BAB II

LANDASAN TEORI

2.1 Definisi Aplikasi *Mobile*

Sumber Blogspot (13 : 01 : 2012) Aplikasi adalah program yang digunakan untuk melakukan sesuatu pada sistem komputer. *Mobile* dapat diartikan sebagai perpindahan yang mudah dari suatu tempat ke tempat yang lain, misalnya telepon *mobile* berarti bahwa terminal telepon yan dapat berpindah dengan mudah dari satu tempat ke tempat lain tanpa terjadi pemutusan atau terputusnya komunikasi.

Sistem aplikasi *mobile* merupakan aplikasi yang dapat digunakan walaupun pengguna berpidah dengan mudah dari satu tempat ke tempat lain tanpa terjadi pemutusan atau terputusnya komunikasi. Aplikasi ini dapat diakses melalui perangkat nirkabel seperti *pager,* telepon seluler dan PDA.

2.2 Definisi Transportasi

Sumber Wikipedia (13 : 01 : 2012) Transportasi adalah pemindahan manusia atau barang dari satu tempat ke tempat lainnya dengan menggunakan sebuah kendaraan yang digerakkan oleh manusia atau mesin. Transportasi digunakan untuk memudahkan manusia dalam melakukan aktivitas sehari-hari. Di negara maju, mereka biasanya menggunakan kereta bawah tanah (*subway*) dan taksi. Penduduk disana jarang yang mempunyai kendaraan pribadi karena mereka sebagian besar menggunakan angkutan umum sebagai transportasi mereka.

Transportasi sendiri dibagi 3 yaitu, transportasi darat, laut, dan udara. Transportasi udara merupakan transportasi yang membutuhkan banyak uang untuk memakainya. Selain karena memiliki teknologi yang lebih canggih, transportasi udara merupakan alat transportasi tercepat dibandingkan dengan alat transportasi lainnya.

2.3 Android

Menurut Nazruddin Safaat H (2011 : 1) Android adalah sistem operasi untuk perangkat mobile berbasis Linux yang mencakup sistem operasi, *middleware* dan aplikasi. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak. Android umum digunakan di *smartphone* dan juga tablet PC. Fungsinya sama seperti sistem operasi Symbian di Nokia, iOS di Apple dan BlackBerry OS Android tidak terikat ke satu merek *Handphone* saja, beberapa vendor terkenal yang sudah memakai Android antara lain Samsung, Sony Ericsson, HTC, Nexus, Motorolla, dan lain-lain.

Keunggulan utama Android adalah gratis dan *open source,* yang membuat *smartphone* Android dijual lebih murah dibandingkan dengan Blackberry atau iPhone meski fitur (*hardware*) yang ditawarkan Android lebih baik.

Beberapa fitur utama dari Android antara lain *WiFi hotspot*, *Multi-touch*, *Multitasking, GPS, accelerometers, support java*, mendukung banyak jaringan *(GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE & WiMAX)* serta juga kemampuan dasar *handphone* pada umumnya.

2.3.1 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.

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.

2.3.2 Arsitektur Android

Secara garis besar, arsitektur android terdiri atas *Applications* dan *Widgets*, *Applications Frameworks, Libraries, Android Run Time,* dan *Linux Kernel.*

a. *Applications dan Widgets* merupakan layer (lapis) dimana kita berhubungan dengan aplikasi kita.
b. *Applications Framework* merupakan *Open Development Platform* yang ditawarkan Android untuk dapat ditawarkan Android untuk dapat dikembangkan guna membangun aplikasi. Pengembang memiliki akses penuh menuju *API Framework* seperti yang dilakukan oleh aplikasi kategori inti. Komponen-komponen yang termasuk dalam *Applications Framework* adalah sebagai berikut :
    1. *Views* yang kaya akan dan extensible yang dapat digunakan untuk membangun aplikasi, termasuk list, grids, kotak teks, tombol dan bahkan sebuah *embeddable web*.
    2. *Content Provider* yang memungkinkan aplikasi untuk mengakses data (*seperti dari daftar kontak telepon*) atau membagi data tersebut.
    3. *Resource Manager* yang menyediakan akses ke kode non sumber lokal seperti string, gambar, dan tata letak file.

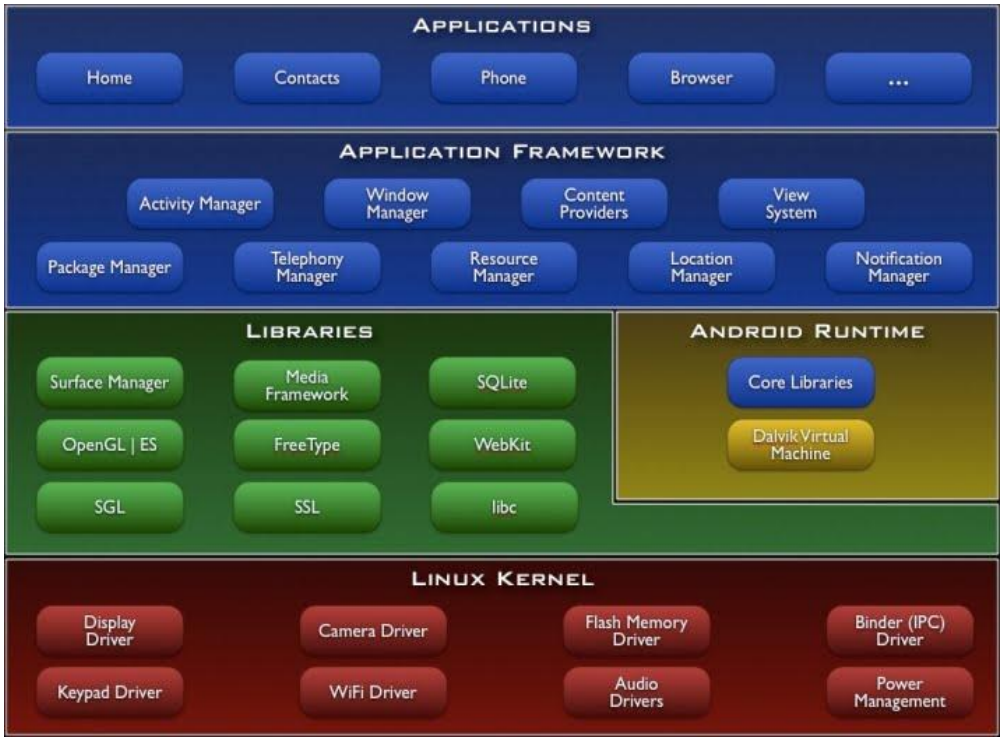
4. *Notification Manager* yang memungkinan semua aplikasi menampilkan alert yang bias dikustomisasi di dalam status bar.
5. *Activity Manager* yang mengelola siklus hidup aplikasi dan menyediakan navigasi umum backstack.

c. *Libraries* merupakan layer dimana fitur-fitur Android berada. Android menyertakan libraries C / C+ + yang digunakan oleh berbagai komponen dari sistem android. Kemampuan ini disediakan kepada *Developer* aplikasi melalui *Framework* aplikasi android.

1. Sistem *C Library* merupakan variasi dari implementasi BSD-berasal pelaksanaan sistem standar *C Library (libe),* sesuai untuk perangkat *embedded* berbasis Linux.
2. Media *Library* merupakan library yang mendukung pemutaran rekaman dan popular banyak format audio dan video, serta file gambar.
3. *Graphic Library* termasuk didalamnya SGL dan OpenGL untuk tampilan 2D dan 3D.
4. *Surface Manager* merupakan *library* untnuk mengelola akses ke subsistem layar.
5. *LibWebcore* merupakan mesin web modern yang *powerfull* yang baik untuk *browser* android.
6. *FreeType* mendukung bitmap dan *vector font rendering*.
7. *SQLite* merupakan mesin *database* yang kuat dan ringan, dan penghubung tersedia untuk semua aplikasi.

d. *Android Run Time* merupakan Layer yang membuat aplikasi Android dapat dijalankan, dimana dalam prosesnya menggunakan implementasi Linux. Android terdiri dari satu set *core libraries* yang menyediakan sebagian besar fungsi yang sama dengan yang terdapat pada *core* yang *libraries* bahasa pemrograman Java.

e. *Linux Kernel* Merupakan layer inti dari sistem operasi Android berada. Berisi file-file sistem yang mengatur sistem *processing, memory, resources, drivers,* dan sistem-sistem operasi android lainnya. Android bukan Linux, akan tetapi dibangun diatas Linux Kernel yaitu versi 2.6 sehingga kehandalannya dapat dipercaya. Untuk inti sistem service Linux yang digunakan seperti keamanan, manajemen memori, proses manajemen, network dan driver model. Linux menyediakan Driver layar, Kamera, Keypad, WiFi, Flash Memory, Audio dan IPC (*Interprocess Communication*) untuk mengatur keamanan. Kernel juga bertindak sebagai lapisan abstrak antara *hardware* dan *software stack*-nya. Untuk lebih jelasnya lihat gambar arsitektur Android berikut.



Gambar 2.1 Arsitektur Android

(Diadopsi : http://studyfuture.blogspot.com/2011/03/arsitektur-android.htm)

2.3.3 Komponen Aplikasi Android

Salah satu fitur utama dari android adalah satu aplikasi dapat menggunakan elemen dari aplikasi lain (yang disediakan berdasarkan izin aplikasi itu). Misalnya, jika kita memerlukan aplikasi untuk menampilkan *scrolling list* gambar dan aplikasi lain telah memiliki *scroller* yang sesuai dan membuatnya tersedia untuk aplikasi lain, kita dapat memanggil *scroller* tersebut untuk melakukan suatu pekerjaan, daripada kita membuatnya lagi. Aplikasi tidak memasukan kode dari aplikasi yang lain atau meliriknya. Sebaliknya ia hanya memulai bagian dari aplikasi lainnya saat membutuhkan.

Agar aplikasi dapat bekerja, sistem harus dapat menjalankan sebuah proses aplikasi ketika salah satu bagian itu diperlukan, dan menginisiasi objek Java pada bagian tersebut. Krena itu, tidak seperti pada kebanyakan aplikasi lainnya, pada sistem aplikasi android tidak ada satu *entry point* bagi semua yang ada dalam aplikasi (*misalnya : tidak ada fungsi main()*). Sebaliknya mereka mempunyai komponen penting dalam sistem yang dapat diinisiasi dan dijalankan sesuai dengan kebutuhan.

Ada empat komponen pada aplikasi komponen untuk menyajikan *user interface* (tampilan program) kepada pengguna.

1. *Activities* merupakan komponen utama untuk menyajikan *user interface* (tampilan program) pada pengguna.
2. *Service* merupakan komponen yang tidak memiliki *user interface* (tampilan program), tetapi *service* berjalan secara *backgrounds*.
3. *Broadcast Receiver* merupakan komponen yang berfungsi menerima dan bereaksi untuk menyampaikan notifikasi.
4. *Content Provider* merupakan komponen membuat kumpulan aplikasi data secara spesifik sehingga bisa digunakan oleh aplikasi lain.

2.3.4 Versi Android

Banyak *smartphone* dan PC Tablet menggunakan sistem operasi yang berbeda. Semakin versi tinggi, fiturnya semakin canggih dan banyak. Telepon pertama yang memakai sistem operasi Android adalah HTC Dream yang dirilis pada tanggal 22 Oktober 2008. Beberapa uraian versi Android seperti di bawah ini.

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, kamera, camcorder dan galeri yang dintegrasikan; CDMA / EVDO, 802.1x, VPN, Gestures, dan *Text-to-speech engine*; kemampuan dial kontak; teknologi

text to change speech (tidak tersedia pada semua ponsel; pengadaan resolusi VWGA.

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.

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 3.0/3.1 (*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. Perangkat tablet dengan platform Android 3.0 akan segera hadir di Indonesia. Perangkat tersebut bernama Eee Pad Transformer produksi dari Asus. Rencana masuk pasar Indonesia pada Mei 2011.

8. **Android versi 4.0 (ICS :*Ice Cream Sandwich*)**

   Diumumkan pada tanggal 19 Oktober 2011, membawa fitur *Honeycomb* untuk *smartphone* dan menambahkan fitur baru termasuk membuka kunci dengan pengenalan wajah, jaringan data pemantauan penggunaan dan kontrol, terpadu kontak jaringan sosial, perangkat tambahan fotografi, mencari email secara *offline,* dan berbagi informasi dengan menggunakan NFC.

2.3.5 Fitur Android

Sebagai sebuah Sistem Operasi tentunya terdapat beberapa fitur atau layanan yang disediakan oleh Android. Adapun Fitur yang tersedia di Android diantaranya :

1. Kerangka aplikasi: itu memungkinkan penggunaan dan penghapusan komponen yang tersedia.
2. Dalvik mesin virtual: mesin virtual dioptimalkan untuk perangkat mobile.
3. Grafik: grafik di 2D dan grafis 3D berdasarkan pustaka OpenGL.
4. SQLite: untuk penyimpanan data.
5. Mendukung media: audio, video, dan berbagai format gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF)
6. GSM, Bluetooth, EDGE, 3G, dan WiFi (*hardware dependent*)
7. Kamera, Global Positioning Sistem (GPS), kompas, dan *accelerometer* (tergantung *hardware*), dan lain-lain.

2.4 Google *Maps*

Sumber Wikipedia (13 : 04 : 2012) Google *Maps* merupakan sebuah layanan peta dunia virtual berbasis web yang disediakan oleh Google. Layanan ini gratis dan dapat ditemukan di http://maps.google.com.

Google Maps menawarkan peta yang dapat digeser (*panned*), diperbesar (*zoom in*), diperkecil (*zoom out*), dapat diganti dalam beberapa mode (*ma*p, *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 yangdikenal dengan API *Key*.

2.5 *Global Positioning System (GPS)*

Sumber Wikipedia (22 : 04 : 2012) *Global Positioning System* (GPS) adalah sistem untuk menentukan posisi di permukaan bumi dengan bantuan sinkronisasi sinyal satelit. Sistem ini menggunakan 24 satelit yang mengirimkan sinyal gelombang mikro ke Bumi. Sinyal ini diterima oleh alat penerima di permukaan, dan digunakan untuk menentukan posisi, kecepatan, arah, dan waktu. Sistem yang serupa dengan GPS antara lain GLONASS Rusia, Galileo Uni Eropa, IRNSS India.

Sistem ini dikembangkan oleh Departemen Pertahanan Amerika Serikat, dengan nama lengkapnya adalah **NAVSTAR GPS** (kesalahan umum adalah bahwa NAVSTAR adalah sebuah singkatan, ini adalah salah, NAVSTAR adalah nama yang diberikan oleh John Walsh, seorang penentu kebijakan penting dalam program GPS). Kumpulan satelit ini diurus oleh 50th Space Wing Angkatan Udara Amerika Serikat. Biaya perawatan sistem ini sekitar US$750 juta per tahun termasuk penggantian satelit lama, serta riset dan pengembangan.

GPS Tracker atau sering disebut dengan ***GPS Tracking*** adalah teknologi AVL (*Automated Vehicle Locater*) yang memungkinkan pengguna untuk melacak posisi kendaraan, armada ataupun mobil dalam keadaan *Real-Time*. *GPS Tracking* memanfaatkan kombinasi teknologi GSM dan GPS untuk menentukan koordinat sebuah obyek, lalu menerjemahkannya dalam bentuk peta digital.

2.5.1 Cara Kerja GPS

Sistem ini menggunakan sejumlah satelit yang berada di orbit bumi, yang memancarkan sinyalnya ke bumi dan ditangkap oleh sebuah alat penerima. Ada tiga bagian penting dari sistem ini, yaitu bagian kontrol, bagian angkasa, dan bagian pengguna.

1. Bagian Kontrol

Seperti namanya, bagian ini untuk mengontrol. Setiap satelit dapat berada sedikit diluar orbit, sehingga bagian ini melacak orbit satelit, lokasi, ketinggian, dan kecepatan. Sinyal-sinyal sari 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 di kirimkan kepada alat navigasi kita.

2. Bagian Angkasa

Bagian ini terdiri dari kumpulan satelit-satelit yang berada di orbit bumi, sekitar 12.000 mil di atas permukaan bumi. Kumpulan satelit-satelit ini diatur sedemikian rupa sehingga alat navigasi setiap saat dapat menerima paling sedikit sinyal dari empat buah satelit. Sinyal satelit ini dapat melewati awan, kaca, atau plastik, tetapi tidak dapat melewati gedung atau gunung. Satelit mempunyai jam atom, dan juga akan memancarkan informasi ‘waktu/jam’ ini. Data ini dipancarkan dengan kode ‘pseudo-random’. Masing-masing satelit memiliki kodenya sendiri-sendiri. Nomor kode ini biasanya akan ditampilkan di alat navigasi, maka kita bisa melakukan identifikasi sinyal satelit yang sedang diterima alat tersebut. Data ini berguna bagi alat navigasi untuk mengukur jarak antara alat

navigasi dengan satelit, yang akan digunakan untuk mengukur koordinat lokasi. Kekuatan sinyal satelit juga akan membantu alat dalam penghitungan. Kekuatan sinyal ini lebih dipengaruhi oleh lokasi satelit, sebuah alat akan menerima sinyal lebih kuat dari satelit yang berada tepat di atasnya (bayangkan lokasi satelit seperti posisi matahari ketika jam 12 siang) dibandingkan dengan satelit yang berada di garis cakrawala (bayangkan lokasi satelit seperti posisi matahari terbenam/terbit).

Ada dua jenis gelombang yang saat ini dipakai untuk alat navigasi berbasis satelit pada umumnya, yang pertama lebih dikenal dengan sebutan L1 pada 1575.42 MHz. Sinyal L1 ini yang akan diterima oleh alat navigasi. Satelit juga mengeluarkan gelombang L2 pada frekuensi 1227.6 Mhz. Gelombang L2 ini digunakan untuk tujuan militer dan bukan untuk umum.

3. Bagian Pengguna

Bagian ini terdiri dari alat navigasi yang digunakan. Satelit akan memancarkan data almanak dan ephemeris yang akan diterima oleh alat navigasi secara teratur. Data almanak berisikan perkiraan lokasi (*approximate location*) satelit yang dipancarkan terus menerus oleh satelit. Data ephemeris dipancarkan oleh satelit, dan valid untuk sekitar 4-6 jam. Untuk menunjukkan koordinat sebuah titik (dua dimensi), alat navigasi memerlukan paling sedikit sinyal dari 3 buah satelit. Untuk menunjukkan data ketinggian sebuah titik (tiga dimensi), diperlukan tambahan sinyal dari 1 buah satelit lagi.

Dari sinyal-sinyal yang dipancarkan oleh kumpulan satelit tersebut, alat navigasi akan melakukan perhitungan-perhitungan, dan hasil akhirnya adalah

koordinat posisi alat tersebut. Makin banyak jumlah sinyal satelit yang diterima oleh sebuah alat, akan membuat alat tersebut menghitung koordinat posisinya dengan lebih tepat.

Karena alat navigasi ini bergantung penuh pada satelit, maka sinyal satelit menjadi sangat penting. Alat navigasi berbasis satelit ini tidak dapat bekerja maksimal ketika ada gangguan pada sinyal satelit. Ada banyak hal yang dapat mengurangi kekuatan sinyal satelit:

a. Kondisi geografis, seperti yang diterangkan di atas. Selama kita masih dapat melihat langit yang cukup luas, alat ini masih dapat berfungsi.
b. Hutan. Makin lebat hutannya, maka makin berkurang sinyal yang dapat diterima.
c. Air. Jangan berharap dapat menggunakan alat ini ketika menyelam.
d. Kaca film mobil, terutama yang mengandung metal.
e. Alat-alat elektronik yang dapat mengeluarkan gelombang elektromagnetik.
f. Gedung-gedung. Tidak hanya ketika di dalam gedung, berada di antara 2 buah gedung tinggi juga akan menyebabkan efek seperti berada di dalam lembah.
g. Sinyal yang memantul, misal bila berada di antara gedung-gedung tinggi, dapat mengacaukan perhitungan alat navigasi sehingga alat navigasi dapat menunjukkan posisi yang salah atau tidak akurat.

2.6 Konsep Perancangan Berorientasi Objek

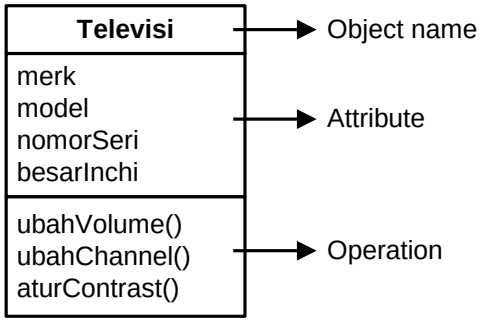
Untuk memahami titik pandang dan maksud dari ‘berorientasi objek’, kita dapat mempelajarinya dari alam secara luas. Obyek ada disekeliling kita, baik yang konkrit atau konseptual. Dalam sudut pandang Eksekutif perusahaan: Karyawan, Absensi, Gaji, Profit dapat disebut sebagai Objek. Seorang Arsitek melihat Gedung, Biaya dan tenaga kerja sebagai objek. Konsep-konsep dasar dalam memahami Objek dapat dilihat pada subjudul berikut:

2.6.1 *Object* / Objek

Menurut Yasmi Afrizal dan Wahyuni (2012 : 162) Objek adalah orang, tempat, benda, kejadian atau konsep-konsep yang ada di dunia nyata dan penting bagi suatu aplikasi. Sebuah objek adalah Entitas yang memiliki Identitas, *State* (keadaan sesaat) dan *Behavior* (perilaku).

State sebuah objek adalah kondisi objek tersebut yang dinyatakan dalam **Atribut** atau *property*. *Behavior* sebuah objek mendefinisikan bagaimana sebuah objek bertindak/bereaksi yang dinyatakan dalam ***Operation***. Satu objek dapat diturunkan menjadi objek dalam bentuk lain, kemudian saling mengkait menyusun sesuatu yang lebih rumit.

Langkah pertama yang harus dilakukan dalam pengembangan PL berorientasi objek adalah melakukan Abstraksi yaitu: kegiatan atau suatu usaha untuk mengenali objek-objek dan mengelompokkannya kedalam suatu kelas. Misalkan objek Hewan: Unggas, Reptil, maka Unggas dan Reptil adalah kelas-kelas dalam objek Hewan. Tata cara atau notasi pembuatan entitas objek digambarkan sebagai berikut:



Gambar 2.2 Notasi Pembuatan Entitas Objek
Diadopsi : Yasmi Afrizal dan Wahyuni (2011 : 163)

2.6.2 *Class* / Kelas

Menurut Yasmi Afrizal dan Wahyuni (2012 : 163) *Class* adalah kumpulan atau himpunan objek-objek yang sejenis, memiliki kesamaan atribut/*property*, perilaku, serta relasi dengan objek lain yang mirip. Notasi kelas digambarkan dengan kotak, dengan nama kelas didalamnya ditulis menggunakan huruf besar di awal kata. Bila sebuah kelas memiliki 2 suku kata atau lebih, maka penulisannya disatukan tanpa spasi dengan huruf awal tiap suku menggunakan huruf besar.Contohnya adalah Barang Elektronik dapat dikatakan sebagai sebuah Kelas apabila memiliki kesamaan dengan objek yang ada padanya misalnya Mesin Cuci, Televisi, Radio, Kulkas adalah objek-objek yang dapat dikelompokkan kedalam satu kelas yaitu Barang Elektronik rumah tangga.



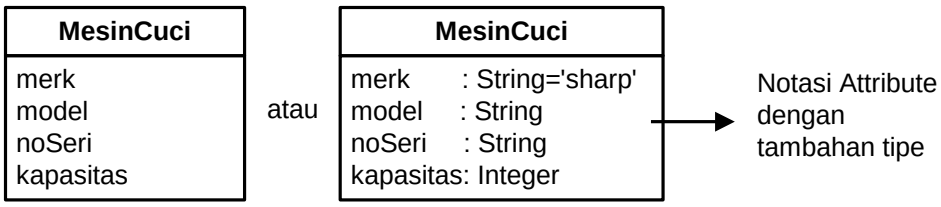
Gambar 2.3 Notasi *Class*
Diadopsi : Yasmi Afrizal dan Wahyuni (2011 : 163)

2.6.3 *Attribute* / Atribut

Menurut Yasmi Afrizal dan Wahyuni (2012 : 163) *Attribute* adalah data yang dimiliki suatu objek atau *property* dari sebuah *Class* yang menggambarkan

batas nilai yang mungkin ada pada objek dari kelas. Sebuah bisa memiliki nol atau lebih atribut. Notasi atribut digambarkan dengan kotak dibawah kotak *class*, dengan nama atribut didalamnya ditulis menggunakan huruf kecil. Jika sebuah atribut memiliki 2 atau lebih suku kata, maka semua suku kata ditulis disatukan tanpa spasi, awal suku kata pertama dengan huruf kecil dan awal suku kata berikutnya dengan huruf besar.

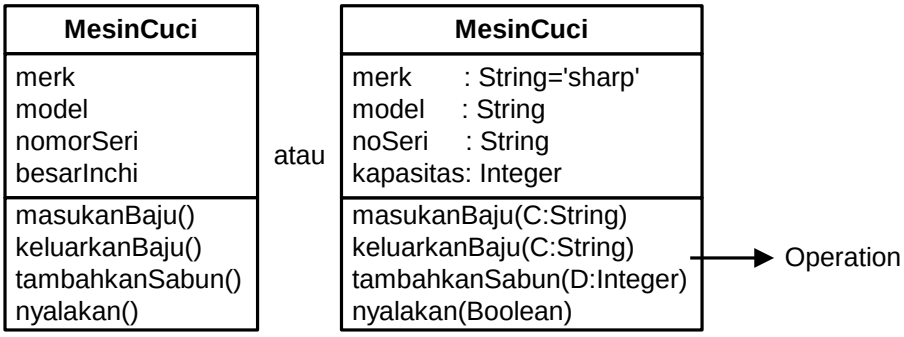
Notasi atribut dapat ditambahkan informasi dengan tipe-tipe atribut tersebut. Penulisan tipe pada atribut dipisahkan dengan tanda titik dua (:), tipe yang ditambahkan berupa String, Floating-Point number, Integer dan Boolean.



Gambar 2.4 Notasi *Attribute*
Diadopsi : Yasmi Afrizal dan Wahyuni (2011 : 164)

2.6.4 *Operation* / Operasi

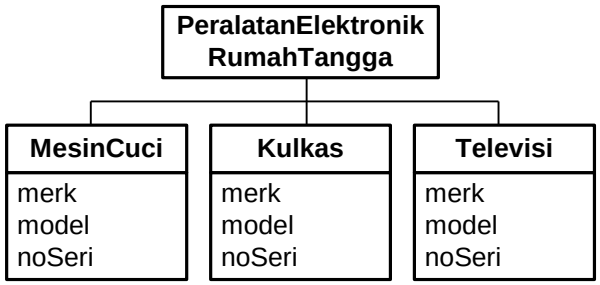
Menurut Yasmi Afrizal dan Wahyuni (2012 : 164) *Operation* adalah sesuatu yang bisa dilakukan oleh sebuah *class*. Notasi penulisannya sama dengan atribut. Bagian *operation* ini juga bisa diberikan tambahan informasi, yaitu dengan menambahkan parameter yang akan dilakukan operation dalam tanda kurung. Contoh parameternya adalah *function*.



Gambar 2.5 Notasi *Operation*
Diadopsi : Yasmi Afrizal dan Wahyuni (2011 : 164)

2.6.5 *Inheritance* / Pewarisan

Menurut Yasmi Afrizal dan Wahyuni (2012 : 164) *Inheritance* atau pewarisan memungkinkan dibuat *class* yang menyerupai *class* lain yang telah ada sebelumnya, tetapi masih memiliki beberapa sifat induknya. Misalkan dari sebuah mobil biasa, anda dapat membuat mobil balap serta mobil angkutan umum. Prosesnya adalah dengan mengubah sifat dari mobil biasa tersebut.



Gambar 2.6 Penggambaran Pewarisan
Diadopsi : Yasmi Afrizal dan Wahyuni (2011 : 165)

2.6.6 *Polymorphisme* / Kebanyakrupaan

Menurut Yasmi Afrizal dan Wahyuni (2012 : 165) *Polymorphisme* adalah objek yang memiliki fungsi sama dengan objek dasar tetapi memiliki satu atau lebih sifat berbeda atau dengan kata lain *Polymorphisme* adalah pemisahan secara jelas diantara subsistem yang berbeda. Sebagai contoh misalkan sebuah kelas

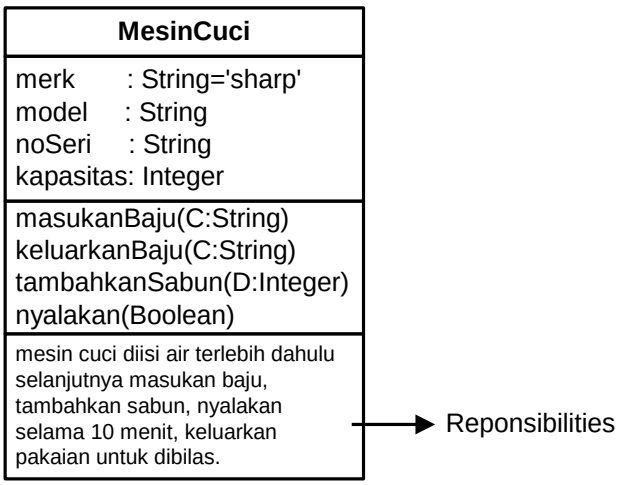
memiliki operasi ‘OPEN’, operasi open ini bisa dipakai untuk membuka pintu, membuka buku, membuka baju dan lainnya. Meskipun ‘OPEN’ memiliki tujuan yang sama, tapi apa yang dilakukannya berbeda.

2.6.7 *Encapsulation* / Pembungkusan

Menurut Yasmi Afrizal dan Wahyuni (2012 : 165) *Encapsulation* sering disebut dengan penyembunyian informasi (*Hidding*), suatu konsep berdasarkan fakta di dunia nyata yang menyatakan bahwa segala sesuatu tidak perlu diperlihatkan. Misalnya kita tidak perlu tahu apa yang dilakukan sistem ketika kita menekan remote untuk menghidupkan televisi.

2.6.8 *Responsibilities* / Tanggung Jawab

Menurut Yasmi Afrizal dan Wahyuni (2012 : 165) *Responsibilities* adalah model tambahan yang digambarkan pada bagian bawah suatu kelas setelah bagian operasi digunakan untuk menjelaskan pernyataan-pernyataan mengenai apa-apa yang bisa dilakukan oleh kelas tersebut.



Gambar 2.7 Pembuatan *Responsibilities*
Diadopsi : Yasmi Afrizal dan Wahyuni (2011 : 166)

2.6.9 *Unified Modeling Language (UML)*

Menurut Yasmi Afrizal dan Wahyuni (2012 : 181) UML (*Unified Modeling Language*) merupakan pengganti dari metode analisis berorientasi objek dan desain berorientasi objek (OOA & OOD) yang dimunculkan sekitar akhir tahun 80-an dan awal tahun 90-an.

UML merupakan gabungan dari metode Grady Booch (Booch Method), James Rumbaugh (OMT) dan Ivar Jacobson (OOSE). Tetapi UML ini akan mencakup lebih luas daripada OOA&D. Pada pertengahan pengembangan UML dilakukan standarisasi proses dengan OMG (*Object Management Group*) dengan harapan UML akan menjadi bahasa standar pemodelan pada masa yang akan datang.

UML disebut sebagai bahasa pemodelan bukan metode. Kebanyakan metode terdiri paling sedikit prinsip, bahasa pemodelan dan proses. Bahasa pemodelan (sebagian besar grafik) merupakan notasi dari metode yang digunakan untuk mendesain secara cepat.

Bahasa pemodelan merupakan bagian terpenting dari metode. Ini merupakan bagian kunci tertentu untuk komunikasi. Jika anda ingin berdiskusi tentang desain dengan seseorang, maka Anda hanya membutuhkan bahasa pemodelan bukan proses yang digunakan untuk mendapatkan desain.

UML merupakan bahasa standar untuk penulisan *Blueprint Software* yang digunakan untuk Visualisasi (*Visualize*), Spesifikasi (*Specify*), Pembentukan (*Construct*) dan Pendokumentasian (*Documentation*) alat-alat dari sistem perangkat lunak.

Tujuan penggunaan UML adalah sebagai berikut:

1. Memodelkan suatu sistem (bukan hanya perangkat lunak) yang menggunakan konsep berorientasi objek
2. Menciptakan suatu bahasa pemodelan yang dapat digunakan baik oleh manusia maupun mesin.

Keunggulan menggunakan UML dibandingkan menggunakan metodologi terstruktur:

1. *Uniformity*

   Pengembang cukup menggunakan 1 metodologi dari tahap analsis hingga perancangan. Memungkinkan merancang komponen antarmuka secara terintegrasi bersama perancangan Perangkat Lunak dan perancangan struktur data
2. *Understandability*

   Kode yang dihasilkan dapat diorganisasi kedalam kelas-kelas yangberhubungan dengan masalah sesungguhnya sehingga lebih mudah untuk dipahami.
3. *Stability*

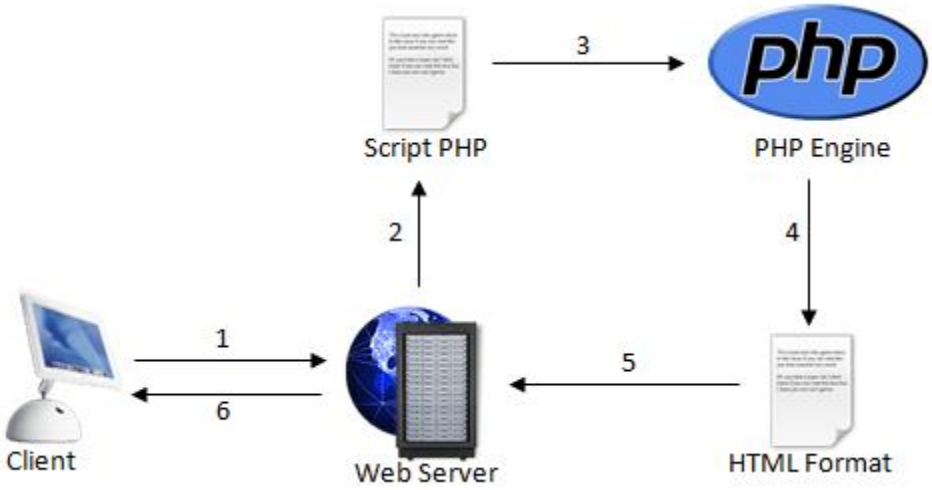
   Kode program yang dihasilkan relatif stabil sepanjang waktu, karena mendekati permasalahan yang sesungguhnya.
4. *Reusability*

Dengan metodologi berorientasi objek, dimungkinkan penggunaan ulang kode, sehingga pada akhirnya akan sangat mempercepat waktu pengembangan perangkat lunak (atau sistem informasi)

2.7 PHP

Menurut Bunafit Nugroho (2008 : 113) PHP:*Hypertext Preprocessor* merupakan bahasa yang hanya dapat berjalan pada server dan hasilnya dapat ditampilkan pada *Client*.

PHP adalah produk open source yang dapat digunakan secara gratis tanpa hasrus membayar untuk menggunakannya. Intepreter PHP dalam mengeksekusi kode PHP pada sisi server (disebut *server-side*), sedangkan tanpa adanya *interpreter* PHP, maka semua skrip dan aplikasi PHP yang dibuat tidak dapat dijalankan. Proses eksekusi kode PHP yang dilakukan oleh *Apache Web Server* dan *interpreter* secara diagram dapat digambarkan sebagai berikut.



Gambar 2.8 Struktur Pembacaan *Web Server*

Diadopsi : Bunafit Nugroho (2008 : 114)

PHP merupakan bahasa standar yang digunakan dalam dunia *website*, PHP adalah bahasa program yang berbentuk skrip yang diletakkan didalam server web. Jika dilihat dari sejarah mulanya PHP diciptakan dari ide Rasmus Lerdof untuk kebutuhan pribadinya, skrip tersebut sebenarnya dimaksudakan untuk digunakan sebagai keperluan untuk membuat *website* pribadi, akan tetapi kemudian dikembangkan lagi sehingga menjadi sebuah bahasa yang disebut "*Personal Home Page*", inilah awal munculnya PHP sampai saat ini.

2.8 Basis Data

Sumber Wikipedia (13 : 01 : 2012) Basis data adalah kumpulan informasi yang disimpan di dalam komputer secara sistematik sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Perangkat lunak yang digunakan untuk mengelola dan memanggil kueri (*query*) basis data disebut sistem manajemen basis data (*database management sistem*, DBMS). Sistem basis data dipelajari dalam ilmu informasi.

Konsep dasar dari basis data adalah kumpulan dari catatan-catatan, atau potongan dari pengetahuan. Sebuah basis data memiliki penjelasan terstruktur dari jenis fakta yang tersimpan di dalamnya: penjelasan ini disebut skema. Skema menggambarkan obyek yang diwakili suatu basis data, dan hubungan di antara obyek tersebut. Ada banyak cara untuk mengorganisasi skema, atau memodelkan struktur basis data: ini dikenal sebagai model basis data atau model data. Model yang umum digunakan sekarang adalah model relasional, yang menurut istilah layman mewakili semua informasi dalam bentuk tabel-tabel yang saling

berhubungan dimana setiap tabel terdiri dari baris dan kolom (definisi yang sebenarnya menggunakan terminologi matematika). Dalam model ini, hubungan antar tabel diwakili dengan menggunakan nilai yang sama antar tabel. Model yang lain seperti model hierarkis dan model jaringan menggunakan cara yang lebih eksplisit untuk mewakili hubungan antar tabel.

Istilah *basis data* mengacu pada koleksi dari data-data yang saling berhubungan, dan perangkat lunaknya seharusnya mengacu sebagai *sistem manajemen basis data* (*database management sistem/DBMS*). Jika konteksnya sudah jelas, banyak administrator dan *programer* menggunakan istilah basis data untuk kedua arti tersebut.

2.9 MySQL

Menurut Bunafit Nugroho (2008 : 91) MySql (*My Structurd* Query *Language*) atau yang biasa dibaca **mai-se-kuel** adalah sebuah program pembuat dan pengelola *database* atau yang sering disebut dengan DBMS(*DataBase Management Sistem*), sifat dari DBMS ini adalah *Open Source,* yang bisa didapatkan secara gratis pada alamat http://www.mysql.com. Saat kita mendengar open source maka seakan kita ingat dengan sistem operasi handal keturunan Unix yaitu Linux.

MySql sebenarnya produk yang berjalan pada platform linux, dengan adanya perkembangan dan banyaknya pengguna, serta lisensi dari *DataBase* ini adalah *Open Source*, maka para pengembang kemudian merilis versi windows.

Selain itu MySql juga merupakan program pengakses *DataBase* yang bersifat jaringan, sehingga dapat digunakan untuk aplikasi *multiuser*(banyak pengguna). Kelebihan lain dari MySql adalah menggunakan bahasa *Query*(permintaan) standar SQL(*Standart Query Language*). SQL adalah suatu bahasa permintaan yang terstruktur, Sql telah distandarkan untuk semua program pengakses *database* seperti Oracle, PosgreSql, sql server dll.

Sebagai sebuah program penghasil *database*, MySql tidak mungkin berjalan sendiri tanpa adanya sebuah aplikasi pengguna atau *interface* yang berguna sebagai prpogram aplikasi pengakses *database* yang dihasilkan. MySql dapat didukung hampir semua program aplikasi baik *open source* seperti php maupun yang tidak *open source* yang ada pada platform windows seperti Visual Basic dan Delphi.

2.10 Perangkat Lunak yang Digunakan

Dalam pembuatan aplikasi *mobile* ini penulis menggunakan *software* pendukung diantaranya :

1. J2SDK

   J2SDK merupakan platform dasar untuk Java

2. Eclipse

   Menurut Ir. Yuniar Supardi (2011 : 11) Eclipse merupakan perangkat lunak untuk bahasa pemrograman Java. Perangkat lunak ini berguna untuk membuat dan meng-*compile source code* bahasa pemrograman Java.

   Berikut ini adalah sifat dari Eclipse:

a. *Multi-platform*: Target sistem operasi Eclipse adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X.
b. *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.
c. *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.

3. Android SDK platform

   Android SDK merupakan alat atau *tool* yang digunakan untuk membuat aplikasi platform Android menggunakan bahasa pemrograman Java.

4. Android *Development Tool*/Plugins Eclipse

   ADT akronim Android Development Tool atau sering disebut juga dengan Plugin Eclipse, yang merupakan penghubung antara IDE Eclipse dengan Android SDK sehingga kita dapat membuat proyek yang berbasis android.

5. *Astah Community 6.1*

   Merupakan *tool* / *model* untuk merancang pengembangan *software* yang berbasis *object oriented*

6. *Adobe Dreamweaver* CS3

   Merupakan editor untuk pemrograman *website*.

7. XAMPP

Menurut Bunafit Nugroho (2008 : 74) XAMPP merupakan paket PHP yang berbasis *open source* yang dikembangkan oleh sebuah komunitas *open source*. Dengan menggunakan XAMPP kita tidak usah lagi bingung melakukan penginstallan program-program yang lain, karena semua kebutuhan sudah disediakan oleh XAMPP. Berikut adalah beberapa paket yang telah disediakan :

1. Apache HTTPD 2.0.54
2. MySql 4.1.12
3. PHP 5.0.3
4. File Zilla FTP Server 0.9 Beta
5. PHP Myadmin 2.6.1-pl3