitu kelas Memimpin, dikarenakan antara kelas tersebut mempunyai relasi *many to many*
7. Kelas Karyawan dan kelas Proyek menghasilkan kelas baru yaitu kelas Bekerja, dikarenakan *many to many*.

### 2.10.3 Use Case Diagram

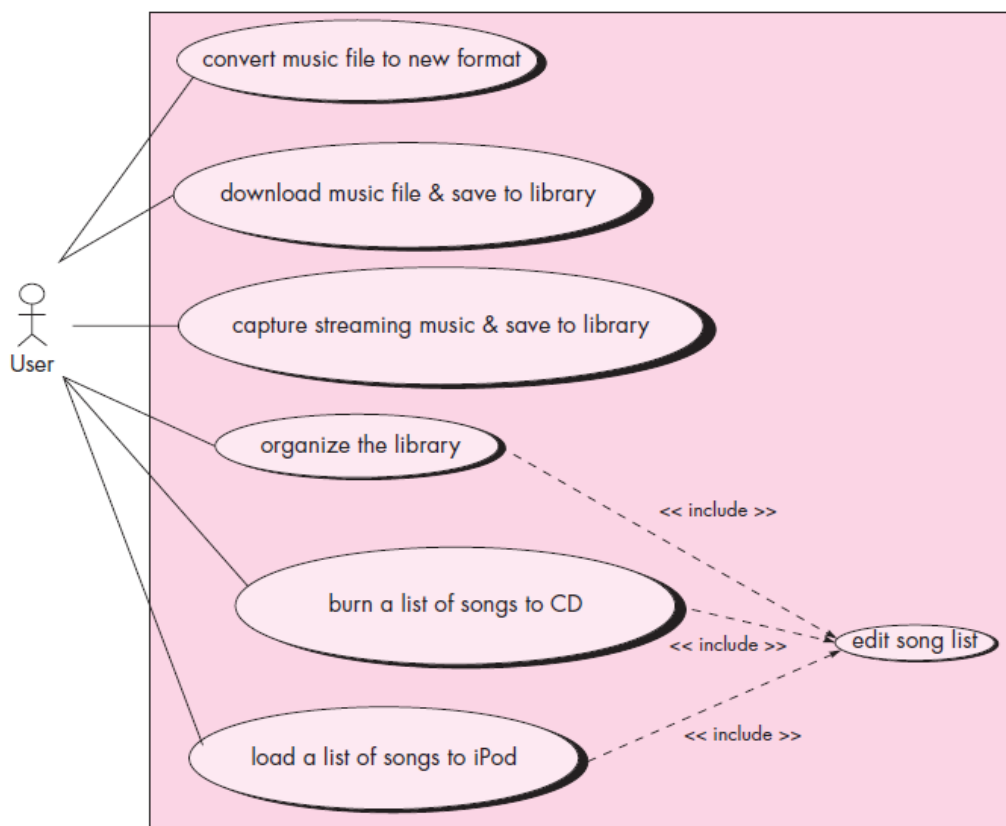
Use case adalah kegiatan atau urutan interaksi yang saling berkaitan antara sistem dan aktor. Use case bekerja dengan cara mendeskripsikan tipe interaksi antara user sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. Use case juga digunakan untuk membentuk perilaku (behaviour) sistem yang akan dibuat. Sebuah use case menggambarkan sebuah interaksi antara pengguna (aktor) dengan sistem yang ada [33].

Deskripsi dari use case sebagai berikut:

- a) Diagram use case merupakan pemodelan untuk menggambarkan kelakuan (*behavior*) sistem yang akan dibuat.
- b) Diagram use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat.

- c) Diagram use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Yang ditekankan pada diagram ini adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”.
- d) Sebuah use case merepresentasikan sebuah interaksi antara aktor (user atau sistem lainnya) dengan sistem.
- e) Use case menjelaskan secara sederhana fungsi sistem dari sudut pandang user.

Sebagai contoh dari use case seperti pada Gambar 2.6.



**Gambar 2.6 Use Case [33]**

Penjelasan logika dari use case diatas adalah:

1. User dapat secara langsung mengkonversi file musik ke format baru.
2. User dapat secara langsung mengunduh file musik dan dapat menyimpan ke *library*.
3. User dapat secara langsung streaming musik dan menyimpan musik tersebut ke *library*.



2. User dapat secara langsung mengatur *library*-nya dan dapat mengedit *list* dari lagu yang ada di *library* jika user ingin mengatur *library*-nya.
3. User dapat *burn* lagu ke CD dan dapat mengedit *list* dari lagu.
4. User dapat memuat lagu ke iPod dan dapat mengedit *list* lagu.

#### 2.10.4 Sequence Diagram

Pengertian Sequence Diagram merupakan salah satu dari diagram – diagram yang ada pada UML, sequence diagram ini adalah diagram yang menggambarkan kolaborasi dinamis antara sejumlah object. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara object juga interaksi antara object. Sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem. Sequence Diagram bisa digunakan untuk menjelaskan sebuah serangkaian langkah-langkah yang mengirimkan message antar satu lifeline ke lifeline yang lain. Setiap pesan yang dikirimkan bisa memberikan respon (return) relatif pada skenario yang dirancang di Use Case Diagram. Interaksi yang terjadi bisa bersifat instansiasi sebuah object maupun static method dari sebuah class [33].

Berikut komponen – komponen yang ada pada sequence diagram:

- a) *Object* adalah komponen berbentuk kotak yang mewakili sebuah class atau object. Mereka mendemonstrasikan bagaimana sebuah object berperilaku pada sebuah system.
- b) *Activation boxes* adalah komponen yang berbentuk persegi panjang yang menggambarkan waktu yang diperlukan sebuah object untuk menyelesaikan tugas. Lebih lama waktu yang diperlukan, maka activation boxes akan lebih panjang.
- c) *Actors* adalah komponen yang berbentuk stick figure. Komponen yang mewakili seorang pengguna yang berinteraksi dengan system.
- d) *Lifeline* adalah komponen yang berbentuk garis putus – putus. Lifeline biasanya memuat kotak yang berisi nama dari sebuah object. Berfungsi menggambarkan aktifitas dari object.

