BAB II
LANDASAN TEORI

2.1 Aplikasi

Perangkat lunak aplikasi adalah suatu subkelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna. Biasanya dibandingkan dengan perangkat lunak sistem yang mengintegrasikan berbagai kemampuan komputer, tapi tidak secara langsung menerapkan kemampuan tersebut untuk mengerjakan suatu tugas yang menguntungkan pengguna. Contoh utama perangkat lunak aplikasi adalah pengolah kata, lembar kerja, dan pemutar media.

Beberapa aplikasi yang digabung bersama menjadi suatu paket kadang disebut sebagai suatu paket atau suite aplikasi (*application suite*). Contohnya adalah *Microsoft Office* dan *OpenOffice.org*, yang menggabungkan suatu aplikasi pengolah kata, lembar kerja, serta beberapa aplikasi lainnya. Aplikasi-aplikasi dalam suatu paket biasanya memiliki antarmuka pengguna yang memiliki kesamaan sehingga memudahkan pengguna untuk mempelajari dan menggunakan setiap aplikasi. Sering kali, aplikasi ini memiliki kemampuan untuk saling berinteraksi satu sama lain sehingga menguntungkan pengguna. Contohnya, suatu lembar kerja dapat dibenamkan dalam suatu dokumen pengolah kata walaupun dibuat pada aplikasi lembar kerja yang terpisah.

2.2 Fasilitas Umum

Menurut Kamus Besar Bahasa Indonesia (KBBI) fasilitas umum adalah fasilitas yang disediakan untuk kepentingan umum. Namun biasanya fasilitas umum identik dengan pusat pelayanan masyarakat baik yang berkaitan dengan kebutuhan pemerintahan, perekonomian, ataupun kebutuhan kebutuhan yang lain. Biasanya fasilitas umum meliputi beberapa hal antara lain Rumah Sakit, SPBU, ATM, Restoran, Telepon Umum, dan lain sebagainya.

2.3 Data dan Informasi

Untuk menuju pada pengertian Sistem Informasi secara utuh, diperlukan pemahaman yang tepat tentang konsep data dan informasi. Keterkaitan data dan informasi sangatlah erat sebagaimana antara sebab dan akibat. Bahwa data merupakan dasar dari sebuah informasi, sedangkan informasi merupakan elemen yang dihasilkan dari suatu bentuk pengolahan data.

2.3.1 Data

Banyak pengertian data yang dirangkum dari berbagai sumber. Teguh Wahyono dalam bukunya Sistem Informasi: Kosep Dasar, Analisis Desain, dan Implementasi menjelaskan beberapa pengertian data sebagai berikut :

1. Menurut berbagai kamus bahasa Inggris-Indonesia, data diterjemahkan sebagai istilah yang berasal dari kata *"datum"* yang berarti fakta atau bahan-bahan keterangan.
2. Dari sudut pandang bisnis, terdapat pengertian data bisnis sebagai berikut: *"Business data is an organization's description of thing*

(resources) and events (transactions) that it face". Jadi data, dalam hal ini disebut sebagai data bisnis, merupakan deskripsi organisasi tentang sesuatu (*resources*) dan kejadian (*transactions*) yang terjadi.

3. Pengertian yang lain mengatakan bahwa *"data is the description of things and events that we face"*. Jadi data merupakan deskripsi dari sesuatu dan kejadian yang dihadapi.
4. Gordon B. Davis dalam bukunya *Management Informations System: Conceptual Foundations, Structures, And Development* menyebutkan data sebagai bahan mentah dari informasi, yang dirumuskan sebagai sekelompok lambang-lambang tidak acak yang menunjukkan jumlah atau tindakan atau hal-hal lain.

Dari keempat pengertian di atas, dapat diambil kesimpulan bahwa data adalah suatu bahan baku informasi, didefinisikan sebagai kelompok teratur simbol-simbol yang mewakili kuantitas, tindakan, benda, dan sebagainya. Data terbentuk dari karakter-karakter dapat berupa alfabet, angka, maupun simbol khusus seperti *, #, dan /. Data disusun untuk diolah dalam bentuk struktur data, struktur file, dan basis data.

2.3.2 Informasi

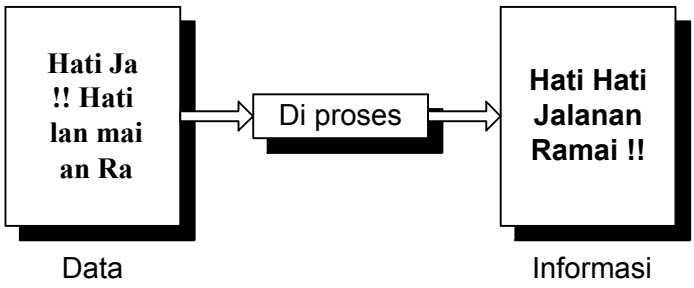
Masih dari Teguh Wahyono, dalam bukunya menjelaskan beberapa pengertian informasi dari berbagai sumber, sebagai berikut :

1. Menurut Gordon B. Davis dalam bukunya *Management Informations System: Conceptual Foundations, Structures, and Deveopment* menyebutkan informasi sebagai data yang telah diolah menjadi bentuk

yang berguna bagi penerimanya dan nyata, berupa nilai yang dapat dipahami di dalam keputusan sekarang maupun masa yang akan datang.

2. Menurut Barry E. Cushing dalam buku *Accounting Information System And Business Organization*, dikatakan bahwa informasi merupakan sesuatu yang menunjukan hasil pengolahan data yang diorganisasikan dan berguna kepada orang yang menerimanya.
3. Menurut Robert N. Anthony dan John Dearden dalam buku *Management Control System*, menyebutkan informasi sebagai suatu kenyataan, data, item yang menambah pengetahuan bagi penggunanya.

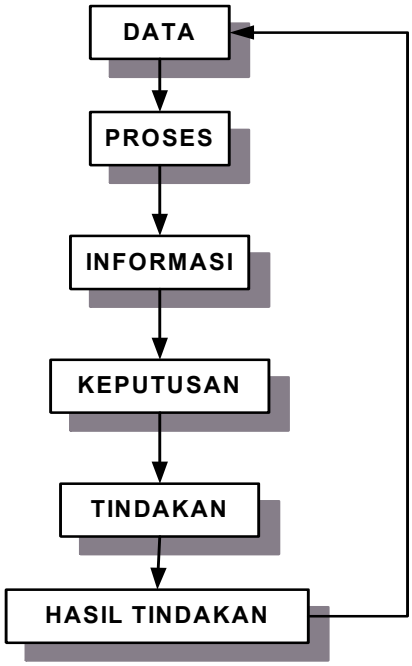
Dari beberapa pengertian di atas dapat disimpulkan bahwa informasi merupakan hasil dari pengolahan data menjadi bentuk yang lebih berguna bagi yang menerimanya yang menggambarkan suatu kejadian-kejadian nyata dan dapat digunakan sebagai alat Bantu untuk pengambilan suatu keputusan.



Gambar 2.1 Hubungan Data dan Informasi

2.3.3 Siklus Informasi

Pengolahan data menjadi suatu informasi dapat digambarkan sebagai sebuah siklus yang berkesinambungan seperti berikut:

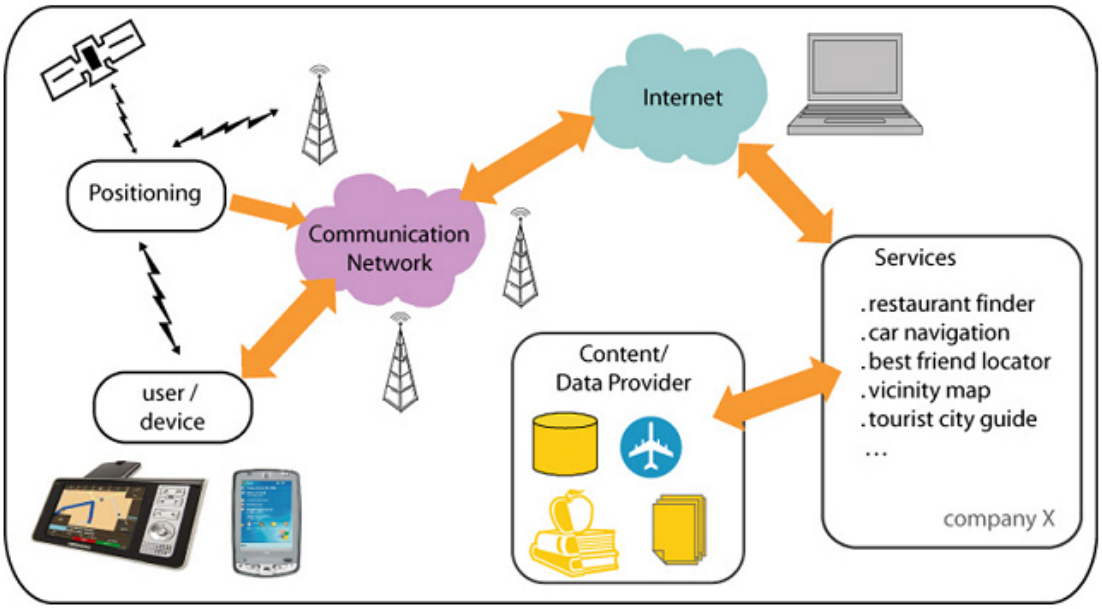


Gambar 2.2 Siklus Informasi

Secara sederhana dapat dikatakan bahwa data diolah menjadi suatu informasi. Dan pada tahapan selanjutnya, sebuah informasi akan menjadi data untuk terciptanya informasi yang lain.

2.4 *Location Based Service* (*LBS*)

Location Based Service (LBS) atau dalam bahasa indonesia diartikan sebagai Layanan Berbasis Lokasi merupakan suatu layanan yang bereaksi aktif terhadap perubahan entitas posisi sehingga mampu mendeteksi letak objek dan memberikan layanan sesuai dengan letak objek yang telah diketahui. LBS ini bekerja memanfaatkan lokasi dari devices untuk menyediakan informasi mengenai lokasi dengan memanfaatkan *Global Positioning System* (*GPS*). Untuk lebih jelasnya lihat gambar 2.3 dibawah ini.



Gambar 2.3 Cara Kerja *Location Based Services (LBS)*

2.4.1 *Component Location Based Service (LBS)*

Dalam menggunakan layanan berbasis lokasi elemen yang diperlukan *antara lain* :

1. *Mobile Devices* yaitu sebuah alat yang digunakan untuk meminta informasi yang dibutuhkan. Biasanya perangkat yang memungkinkan yaitu PDA, *Mobile Phones*, Laptop, dan perangkat lainnya yang mempunyai fasilitas navigasi.
2. *Communication Network* adalah jaringan selular yang mengirimkan data pengguna dan permintaan layanan.
3. *Positioning Component* untuk pengolahan layanan biasanya posisi pengguna harus ditentukan. Posisi pengguna dapat diperoleh menggunakan jaringan komunikasi atau dengan menggunakan *Global Positioning System* (*GPS*).

4. *Service and Application Provider* adalah penyedia layanan pengguna selular yang bertanggung jawab untuk memproses layanan.
5. *Data and Content Provider* yaitu penyedia layanan informasi data yang dapat diminta oleh pengguna.

2.5 *Wireless Mobile Network*

Pada gambar kerja *Location Based Services* komunikasi yang dilakukan oleh pengguna *devices* untuk transfer data dan pesan layanan permintaan data yaitu mengunakan jaringan nirkabel. Jaringan nirkabel untuk *mobile* sendiri yang umum saat ini dapat diklasifikasikan menjadi beberapa generasi anatara lain :

1. **Generasi 2G**

Evolusi perkembangan jaringan *wireless* dengan mode *circuit switch* ini lebih dikenal dengan terminologi generasi (xG). Terminologi ini muncul seiring dengan berkembangnya sistem komunikasi wireless dengan teknologi *"Global System for Mobile communication"* (GSM). GSM dikategorikan sebagai teknologi 2G.

Karakteristik dari teknologi 2G adalah sistem komunikasi wireless bergerak (*Mobile wireless*) dengan teknologi digital. GSM adalah standard yang dikeluarkan oleh *"European Telecommunication Standard Institute"* (ETSI). GSM beroperasi pada band frekuensi 900 MHz dan 1800 MHz. Dengan teknologi digital berbasis TDD/TDMA, kualitas suara yang dihasilkan lebih baik dibandingkan dengan sistem analog sebelumnya, yang kemudian dikategorikan sebagai generasi 1 (1G).

Walaupun layanan utamanya adalah *voice*, GSM juga dapat melayani trafik data dengan *rate* terbatas sebesar 9,6 Kbps. Dengan semakin meningkatnya kebutuhan trafik data, maka dengan mengkombinasikan 2 kanal GSM, *rate* data dapat ditingkatkan menjadi 28,8 Kbps. Teknik ini dilakukan pada mode *circuit switch* dan dikenal dengan *High Speed Circuit Switch Data* (GSM-HSCSD). Pelanggan dikenakan tarif berdasarkan lama/waktu koneksi, bukan pada besarnya data yang ditransfer (ini memang lumrah untuk trafik data pada mode jaringan CS).

Beberapa teknologi selain GSM yang masuk kategori 2G antara lain *Interim Standard-95* (IS-95) yang dikeluarkan oleh Qualcomm. Teknologi ini berbasis CDMA, sehingga dikenal juga dengan c*dma one*. Selain itu juga ada versi digital dari *Advance Mobile Phone System* (D-AMPS). Tidak seperti pendahulunya yang berbasis FDD/FDMA, D-AMPS sudah menggunakan teknologi digital TDMA. Sistem ini banyak digunakan di amerika utara. Di China juga berkembang teknologi *Personal Handy-phone System* (*PHS*), yang sebenarnya adalah telepon cordles tapi dengan kemampuan *handover* dan punya jangkauan yang lumayan jauh (belakangan banyak dijual Indonesia untuk *cordless* PSTN). Dan beberapa teknologi 2G lainnya.

2. Generasi 2.5G

Mengingat kebutuhan trafik data yang semakin meningkat (dengan tarif yang harus lebih murah) ditambah dengan berkembangnya teknologi 3G oleh kompetitor, maka diupayakan untuk mengadopsi mode *packet switch* ke teknologi GSM eksisting, dan dikenal dengan *General Packet Radio Service* (GSM/GPRS).

Data rate dapat ditingkatkan menjadi 115 Kbps. Bila sebelumnya 8 time slot dalam 1 kanal TDMA GSM digunakan untuk trafik suara 8 user, maka GPRS dapat menggunakan seluruh 8 time slot tadi untuk trafik data (1 time slot = 14,4 Kbps). Tapi pada prakteknya alokasi slot harus dibagi-bagi untuk *voice*, downlink dan uplink. Sehingga biasanya hanya 4 slot yang untuk *downlink* (57,6 Kbps) 1 slot untuk *uplink*, sisanya untuk *voice*.

Tapi ingat, ini semua bisa diatur oleh operatornya, tergantung kebijakan manajemen. Jadi jika layanan GPRS terkesan lambat bisa jadi cuma 1 slot untuk *downlink* data, sisanya untuk suara (pada masa lalu ARPU operator sebagian besar masih dari layanan *voice*, berbeda dengan kondisi sekarang dimana akses data sudah semakin tinggi). Karena sudah menggunakan mode *packet switch*, tarif (harusnya) ditentukan oleh jumlah data yang ditransfer, bukan oleh waktu koneksi lagi. Teknik ini mengubah teknologi GSM meningkat menjadi 2,5G.

Teknologi lain yang masuk kategori 2,5G adalah *Wideband Integrated Dispatch Enhanced Network* (WiDEN). Sistem ini dikembangkan oleh Motorola dan merupakan peningkatan dari iDEN (2G), dan banyak digunakan di berbagai

3. Generasi 2.75G

Khusus GPRS, karena sudah mendukung protocol IP, operator GSM selain bisa melayani publik juga bisa terkoneksi dengan jaringan *private* melalui *Interface* X.25 dan protokol TCP/IP. Dari sisi operator ini bisa meningkatkan layanan yang bernilai tambah *"Value Added Service"* (VAS). Mengingat VAS sangat potensial, maka perlu ditingkatkan *rate* datanya. Dengan mengubah teknik modulasi, maka data *rate* pada jaringan GSM dapat ditingkatnya menjadi 384

Kbps. Ini dikenal dengan istilah *"Enhance Data for GSM Evolution"* (GSM/EDGE), atau teknologi 2,75G.

Tinggal keputusan para operator, hitung-hitung untung rugi dari sisi bisnis, apakah tetap mempertahankan teknologi GSM/GPRS/EDGE nya atau lompat ke teknologi 3G yang lebih menjanjikan kecepatan, kapasitas, variasi layanan dan tarif yang murah. Teknologi lain dalam kategori 2,75G adalah CDMA2000 yang merupakan kelanjutan dari IS-95/cdmaOne. Standard CDMA2000 dikeluarkan oleh *3rd Generation Partnership Project 2* (3GPP2), sebuah consorsium dari ARIB/TTC (Jepang), TIA(USA), CCSA (China) dan TTA (Korea). Consorsium ini mengupayakan pengembangan generasi ke 3 dari teknologi generasi sebelumnya (2G).

Ada beberapa standard untuk teknologi CDMA2000, yaitu CDMA2000 1x, CDMA2000 3x, CDMA2000 1xEV-DO dan CDMA2000 1xEV-DV. Tapi untuk 1xEV DO dan 1xEV-DV sudah masuk kategori 3G.

4. Generasi 3G

Teknologi 3G terbagi menjadi GSM dan CDMA. Teknologi 3G sering disebut dengan *Mobile broadband* karena keunggulannya sebagai modem untuk internet yang dapat dibawa ke mana saja. Perkembangan teknologi 3G secara komersial dimulai pada Oktober, 2001, ketika NTTDoCoMo dari Jepang dengan teknologi W-CDMA menjual produknya untuk pertama kali secara terbatas. Kemudian disusul oleh SK Telecom, Korea Selatan pada tahun 2002 dengan teknologi 1xEV-DO, diikuti oleh KTF dari Korea Selatan dengan teknologi EV-DO. Keberhasilan layanan 3 G di kedua negara ini disebabkan oleh faktor

dukungan pemerintah. Pemerintah Jepang tidak mengenakan biaya di muka (*upfront fee*) atas penggunaan lisensi spektrum 3G atas operator-operator di Jepang (ada tiga operator: NTT Docomo, KDDI dan Vodafone). Sedangkan pemerintah Korea Selatan, walau pun mengenakan biaya di muka, memberikan insentif dan bantuan dalam pengembangan nirkabel pita lebar sebagai bagian dalam strategi pengembangan infrastruktur.

Di Eropa, dipelopori oleh British Telecom dan Telenor dengan teknologi W-CDMA pada Desember 2001. Di Amerika Serika jaringan 3G dipelopori oleh Monet *Mobile Networks* dengan teknologi CDMA20001xEV-DO, diikuti oleh Verizon *Wireless* pada tahun 2003. Di Australia jaringan 3G komersial pertama kali diperkenalkan oleh Hutchinson Telecommunication dengan nama *Three* pada bulan maret 2003. Pada bulan Desember 2007 jaringan 3G telah dioperasikan di 40 negara dan 154 jaringan HSDPA telah beroperasi di 71 negara, dan 200 juta pelanggan telah terhubung melalui jaringan 3G.

Perkembangan teknologi 3G mengharuskan pengaturan spektrum secara global, melalui penyediaan pita (band) yang lebih luas. Adanya teknologi 3G sebagai hasil pengembangan teknologi generasi kedua, yaitu hasil perkembangan evolusioner, yang masih menggunakan perangkat jaringan 2G yang diperluas dan hasil perkembangan revolusioner yang memerlukan jaringan dan alokasi frekuensi yang sama sekali baru. Secara evolusioner, IMT-2000 telah menerapkan dua macam evolusi ke 3G, yakni dari 2G CDMA standard IS-95 (cdmaOne) ke IMT-SC (cdma2000) dan dari 2G TDMA standars (GSM/IS-136) ke IMT-SC (EDGE).

Secara revolusioner, IMT-2000 membangun alokasi spektrum yang baru terkait tuntutan saluran yang makin luas.

5. Generasi 3.5G

Generasi 3,5G merupakan pengembangan dari 3G yang memungkinkan pengiriman data lebih cepat. Perbandingan antara 3G dan 3,5G terlihat jelas pada kecepatan transmisinya. Pada 3G, kecepatan transmisi maksimal 384kbps, sementara pada 3,5G kecepatan transmisi maksimal mencapai 3,6Mbps. Generasi 3G dan 3,5G mendukung layanan *video call* yang memungkinkan penelpon dan penerima saling bertatap muka.

6. Generasi 4G

Belakangan ini industri nirkabel mulai mengembangkan teknologi 4G, meskipun sebenarnya teknologi 4G ini seperti Long Term Evolution (LTE) hanya merupakan evolusi dari teknologi 3GPP dan Ultra *Mobile* Broadband (UMB) berasal dari 3GPP2, sehingga sulit untuk membedakan dengan jelas teknologi 3G dan 4 G. Salah satu teknolgoi 4G yaitu WiMax *Mobile* standard telah diterima oleh ITU untuk ditambahkan pada IMT-2000, sehingga teknologi baru ini masih digolongkan ke dalam keluarga 3G. International Telecommunication Union (ITU) sedang mempelajari kemampuan *Mobile* broadband yang disebut IMT-advanced yang disebut teknologi generasi keempat (4G). Diharapkan ITU segera melaksanakan penggunaan IMT-2000 (3G) dan IMT-Advanced (4G), konsekuensinya ITU harus menambah pita baik dibawah 1 GHz maupun diatas 2GHz.

2.6 *Google Maps*

Google Maps merupakan layanan web milik Google yang menyediakan pencarian peta *online*, informasi jalan, serta berbagai data geografis lainnya. Layanan web ini selain menyediakan platform pencarian geografis lokal, juga menyediakan informasi lokal lain seperti keadaan jalan, panduan arah berkendara, ataupun direktori bisnis yang cukup lengkap.

Google Maps menawarkan peta yang dapat digeser (*panned*), diperbesar (*zoom in*), diperkecil (*zoom out*), dapat diganti dalam beberapa mode (*maps*, satelit, *hybrid*, dan lain-lain), fitur pencarian rute (*routing*), penunjuk arah dari satu objek peta ke objek yang lain (*direction*) dan juga pencarian tempat (*place*). Sampai saat ini, *Google Maps* masih berada dalam tahap beta, dan masih terus dikembangkan dengan data yang selalu diperbarui secara berkala.

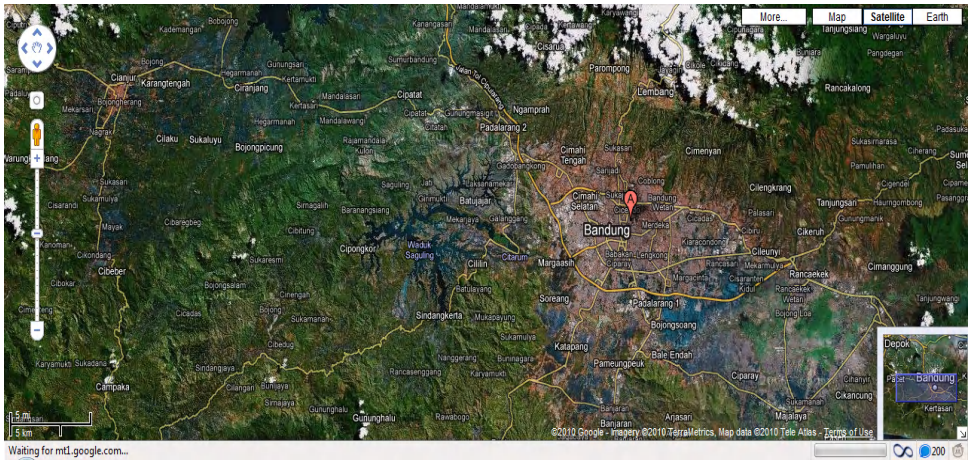
Google Maps merupakan hak cipta Google secara propriety, sehingga dalam menggunakannya memerlukan adanya perjanjian, Google membuat mekanisme untuk dapat mengakses *Google Maps* dengan coding aplikasi dengan sebuah kunci yang dikenal dengan *API Key*. Ada tiga jenis tampilan yang bisa dipilih dari Google Maps, yaitu:

1. *Map* menampilkan peta dalam bentuk peta garis. Map ditunjukan seperti pada gambar di bawah ini.



Gambar 2.4 Tampilan Bentuk *Map*

2. *Sattelite* menampilkan peta dalam bentuk citra/foto satelit. Satelit ditunjukan seperti pada gambar di bawah ini.



Gambar 2.5 Tampilan Bentuk *Sattelite*

3. *Eart* menampilkan peta dalam bentuk gabungan map dan satelit. *Eart* ditunjukan seperti pada gambar di bawah ini.



Gambar 2.6 Tampilan Bentuk *Eart*

2.7 *Global Positioning System (GPS)*

GPS adalah Sistem radio navigasi dan penentuan posisi menggunakan satelit yang dimiliki dan dikelola oleh Amerika Serikat. GPS didesain untuk memberikan posisi dan kecepatan 3D serta informasi mengenai waktu secara kontinyu, dan dapat digunakan oleh banyak orang sekaligus. Saat ini GPS mulai banyak diaplikasikan di Indonesia, terutama terkait dengan aplikasi-aplikasi yang menuntut informasi tentang posisi.

GPS terdiri dari 3 segmen, yaitu :

a. Segmen angkasa atau *space segmen* = segmen satelit

   GPS adalah Stasiun radio di angkasa yang dilengkapi dengan antena-antena untuk mengirim dan menerima sinyal-sinyal gelombang. Sinyal-sinyal itu selanjutnya diterima oleh *receiver* GPS dipermukaan bumi dan digunakan untuk penentuan posisi, kecepatan maupun waktu. Selain itu satelit juga dilengkapi dengan peralatan untuk mengontrol tingkah laku

satelit, serta sensor-sensor untuk mendeteksi peledakan nuklir dan lokasinya.

b. Segmen Sistem Kontrol

Segmen ini berfungsi untuk mengontrol dan memantau operasional satelit dan memastikan bahwa satelit berfungsi sebagaiman mestinya .

Fungsi ini mencakup beberapa tugas :

1. Menjaga agar semua satelit masing-masing berada pada posisi orbit yang seharusnya (*station keeping*).
2. Memantau dan menjaga kondisi dari semua sub sistem satelit.
3. Memantau panel matahari satelit, level daya baterai.
4. Menentukan dan menjaga waktu sistem GPS.

c. Segmen Pengguna

Segmen pengguna terdiri dari para pengguna satelit GPS, baik di darat, laut maupun di angkasa . Alat penerima sinyal (*receiver*) diperlukan untuk menerima dan memperoses sinyal-sinyal dari satelit GPS untuk digunakan dalam penentuan posisi, kecepatan maupun waktu.

Komponen utama dari *receiver* GPS adalah :

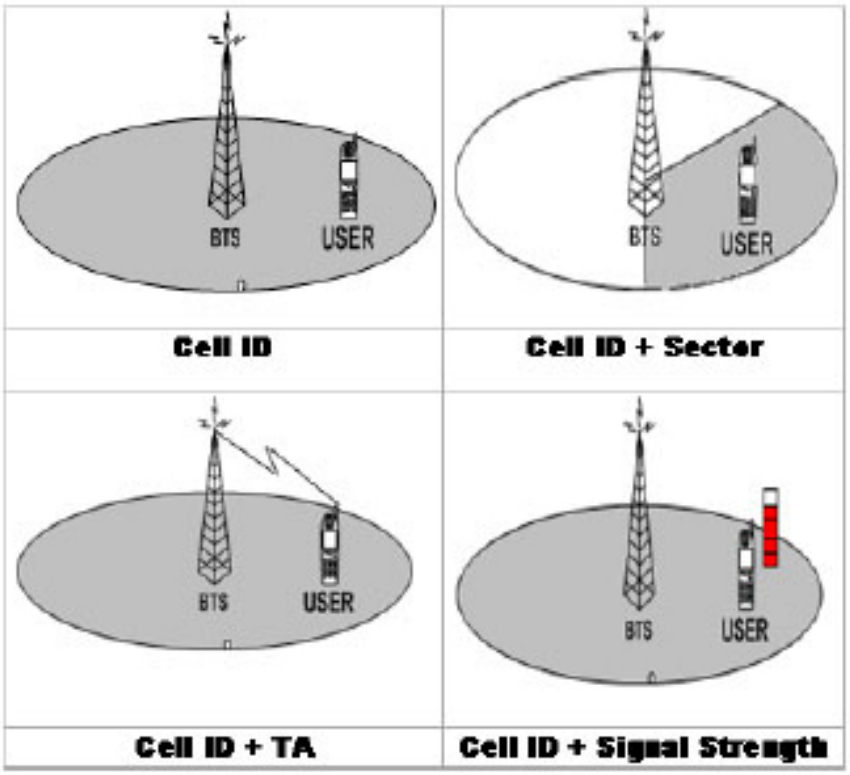
1. Antena dengan pre-amplifeier ;
2. Bagian RF (*Radio Frequency*) dengan pengidentifikasi sinyal dan pemroses sinyal;
3. Pemroses mikro untuk pengontrol *receiver*, data *sampling*, dan pemroses data
4. Osilator presesi

5. Catu daya
6. Unit perintah dan tampilan
7. Memori serta perekam data

2.8 Metode Penentuan Lokasi *Cell – ID (Cell Identification)*

Penentuan posisi didasarkan pada daerah geografis yang tercakup oleh sebuah *cell* berhubungan dengan daerah cakupan dari sinyal radio. Ketika sebuah *Mobile* terhubung secara aktif dengan sebuah base station, berarti *mobile* tersebut diasumsikan berada dalam *cell* dari *base station* tersebut. Untuk mengukur jarak dan arah *handset* dari *base station* tidak dapat diketahui dengan pasti. Oleh karena itu, untuk lebih meningkatkan lagi akurasi hasil pencarian, metode *Cell ID* ini seringkali dikombinasikan dengan metode lain misalnya :

1. *Timing Advanced* (*TA*), dengan menggunakan TA ini, metode *Cell ID* akan ditambahkan sebuah fungsionalitas untuk menghitung *Round Trip Time* (*RTT*), yaitu waktu transmisi sebuah frame (dari *base station* ke handphone) dan waktu penerimaan sebuah frame (dari handphone ke *base station*). Dengan tambahan metode ini, jarak antara handphone dan base station dapat ditentukan dengan keakuratan 50 m.
2. *Network Measurement Report* (*NMR*), dengan berdasar pada besar kecilnya sinyal (*Received Signal Strength*) yang diterima handphone yang ada di suatu "*sector cell*", maka posisi itu dapat ditentukan lebih akurat.



Gambar 2.7 Arsitektur *Cell ID*

2.9 Android

Android adalah sistem operasi untuk telepon seluler yang berbasis Linux. Android menyediakan platform terbuka bagi para pengembang buat menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam piranti bergerak. Awalnya, Google Inc. membeli Android Inc., pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan Android, dibentuklah Open Handset Alliance, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-*Mobile*, dan Nvidia.

2.9.1 Sejarah Android

Pada Juli 2000, Google bekerjasama dengan Android Inc., perusahaan yang berada di Palo Alto, California Amerika Serikat. Para pendiri Android Inc. bekerja pada Google, di antaranya Andy Rubin, Rich Miner, Nick Sears, dan Chris White. Saat itu banyak yang menganggap fungsi Android Inc. hanyalah sebagai perangkat lunak pada telepon seluler. Sejak saat itu muncul rumor bahwa Google hendak memasuki pasar telepon seluler. Di perusahaan Google, tim yang dipimpin Rubin bertugas mengembangkan program perangkat seluler yang didukung oleh kernel Linux. Hal ini menunjukkan indikasi bahwa Google sedang bersiap menghadapi persaingan dalam pasar telepon seluler. versi android terbaru yaitu versi 3.0. Android juga sudah bergabung dengan beberapa smart *Mobile* seperti Nokia, Sony Ericsson, dan lainnya.

Sekitar September 2007 sebuah studi melaporkan bahwa Google mengajukan hak paten aplikasi telepon seluler (akhirnya Google mengenalkan Nexus One, salah satu jenis telepon pintar GSM yang menggunakan Android pada sistem operasinya. Telepon seluler ini diproduksi oleh HTC Corporation dan tersedia di pasaran pada 5 Januari 2010).

Pada 9 Desember 2008, diumumkan anggota baru yang bergabung dalam program kerja Android ARM Holdings, Atheros Communications, diproduksi oleh Asustek Computer Inc, Garmin Ltd, Softbank, Sony Ericsson, Toshiba Corp, dan Vodafone Group Plc. Seiring pembentukan Open Handset Alliance, OHA mengumumkan produk perdana mereka, Android, perangkat bergerak (*Mobile*)

yang merupakan modifikasi kernel Linux 2.6. Sejak Android dirilis telah dilakukan berbagai pembaruan berupa perbaikan bug dan penambahan fitur baru.

Telepon pertama yang memakai sistem operasi Android adalah HTC Dream, yang dirilis pada 22 Oktober 2008. Pada penghujung tahun 2009 diperkirakan di dunia ini paling sedikit terdapat 18 jenis telepon seluler yang menggunakan Android.

1. **Android versi 1.1**

Pada 9 Maret 2009, Google merilis Android versi 1.1. Android versi ini dilengkapi dengan pembaruan estetis pada aplikasi, jam alarm, *voice search* (pencarian suara), pengiriman pesan dengan Gmail, dan pemberitahuan email.

2. **Android versi 1.5 (Cupcake)**

Pada pertengahan Mei 2009, Google kembali merilis telepon seluler dengan menggunakan Android dan SDK (*Software Development Kit*) dengan versi 1.5 (Cupcake). Terdapat beberapa pembaruan termasuk juga penambahan beberapa fitur dalam seluler versi ini yakni kemampuan merekam dan menonton video dengan modus kamera, mengunggah video ke Youtube dan gambar ke Picasa langsung dari telepon, dukungan Bluetooth A2DP, kemampuan terhubung secara otomatis ke headset Bluetooth, animasi layar, dan keyboard pada layar yang dapat disesuaikan dengan sistem.

3. **Android versi 1.6 (Donut)**

Donut (versi 1.6) dirilis pada September dengan menampilkan proses pencarian yang lebih baik dibanding sebelumnya, penggunaan baterai indikator dan kontrol applet VPN. Fitur lainnya adalah galeri yang memungkinkan

pengguna untuk memilih foto yang akan dihapus pada kamera, camcorder dan galeri yang dintegrasikan pada CDMA / EVDO, 802.1x, VPN, Gestures, dan *Text-to-speech engine*. Kemampuan dial kontak teknologi *text to change speech* (tidak tersedia pada semua ponsel.

4. Android versi 2.0/2.1 (Eclair)

Pada 3 Desember 2009 kembali diluncurkan ponsel Android dengan versi 2.0/2.1 (Eclair), perubahan yang dilakukan adalah pengoptimalan hardware, peningkatan *Google Maps* 3.1.2, perubahan UI dengan browser baru dan dukungan HTML5, daftar kontak yang baru, dukungan *flash* untuk kamera 3,2 MP, *digital Zoom*, dan Bluetooth 2.1.

Untuk bergerak cepat dalam persaingan perangkat generasi berikut, Google melakukan investasi dengan mengadakan kompetisi aplikasi *Mobile* terbaik (*killer apps* - aplikasi unggulan). Kompetisi ini berhadiah $25,000 bagi setiap pengembang aplikasi terpilih. Kompetisi diadakan selama dua tahap yang tiap tahapnya dipilih 50 aplikasi terbaik.

Dengan semakin berkembangnya dan semakin bertambahnya jumlah handset Android, semakin banyak pihak ketiga yang berminat untuk menyalurkan aplikasi mereka kepada sistem operasi Android. Aplikasi terkenal yang diubah ke dalam sistem operasi Android adalah Shazam, Backgrounds, dan WeatherBug. Sistem operasi Android dalam situs Internet juga dianggap penting untuk menciptakan aplikasi Android asli, contohnya oleh MySpace dan Facebook.

5. Android versi 2.2 (Froyo: Frozen Yoghurt)

Pada 20 Mei 2010, Android versi 2.2 (Froyo) diluncurkan. Perubahan-perubahan umumnya terhadap versi-versi sebelumnya antara lain dukungan Adobe Flash 10.1, kecepatan kinerja dan aplikasi 2 sampai 5 kali lebih cepat, intergrasi V8 JavaScript *engine* yang dipakai Google Chrome yang mempercepat kemampuan *rendering* pada browser, pemasangan aplikasi dalam SD Card, kemampuan *WiFi Hotspot portabel*, dan kemampuan auto update dalam aplikasi Android Market.

6. Android versi 2.3 (Gingerbread)

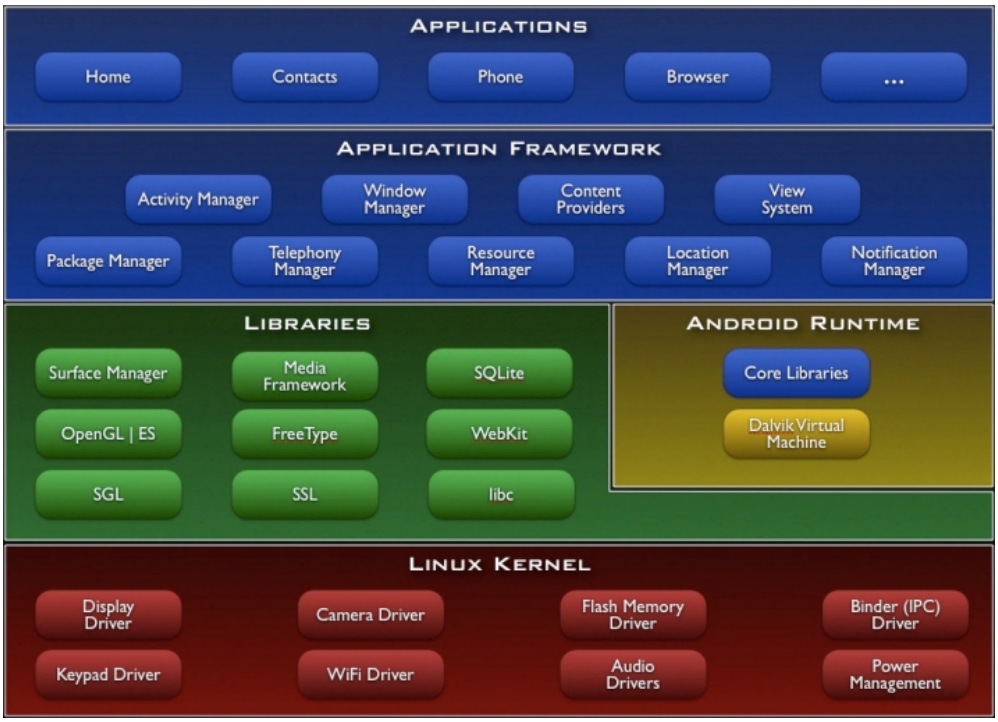
Pada 6 Desember 2010, Android versi 2.3 (Gingerbread) diluncurkan. Perubahan-perubahan umum yang didapat dari Android versi ini antara lain peningkatan kemampuan permainan (*gaming*), peningkatan fungsi *copy paste*, layar antar muka (*User Interface*) didesain ulang, dukungan format video VP8 dan WebM, efek audio baru *(reverb, equalization, headphone virtualization, dan bass boost)*, dukungan kemampuan *Near Field Communication (NFC)*, dan dukungan jumlah kamera yang lebih dari satu.

7. Android versi 3.0 (Honeycomb)

Android Honeycomb dirancang khusus untuk tablet. Android versi ini mendukung ukuran layar yang lebih besar. *User Interface* pada Honeycomb juga berbeda karena sudah didesain untuk *tablet*. Honeycomb juga mendukung multi prosesor dan juga akselerasi perangkat keras (*hardware*) untuk grafis. Tablet pertama yang dibuat dengan menjalankan Honeycomb adalah Motorola Xoom.

2.9.2 Anatomi Aplikasi Android

Dalam paket sistem operasi android tediri dari beberapa unsur seperti tampak pada gambar 2.8. Secara sederhana arsitektur android merupakan sebuah kernel Linux dan sekumpulan pustaka C / C++ dalam suatu *Framework* yang menyediakan dan mengatur alur proses aplikasi.



Gambar 2.8 *Detail* Anatomi Android

2.9.2.1 Linux Kernel

Android dibangun di atas kernel Linux 2.6. Namun secara keseluruhan android bukanlah linux, karena dalam android tidak terdapat paket standar yang dimiliki oleh linux lainnya. Linux merupakan sistem operasi terbuka yang handal dalam manajemen memori dan proses. Oleh karenanya pada android hanya terdapat beberapa servis yang diperlukan seperti keamanan, manajemen memori, manajemen proses, jaringan dan driver. Kernel linux menyediakan driver layar,

kamera, *keypad*, *WiFi, Flash Memory, audio*, dan IPC (*Interprocess Communication*) untuk mengatur aplikasi dan lubang keamanan.

2.9.2.2 Libraries

Android menggunakan beberapa paket pustaka yang terdapat pada C/C++ dengan standar *Berkeley Software Distribution* (BSD) hanya setengah dari yang aslinya untuk tertanam pada kernel Linux. Beberapa pustaka diantaranya:

1. *Media Library* untuk memutar dan merekam berbagai macam format audio dan video.
2. *Surface Manager* untuk mengatur hak akses layer dari berbagai aplikasi.
3. *Graphic Library* termasuk didalamnya *SGL* dan *OpenGL*, untuk tampilan 2D dan 3D.
4. *SQLite* untuk mengatur relasi database yang digunakan pada aplikasi.
5. *SSl* dan *WebKit* untuk browser dan keamanan internet.

Pustaka-pustaka tersebut bukanlah aplikasi yang berjalan sendiri, namun hanya dapat digunakan oleh program yang berada di level atasnya. Sejak versi Android 1.5, pengembang dapat membuat dan menggunakan pustaka sendiri menggunakan *Native Development Toolkit* (NDK).

2.9.2.3 Android Runtime

Pada android tertanam paket pustaka inti yang menyediakan sebagian besar fungsi android. Inilah yang membedakan Android dibandingkan dengan sistem operasi lain yang juga mengimplementasikan Linux. *Android Runtime* merupakan mesin virtual yang membuat aplikasi android menjadi lebih tangguh

dengan paket pustaka yang telah ada. Dalam Android Runtime terdapat 2 bagian utama, diantaranya:

1. Pustaka Inti, android dikembangkan melalui bahasa pemrograman Java, tapi *Android Runtime* bukanlah mesin virtual Java. Pustaka inti android menyediakan hampir semua fungsi yang terdapat pada pustaka Java serta beberapa pustaka khusus android.
2. Mesin Virtual Dalvik, Dalvik merupakan sebuah mesin virtual yang dikembangkan oleh Dan Bornstein yang terinspirasi dari nama sebuah perkampungan yang berada di Iceland. Dalvik hanyalah interpreter mesin virtual yang mengeksekusi file dalam format *Dalvik Executable* (*.dex). Dengan format ini Dalvik akan mengoptimalkan efisiensi penyimpanan dan pengalamatan memori pada file yang dieksekusi. Dalvik berjalan di atas kernel Linux 2.6, dengan fungsi dasar seperti *threading* dan manajemen memori yang terbatas.

2.9.2.4 Aplication Framework

Kerangka aplikasi menyediakan kelas-kelas yang dapat digunakan untuk mengembangkan aplikasi android. Selain itu, juga menyediakan abstraksi generik untuk mengakses perangkat, serta mengatur tampilan *User Interface* dan sumber daya aplikasi. Bagian terpenting dalam kerangka aplikasi android adalah sebagai berikut:

1. *Activity Manager*, berfungsi untuk mengontrol siklus hidup aplikasi dan menjaga keadaan ”*Backstack“* untuk navigasi penggunaan.

2. *Content Providers*, berfungsi untuk merangkum data yang memungkinkan digunakan oleh aplikasi lainnya, seperti daftar nama.
3. *Resuource Manager*, untuk mengatur sumber daya yang ada dalam program. Serta menyediakan akses sumber daya diluar kode program, seperti karakter, grafik, dan *file layout*.
4. *Location Manager,* berfungsi untuk memberikan informasi detail mengenai lokasi perangkat android berada.
5. *Notification Manager,* mencakup berbagai macam peringatan seperti, pesan masuk, janji, dan lain sebagainya yang akan ditampilkan pada *status bar*.

2.9.2.5 Aplication Layer

Puncak dari diagram arsitektur android adalah lapisan aplikasi dan *widget*. Lapisan aplikasi merupakan lapisan yang paling tampak pada pengguna ketika menjalankan program. Pengguna hanya akan melihat program ketika digunakan tanpa mengetahui proses yang terjadi dibalik lapisan aplikasi. Lapisan ini berjalan dalam *Android runtime* dengan menggunakan kelas dan service yang tersedia pada *Framework* aplikasi.

Lapisan aplikasi android sangat berbeda dibandingkan dengan sistem operasi lainnya. Pada android semua aplikasi, baik aplikasi inti (*native*) maupun aplikasi pihak ketiga berjalan diatas lapisan aplikasi dengan menggunakan pustaka API (*Application Programming Interface*) yang sama.

2.9.3 Komponen Aplikasi Android

Fitur penting android adalah bahwa satu aplikasi dapat menggunakan elemen dari aplikasi lain (untuk aplikasi yang memungkinkan). Sebagai contoh, sebuah aplikasi memerlukan fitur *scroller* dan aplikasi lain telah mengembangkan fitur *scroller* yang baik dan memungkinkan aplikasi lain menggunakannya. Maka pengembang tidak perlu lagi mengembangkan hal serupa untuk aplikasinya, cukup menggunakan *scroller* yang telah ada.

Agar fitur tersebut dapat bekerja, sistem harus dapat menjalankan aplikasi ketika setiap bagian aplikasi itu dibutuhkan, dan pemanggilan objek java untuk bagian itu. Oleh karenanya android berbeda dari sistem-sistem lain, Android tidak memiliki satu tampilan utama program seperti fungsi main() pada aplikasi lain. Sebaliknya, aplikasi memiliki komponen penting yang memungkinkan sistem untuk memanggil dan menjalankan ketika dibutuhkan.

2.9.3.1 Activities

Activity merupakan bagian yang paling penting dalam sebuah aplikasi, karena *Activity* menyajikan tampilan visual program yang sedang digunakan oleh pengguna. Setiap *Activity* dideklarasikan dalam sebuah kelas yang bertugas untuk menampilkan antarmuka pengguna yang terdiri dari *Views* dan respon terhadap *Event*. Setiap aplikasi memiliki sebuah *activity* atau lebih. Biasanya pasti akan ada *activity* yang pertama kali tampil ketika aplikasi dijalankan.

Perpindahan antara *activity* dengan *activity* lainnya diatur melalui sistem, dengan memanfaatkan *activity stack.* Keadaan suatu *activity* ditentukan oleh

posisinya dalam tumpukan *acitivity*, LIFO (*Last In First Out*) dari semua aplikasi yang sedang berjalan. Bila suatu *activity* baru dimulai, *activity* yang sebelumnya digunakan maka akan dipindahkan ketumpukan paling atas. Jika pengguna ingin menggunakan *activity* sebelumnya, cukup menekan tombol *Back*, atau menutup *activity* yang sedang digunakan, maka *activity* yang berada diatas akan aktif kembali. *Memory Manager* android menggunakan tumpukkan ini untuk menentukan prioritas aplikasi berdasarkan *activity*, memutuskan untuk mengakhiri suatu aplikasi dan mengambil sumber daya dari aplikasi tersebut.

Ketika *activity* diambil dan disimpan dalam tumpukkan *activity* terdapat 4 kemungkinan kondisi transisi yang akan terjadi :

1) ***Active*,** setiap *activity* yang berada ditumpukan paling atas, maka dia akan terlihat, terfokus, dan menerima masukkan dari pengguna. Android akan berusaha untuk membuat *activity* aplikasi ini untuk untuk tetap hidup dengan segala cara, bahkan akan menghentikan *activity* yang berada dibawah tumpukkannya jika diperlukan. Ketika *activity* sedang aktif, maka yang lainnya akan dihentikan sementara.
2) ***Paused*,** dalam beberapa kasus *activity* akan terlihat tapi tidak terfokus pada kondisi inilah disebut *paused.* Keadaan ini terjadi jika *activity* transparan dan tidak *fullscreen* pada layar. Ketika *activity* dalam keadaan *paused*, dia terlihat *active* namun tidak dapat menerima masukkan dari pengguna. Dalam kasus ekstrim, android akan menghentikan *activity* dalam keadaan *paused* ini, untuk menunjang sumber daya bagi *activity* yang sedang aktif.

3) ***Stopped*****,** ketika sebuah *activity* tidak terlihat, maka itulah yang disebut *stopped. Activity* akan tetap berada dalam memori dengan semua keadaan dan informasi yang ada. Namun akan menjadi kandidat utama untuk dieksekusi oleh sistem ketika membutuhkan sumberdaya lebih. Oleh karenanya ketika suatu *activity* dalam kondisi *stopped* maka perlu disimpan data dan kondisi antarmuka saat itu. Karena ketika *activity* telah keluar atau ditutup, maka dia akan menjadi *inactive.*
4) ***Inactive*****,** kondisi ketika *activity* telah dihentikan dan sebelum dijalankan. *Inactive activity* telah ditiadakan dari tumpukan *activity* sehingga perlu *restart* ulang agar dapat tampil dan digunakan kembali.

Kondisi transisi ini sepenuhnya ditangani oleh manajer memori android. Android akan memulai menutup aplikasi yang mengandung *activity inactive*, kemudian *stopped activity*, dan dalam kasus luar biasa *paused activity* juga akan di tutup.

2.9.3.2 Services

Suatu *service* tidak memiliki tampilan antarmuka, melainkan berjalan di *background* untuk waktu yang tidak terbatas. Komponen *service* diproses tidak terlihat, memperbarui sumber data dan menampilkan notifikasi. *Service* digunakan untuk melakukan pengolahan data yang perlu terus diproses, bahkan ketika *Activity* tidak aktif atau tidak tampak.

2.9.3.3 Intents

Intens merupakan sebuah mekanisme untuk menggambarkan tindakan tertentu, seperti memilih foto, menampilkan halaman web, dan lain sebagainya. *Intents* tidak selalu dimulai dengan menjalankan aplikasi, namun juga digunakan oleh sistem untuk memberitahukan ke aplikasi bila terjadi suatu hal, misal pesan masuk. *Intents* dapat eksplisit atau implisit, contohnya jika suatu aplikasi ingin menampilkan *URL*, sistem akan menentukan komponen apa yang dibutuhkan oleh *Intents* tersebut.

2.9.3.4 Broadcast Receiver

Broadcast Receivers merupakan komponen yang sebenarnya tidak melakukan apa-apa kecuali menerima dan bereaksi menyampaikan pemberitahuan. Sebagian besar *Broadcast* berasal dari sistem misalnya, Batre sudah hampir habis, informasi zona waktu telah berubah, atau pengguna telah merubah bahasa *default* pada perangkat. Sama halnya dengan *service, Broadcast Receivers* tidak menampilkan antarmuka pengguna. Namun, *Broadcast Receivers* dapat menggunakan *Notification Manager* untuk memberitahukan sesuatu kepada pengguna.

2.9.3.5 Content Providers

Content Providers digunakan untuk mengelola dan berbagi database. Data dapat disimpan dalam file sistem, dalam database *SQLite,* atau dengan cara lain yang pada prinsipnya sama. Dengan adanya *Content Provider* memungkinkan antar aplikasi untuk saling berbagi data. Komponen ini sangat berguna ketika

sebuah aplikasi membutuhkan data dari aplikasi lain, sehingga mudah dalam penerapannya.

2.9.4 Tipe Aplikasi Android

Terdapat tiga kategori aplikasi pada android :

1. ***Foreground Activity***

   Aplikasi yang hanya dapat dijalankan jika tampil pada layar dan tetap efektif walaupun tidak terlihat. Aplikasi dengan tipe ini pasti mempertimbangkan siklus hidup *activity*, sehingga perpindahan antar *activity* dapat berlangsung dengan lancar.

2. ***Background Service***

   Aplikasi yang memiliki interaksi terbatas dengan user, selain dari pengaturan konfigurasi, semua dari prosesnya tidak tidak tampak pada layar. Contohnya aplikasi penyaringan panggilan atau sms auto respon.

3. ***Intermittent Activity***

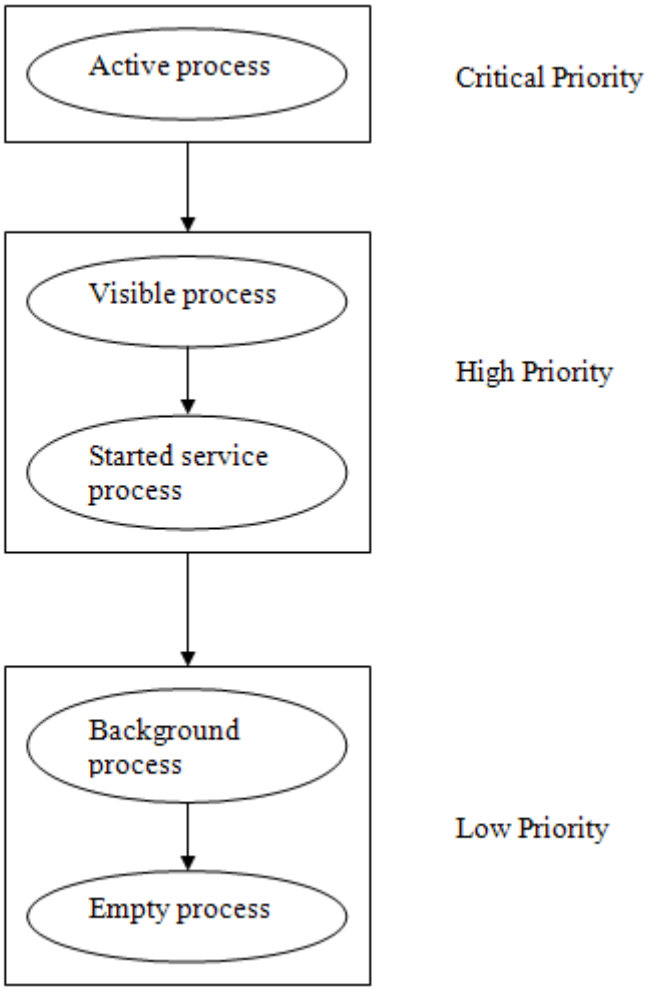
   Aplikasi yang masih membutuhkan beberapa masukkan dari pengguna, namun sebagian sangat efektif jika dijalankan di *background* dan jika diperlukan akan memberi tahu pengguna tentang kondisi tertentu. Contohnya pemutar musik.

Untuk aplikasi yang kompleks akan sulit untuk menentukan kategori aplikasi tersebut apalagi aplikasi memiliki ciri-ciri dari semua kategori. Oleh karenanya perlu pertimbangan bagaimana aplikasi tersebut digunakan dan menentukan kategori aplikasi yang sesuai.

2.9.5 Siklus Hidup Android

Siklus hidup aplikasi android dikelola oleh sistem, berdasarkan kebutuhan pengguna, sumberdaya yang tersedia, dan sebagainya. Misalnya Pengguna ingin menjalankan browser web, pada akhirnya sistem yang akan menentukan menjalankan aplikasi. Sistem sangat berperan dalam menentukan apakah aplikasi dijalankan, dihentikan sementara, atau dihentikan sama sekali. Jika pengguna ketika itu sedang menjalankan sebuah *Activity*, maka sistem akan memberikan perioritas utama untuk aplikasi yang tersebut. Sebaliknya, jika suatu *Activity* tidak terlihat dan sistem membutuhkan sumber daya yang lebih, maka *Activity* yang prioritas rendah akan ditutup.

Android menjalankan setiap aplikasi dalam proses secara terpisah, yang masing-masing memiliki mesin virtual pengolah sendiri, dengan ini melindungi penggunaan memori pada aplikasi. Selain itu juga android dapat mengontrol aplikasi mana yang layak menjadi prioritas utama. Karenanya android sangat sensitive dengan siklus hidup aplikasi dan komponen-komponennya. Perlu adanya penanganan terhadap setiap kondisi agar aplikasi menjadi stabil. Gambar 2.9 menunjukkan prioritas dari aplikasi.



Gambar 2.9 Prioritas aplikasi berdasarkan *activity*

2.9.6 Kelebihan Android

Sudah banyak *platform* untuk perangkat selular saat ini, termasuk didalamnya Symbian, iPhone, Windows *Mobile*, BlackBerry, Java *Mobile* Edition, Linux *Mobile* (LiM), dan banyak lagi. Namun ada beberapa hal yang menjadi kelebihan Android. Walaupun beberapa fitur-fitur yang ada telah muncul sebelumnya pada platform lain, Android adalah yang pertama menggabungkan hal seperti berikut :

1. Keterbukaan, Bebas pengembangan tanpa dikenakan biaya terhadap sistem karena berbasiskan Linux dan *open source*. Pembuat perangkat menyukai hal ini karena dapat membangun *platform* yang sesuai yang diinginkan tanpa harus membayar royality. Sementara pengembang *software* menyukai karena android dapat digunakan diperangkat manapun dan tanpa terikat oleh *vendor* manapun.
2. Arsitektur komponen dasar android terinspirasi dari teknologi internet *Mashup*. Bagian dalam sebuah aplikasi dapat digunakan oleh aplikasi lainnya, bahkan dapat diganti dengan komponen lain yang sesuai dengan aplikasi yang dikembangkan.
3. Banyak dukungan *service*, kemudahan dalam menggunakan berbagai macam layanan pada aplikasi seperti penggunaan layanan pencarian lokasi, database SQL, browser dan penggunaan peta. Semua itu sudah tertanam pada android sehingga memudahkan dalam pengembangan aplikasi.
4. Siklus hidup aplikasi diatur secara otomatis, setiap program terjaga antara satu sama lain oleh berbagai lapisan keamanan, sehingga kerja sistem menjadi lebih stabil. Pengguna tak perlu kawatir dalam menggunakan aplikasi pada perangkat yang memorinya terbatas.
5. Dukungan grafis dan suarat terbaik, dengan adanya dukungan 2D grafis dan animasi yang diilhami oleh *Flash* menyatu dalam 3D menggunakan *OpenGL* memungkinkan membuat aplikasi maupun game yang berbeda.

6. Portabilitas aplikasi, aplikasi dapat digunakan pada perangkat yang ada saat ini maupun yang akan datang. Semua program ditulis dengan menggunakan bahas pemrograman Java dan dieksekusi oleh mesin virtual Dalvik, sehingga kode program portabel antara ARM, X86, dan arsitektur lainnya. Sama halnya dengan dukungan masukan seperti penggunaan *Keyboard,* layar sentuh, *trackball* dan resolusi layar semua dapat disesuaikan dengan program.

2.10 Aplikasi Android Berbasis Lokasi

Berikut ini merupakan salah satu contoh aplikasi yang dikembangkan pada *platform* android, dan memiliki kesamaan tujuan dengan aplikais yang akan dibangun pada tugas akhir ini. Aplikasi *WHERE*^{TM} yaitu aplikasi android berbasis GPS ini memberikan informasi mengenai restoran, bioskop, dan pom bensin terdekat. Selain itu aplikasi ini memberikan informasi mengenai cuaca, keadaan lalu lintas, dan informasi buku kuning. Aplikasi yang dikembangkan oleh *Ulocate Communication Inc* ini hanya dapat digunakan untuk wilayah Amerika Serikat.

2.11 *Rational Unified Process (RUP)*

Rational Unified Process merupakan suatu metode rekayasa perangkat lunak yang dikembangkan dengan mengumpulkan berbagai *best practises* yang terdapat dalam industri pengembangan perangkat lunak. Ciri utama metode ini adalah menggunakan *use-case driven* dan pendekatan iteratif untuk siklus pengembangan perankat lunak.

2.12 Konsep Perancangan Berorientasi Obyek

Teknologi objek menganalogikan sistem aplikasi seperti kehidupan nyata yang didominasi oleh objek. Didalam membangun sistem berorientasi objek akan menjadi lebih baik apabila langkah awalnya didahului dengan proses analisis dan perancangan yang berorientasi objek. Tujuannya adalah mempermudah *programmer* didalam mendesain program dalam bentuk objek-objek dan hubungan antar objek tersebut untuk kemudian dimodelkan dalam sistem nyata.
Suatu perusahaan *software* yaitu *Rational Software*, telah membentuk konsarium dengan berbagai organisasi untuk meresmikan pemakaian *Unifed Modelling Language (UML)* sebagai bahasa standar dalam *Object Oriented Analysist Design (OOAD).*

2.12.1 *Unified Modelling Language* (*UML*)

UML dalam sebuah bahasa untuk menentukan visualisasi, konstruksi, dan mendokumentasikan artifacts dari sistem *software*, untuk memodelkan bisnis, dan sistem non-*software* lainnya. UML merupakan sistem arsitektur yang bekerja dalam OOAD dengan satu bahasa yang konsisten untuk menentukan, visualisasi, konstruksi dan mendokumentasikan artifact yang terdapat dalam sistem. Artifact adalah sepotong informasi yang digunakan atau dihasilkan dalam suatu proses rekayasa *software*. *Artifact* dapat berupa model, deskripsi atau *software*.

2.12.2 *Use Case Diagram*

Use Case Diagram menjelakan manfaat sistem jika dilihat menurut pandangan orang yang berada diluar sistem (Aktor). Diagram ini menunjukan

fungsionalitas suatu sistem yang berinteraksi dengan dunia luar. *Use Case Diagram* dapat digunakan selama proses analisis untuk menagkap requitment sistem dan untuk memahami bagaimana sistem bekerja.

2.12.3 *Class Diagram*

Class Diagram menjelaskan dalam visualisasi struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. *Class Diagram* memperlihatakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas dalam model desain dari suatu sistem.

Selama proses analisis, *class diagram* memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan prilaku sistem. Selama tahap desain, *class diagram* berperan dalam menagkap struktur dari semua kelas yang membentuk arsitektur sistem yang dibuat.

2.12.4 *Behavior Diagram*

Behavior diagram dapat dikelompokan menjadi tiga diagram, yaitu :

a. *Statechart Diagram*

   Statechart Diagram berfungsi untuk memodelkan prilaku dinamis satu kelas satu objek.

b. *Activity Diagram*

   Activity Diagram memodelkan alur kerja (work flow) sebuah proses bisnis dan urutan aktifitas dalam suatu proses.

c. Interaction Diagram

   Interaction Diagram dibagi menjadi dua model diagram yaitu :

1. *Sequence Diagram* menjelaskan interaksi objek yang disusun dalam suatu urutan waktu. Diagram ini secara khusus bersosialisasi dengan *use case*. Sequence diagram, memperlihatkan tahap demi tahap apa yang seharusnya terjadi untuk menghasilkan sesuatu dalam *use case*.
2. *Colaboration Diagram* melihat pada interaksi dan hubungan terstruktur antar objek. Tipe diagram ini menekankan pada hubungan (*relationship*) antar objek, sedangkan *sequence diagram* menekankan pada urutan kejadian. Dalam *collaboration diagram* terdapat beberapa objek, *link*, dan *message*.

2.12.5 *Implementatiton Diagram*

Implementation diagram dibagi menjadi dua diagram, yaitu :

1. *Component Diagram* menggambarkan alokasi semua kelas dan objek kedalam komponen-komponen dalam desain fisik sistem *software*. Diagram ini memperlihatkan pengaturan dan kebergantungan antara komponen-komponen *software*, seperti *source code*, *binary code*, dan komponen tereksekusi (*execute components*)
2. *Deployment Diagram* memperlihatkan pemetaan *software* kepada *hardware*. Dimana akan berjalan (di *server/multitier*, *standalone* atau lainnya), dan menggambarkan model koneksi dan kemampuan jaringan dan hal lainya yang bersifat fisik.

2.13 MySQL

MySQL adalah sebuah sistem manajemen database relasi *(relational database management system)* yang bersifat "terbuka" *(open source)*. Terbuka maksudnya adalah MySQL boleh di-*download* oleh siapa saja, baik versi kode program aslinya maupun versi binernya dan dapat digunakan secara (relatif) gratis baik untuk dimodifikasi sesuai dengan kebutuhan seseorang maupun sebagai suatu program aplikasi komputer.

2.13.1 Sejarah MySQL

Sejarah MySQL yang merupakan hasil buah pikiran dari Michael "Monty" Widenius, David Axmark, dan Alan Larson dimulai tahun 1995. Mereka bertiga kemudian mendirikan perusahaan bernama MySQL AB di Swedia. Tujuan awal ditulisnya program MySQL adalah untuk mengembangkan aplikasi web yang digunakan salah satu klien MySQL AB. Situs internet MySQL AB dapat dikunjungi di http://www.mysql.com.

2.13.2 Keistimewaan MySQL

Sebagai *database server* yang memiliki konsep *database modern*, MySQL memiliki banyak keistimewaan. Berikut ini beberapa keistimewaan yang dimiliki oleh MySQL.

1. *Portability*

   MySQL dapat berjalan stabil pada berbagai system operasi seperti Windows, Linux, FreeBSD, Mac OS X Server, dan lain-lain.

2. *Open Source*

MySQL didistribusikan secara *Open Source*, sehingga dapat digunakan secara bebas.

3. *Multi-user*

   MySQL dapat digunakan oleh beberapa *User* dalam waktu yang bersamaan tanpa mengalami masalah. Hal ini memungkinkan sebuah *database server* MySQL dapat diakses *client* secara bersamaan.

4. *Performance Tuning*

   MySQL memiliki kecepatan yang tinggi dalam menangani *query*, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.

5. *Column Types*

   MySQL memiliki tipe kolom yang sangat kompleks, seperti *signed/unsigned integer, float, double, char, varchar, text, blob, date, time, timestamp, year, set* dan *enum.*

6. *Command* dan *Function*

   MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah SELECT dan WHERE dalam *query.*

7. *Security*

   MySQL memiliki lapisan lapisan sekuritas seperti *level subnetmask,* nama *host*, dan izin akses *User* dengan *system* perizinan yang mendetail serta *password* terenkripsi.

8. *Scalability* dan Limits

   MySQL mampu menangani *database* dengan skala besar, dengan jumlah records lebih dari 50 juta dan 60 juta table serta 5 miliar baris.

Selain itu, batas indeks yang dapat ditampung mencapai 32 indeks pada setiap tabelnya.

9. *Connectivity*

   MySQL dapat melakukan koneksi dengan *client* menggunakan *protocol* TCP/IP, Unix Socet (UNIX), atau Named Pipes (NT).

10. *Localisation*

    Deteksi pesan kesalahan pada *client* dengan menggunakan lebih dari 20 bahasa

11. *Interface*

    Interface terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*)

12. *Client* dan *Tools*

    Dilengkapi dengan berbagai *Tool* yang dapat digunakan untuk administrasi *database,* dan pada setiap *tool* yang ada disertakan petunjuk *online*.

13. Struktur Tabel

Struktur tabel yang lebih fleksibel dalam menangani ALTER TABLE, dibanding database lainnya semacam PostgreSQL ataupun Oracle.

2.14 *Personal Home Page Tool* (PHP)

PHP adalah salah satu bahasa *Server-side* yang didesain khusus untuk aplikasi web. PHP dapat disisipkan diantara bahasa HTML dan karena bahasa *Server side,* maka bahasa PHP akan dieksekusi di server, sehingga yang

dikirimkan ke browser adalah “hasil jadi” dalam bentuk HTML, dan kode PHP tidak akan terlihat.

2.14.1 Sejarah PHP

PHP merupakan teknologi yang diperkenalkan pada tahun 1994 oleh Rasmus Lerdorf berupa versi awal yang tidak dipublikasikan, digunakan pada situs pribadinya untuk mencatat siapa saja yang mengakses daftar riwayat hidup onlinenya. Versi pertama digunakan oleh pihak lain pada awal tahun 1995 dan dikenal dengan nama Personal Home Page Tools. di dalamnya terkandung sebuah parser enggine (mesin pengurai) yang sangat disederhanakan, yang hanya mampu mengolah macro khusus dan beberapa utilitas yang sering digunakan pada pembuatan homepage, serta buku tamu, pencacah dan hal semacamnya, parser tersebut ditulis ulang pada tahun 1995 dan dinamakan PHP/FI version 2. FI (form Interpreter) sendiri berasal dari kode lain yang juga ditulis oleh rasmus, yang menterjemahkan HTML dari data. ia menggabungkan Personal; Homepage tool dengan form interpreter dan menambahkan dukungan terhadap server database yang menggunakan format mysql, sehinnbgga lahirlah PHP/FI.PHP/FI tumbuh dengan pesat hingga sekarang.

2.14.2 Keistimewaan PHP

Sebagai bahasa skrip yang bersifat server side PHP memiliki beberapa keunggulan antara lain:

1. Tidak diperlukan kompabilitas browser atau harus menggunakan browser tertentu karena serverlah yang akan mengerjakan skrip PHP tersebut
2. Freeware artinya dapat didistribusikan dengan bebas

Memiliki kemampuan koneksi keberbagai macam database seperti: MySQL, PostgreSQL, Oracle, dBase, Sybase dan banyak lagi.

2.15 Wampserver

Dalam dunia teknologi informasi, kata *server* sudah tidak asing lagi karena untuk menjalankan sebuah sistem informasi atau aplikasi berbasiskan internet harus memiliki paling sedikit satu *server*. Baik itu *server online* atau pun yang *offline*. Agar aplikasi bisa berjalan dengan kinerja yang maksimum.

Salah satu *server offline* yang ada sekarang ini adalah WAMP yang merupakan kependekan dari Windows, Apache, MySQL, dan PHP. WAMP dibuat hanya untuk bekerja pada sistem operasi *windows*.

Apache adalah sebuah *web server* yang membuat akses pengguna yang menggunakan web browser seperti internet explorer atau firefox untuk connet ke komputer dan melihat informasi yang terdapt pada komputer melalui halaman web.

WAMP server digunakan karena *software* lainnya yang membangun aplikasi merupakan software yang mendukung WAMP server.

2.16 Eclipse

Eclipse adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (*platform-independent*). Berikut ini adalah sifat dari Eclipse:

1. Multi-platform: Target sistem operasi Eclipse adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X.
2. Mulit-language: Eclipse dikembangkan dengan bahasa pemrograman Java, akan tetapi Eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti C/C++, Cobol, Python, Perl, PHP, dan lain sebagainya.
3. Multi-role: Selain sebagai IDE untuk pengembangan aplikasi, Eclipse pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, test perangkat lunak, pengembangan web, dan lain sebagainya.

Eclipse pada saat ini merupakan salah satu IDE favorit dikarenakan gratis dan *open source*, yang berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari Eclipse yang membuatnya populer adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan komponen yang dinamakan *plug-in*.

2.16.1 Sejarah Eclipse

Eclipse awalnya dikembangkan oleh IBM untuk menggantikan perangkat lunak IBM Visual Age for Java 4.0. Produk ini diluncurkan oleh IBM pada

tanggal 5 November 2001, yang menginvestasikan sebanyak US$ 40 juta[1] untuk pengembangannya. Semenjak itu konsursium Eclipse Foundation mengambil alih untuk pengembangan Eclipse lebih lanjut dan pengaturan organisasinya.

2.16.2 Arsitektur Eclipse

Sejak versi 3.0, Eclipse pada dasarnya merupakan sebuah *kernel*, yang mengangkat *plug-in*. Apa yang dapat digunakan di dalam Eclipse sebenarnya adalah fungsi dari *plug-in* yang sudah diinstal. Ini merupakan basis dari Eclipse yang dinamakan *Rich Client Platform* (RCP). Berikut ini adalah komponen yang membentuk RCP:

1. *Core platform*
2. OSGi
3. SWT (*Standard Widget Toolkit*)
4. JFace
5. *Eclipse Workbench*

Secara standar Eclipse selalu dilengkapi dengan JDT (*Java Development Tools*), *plug-in* yang membuat Eclipse kompatibel untuk mengembangkan program Java, dan PDE (*Plug-in Development Environment*) untuk mengembangkan *plug-in* baru. Eclipse beserta *plug-in*-nya diimplementasikan dalam bahasa pemrograman Java.

Konsep Eclipse adalah IDE yang terbuka (*open*), mudah diperluas (*extensible*) untuk apa saja, dan tidak untuk sesuatu yang spesifik. Jadi, Eclipse tidak saja untuk mengembangkan program Java, akan tetapi dapat digunakan untuk berbagai macam keperluan, cukup dengan menginstal *plug-in* yang dibutuhkan. Apabila

ingin mengembangkan program C/C++ terdapat *plug-in* CDT (*C/C++ Development Tools*). Selain itu, pengembangan secara visual bukan hal yang tidak mungkin oleh Eclipse, *plug-in* UML2 tersedia untuk membuat diagram UML. Dengan menggunakan PDE setiap orang bisa membuat *plug-in* sesuai dengan keinginannya.

2.16.3 Perkembangan Eclipse

Sejak tahun 2006, Eclipse Foundation mengkoordinasikan peluncuran Eclipse secara rutin dan simultan yang dikenal dengan nama *Simultaneous Release*. Setiap versi peluncuran terdiri dari Eclipse Platform dan juga sejumlah proyek yang terlibat dalam proyek Eclipse. Tujuan dari sistem ini adalah untuk menyediakan distribusi Eclipse dengan fitur-fitur dan versi yang terstandarisasi. Hal ini juga dimaksudkan untuk mempermudah deployment dan maintenance untuk sistem enterprise. Adapun versi eclipse yang telah diluncurkan yaitu :

1. Eclipse 3.0
2. Eclipse 3.1
3. Callisto
4. Europa
5. Ganymede
6. Galileo
7. Helios

2.17 *Software Development Kit* (*SDK*)

Software Development Kit (SDK) adalah suatu kit atau library dari bahasa pemrograman untuk pengembangan atau pembangunan suatu perangkat lunak dan biasanya SDK terdiri dari kumpulan tools yang dibutuhkan. Misalnya bahasa pemrograman java, mempunyai SDK yang berisi suatu *library* yang dapat digunakan untuk membuat suatu aplikasi berbasis java.

2.18 *Java Development Kit* (*JDK*)

Java Development Kit (JDK) adalah sekumpulan perangkat lunak yang dapat kamu gunakan untuk mengembangkan perangkat lunak yang berbasis Java, Sedangkan JRE adalah sebuah implementasi dari Java Virtual Machine yang benar-benar digunakan untuk menjalankan program java.Biasanya, setiap JDK berisi satu atau lebih JRE dan berbagai alat pengembangan lain seperti sumber kompiler java, bundling, debuggers, development libraries dan lain sebagainya. Perbedaan JDK dengan SDK *(Software Development Kit)* yaitu JDK adalah sebuah SDK tetapi sebuah SDK tidak harus menjadi sebuah JDK.

2.19 ADT

Android Development Tools (*ADT*) adalah *plugin* untuk Eclipse *Intergrated Development Environment* (*IDE*) yang dirancang untuk memberikan lingkungan yang terpadu di mana untuk membangun aplikasi Android.

ADT memperluas kemampuan Eclipse untuk membiarkan para *developer* lebih cepat dalam membuat proyek baru Android, membuat aplikasi UI, menambahkan komponen berdasarkan Android *Framework* API, *debug* aplikasi

dalam pengunaan Android SDK, dan membuat file APK untuk mendistribusikan aplikasi.

Mengembangkan aplikasi di Eclipse dengan ADT sangat dianjurkan dan merupakan cara tercepat untuk memulai membuat aplikasi android, karena banyak kemudahan-kemudahan sebagai tools yang terintegrasi seperti, custom XML editor, dan *debug panel ouput*. Selain itu ADT memberikan dorongan luar biasa dalam mengembangkan aplikasi Android.