BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Instansi

2.1.1 Profil Instansi

Era Teknologi informasi saat ini telah mengubah tatanan hidup manusia dalam berkomunikasi dan bersosialisasi. Teknologi informasi memungkinkan adanya komunikasi tanpa adanya batas waktu dan tempat. Pesatnya perkembangan teknologi informasi mendorong manusia untuk mengubah gaya hidupnya, sehingga manusia senantiasa berinovasi menciptakan sesuatu yang baru tidak hanya untuk memenuhi kebutuhan hidupnya saat ini, melainkan menciptakan era baru yang “memaksa” manusia untuk berada pada era dimana teknologi menjadi kekuatan besar dalam "change of life style". Pesatnya perkembangan dunia teknologi informasi ini, bukan hanya dalam hitungan hari melainkan lebih cepat dari itu.

Teknologi Mobile adalah salah satu contoh teknologi yang paling cepat perkembangannya diantara pesatnya perkembangan teknologi Internet. Perkembangan kedua teknologi ini harus tetap diimbangi dengan ilmu dan pengetahuan yang cukup pada sisi pengguna, supaya tujuan positif dari penggunaan teknologi informasi bisa tercapai dan dirasakan oleh semua pihak yang bersangkutan sesuai dengan tujuan awal yaitu memberikan banyak manfaat untuk kebutuhan hidup manusia.

Indonesia saat ini menjadi salah satu negara dengan pengguna teknologi informasi terbesar di dunia. Sebut saja Facebook dan Twitter dalam bidang jejaring sosial, Indonesia disebutkan sebagai peringkat 3 terbesar dalam kategori pengguna dan diprediksikan akan terus meningkat. Selain itu Indonesia juga tercatat sebagai pengguna Blackberry dan pengunduh software aplikasi terbanyak untuk beberapa aplikasi. Dan tentu saja masih banyak prestasi Indonesia dari sisi konsumsi produk teknologi di kancah global.

Apabila kita berbicara mengenai teknologi mobile, maka sudah barang tentu kita bisa membayangkan seberapa pesat dan tingginya kebutuhan teknologi mobile saat ini. Bukan hanya tenar di kalangan dewasa, penggunaan mobile di kalangan anak-anak pun sudah menjadi hal yang lumrah. Telekomunikasi mobile sudah selayaknya mendapatkan perhatian yang lebih dari semua kalangan agar penggunaan telekomunikasi mobile menjadi tepat sasaran dan tidak melenceng dari tujuan.

Berlatar dari kondisi inilah, paseban.com hadir sebagai solusi edukasi bagi para pengguna teknologi Informasi di Indonesia khususnya pengguna mobile. Dengan kata lain, paseban.com mempunyai visi untuk memberikan informasi yang bersifat edukasi seakurat mungkin kepada para pengguna mobile di Indonesia. Paseban.com adalah website portal yang berbasis komunitas dan mengupas tuntas secara detail mengenai perkembangan teknologi mobile saat ini, dengan harapan dapat menciptakan masyarakat mobile indonesia yang cerdas, bertanggung jawab dan peka terhadap perkembangan teknologi informasi.

Paseban.com memberikan informasi dunia telekomunikasi yang lengkap, relevan, akurat dan juga dapat dipertanggung jawabkan.

Paseban.com akan menjadi tempat bertemunya komunitas mobile di Indonesia, mulai dari Pengguna akhir layanan telekomunikasi, Pelaku industri telekomunikasi, dan Pemerintah.

Paseban.com lahir sebagai bentuk partisipasi masyarat mobile Indonesia untuk menciptakan sebuah komunitas positif yang menghubungkan para pengguna mobile di Indonesia dalam satu lingkup komunitas, saling berbagi informasi, memberikan kontribusi positif satu sama lain, dan memenuhi kebutuhan bersosialisasi dari setiap individu "user"nya.

Paseban.com menciptakan kedekatan antara konsumen dan produsen mobile serta memberikan masukan kepada produsen akan kebutuhan dan keinginan dari tiap konsumennya, sehingga memungkinkan produsen untuk tetap berkewajiban menjaga dan meningkatkan kualitas pelayanan. Hal tersebut dapat terpenuhi dengan adanya informasi lengkap mengenai data konsumen, keluhan, testimoni dan informasi lainnya yang berasal dari konsumen. Disisi lain konsumen akan mendapatkan informasi cerdas mengenai jasa dan layanan yang diberikan oleh produsen, aktivitas produsen, promo, bahkan sampai dengan informasi harga produk dan layanan yang ditetapkan oleh produsen, semuanya akan tersaji detail dan lengkap di paseban.com

Paseban.com mempermudah pemerintah dalam mengawasi kebijakan-kebijakan yang diterapkan oleh pihak produsen mobile kepada konsumen mobile serta memudahkan pemeritah untuk mensosialisasikan kebijakan-kebijakannya

kepada masyarakat telekomunikasi di Indonesia. Lebih dari itu paseban.com memfasilitasi pemerintah untuk mengetahui hal-hal yang menjadi isu terhangat di lingkungan masyarakat mobile Indonesia sehingga kedepannya pemerintah dapat membuat sebuah aturan atau kebijakan yang tepat dan sesuai dengan kondisi masyarakat mobile di Indonesia.

Dengan demikian tujuan dari adanya kebijakan yang diterapkan oleh pemerintah yaitu untuk menciptakan suatu kondisi regulasi yang bukan hanya menguntungkan masyarakat tetapi juga mewujudkan tercapainya industri mobile di Indonesia.

Paseban.com akan menjadi tempat bertemunya komunitas mobile di Indonesia, mulai dari Pengguna akhir layanan telekomunikasi, Pelaku industri telekomunikasi, dan Pemerintah.

2.1.2 Logo Instansi

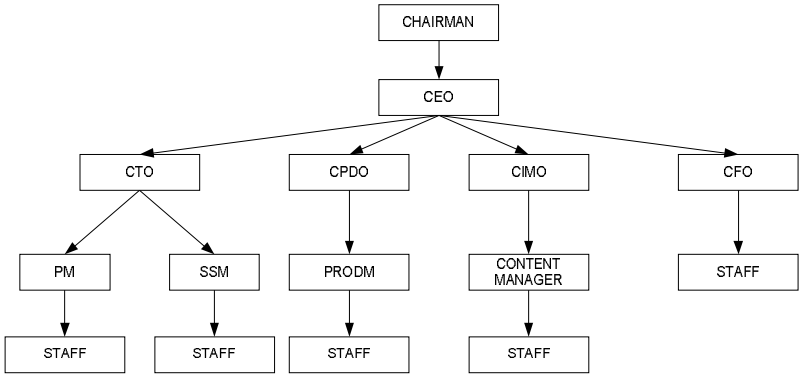
Logo instansi merupakan suatu gambar yang memiliki fungsi dan arti dari sebuah instansi untuk menggambarkan instansi itu sendiri. Logo instansi dari paseban dapat dilihat pada gambar 2.1



Gambar 2. 1 Logo Paseban

2.1.3 Struktur organisasi dan *Job Description*

Struktur organisasi sangatlah diperlukan demi kelancaran proses kinerja suatu organisasi atau perusahaan. Tingkatan atau hierarki dari kekuasaan yang tergambar, akhirnya akan menentukan hak dan kewajiban yang dimiliki oleh masing-masing individu dalam organisasi atau perusahaan tersebut. Berikut ini adalah struktur organisasi yang ada di Paseban.



Gambar 2. 2 Struktur Organisasi

Job Description

1. Chairman
    a. Memimpin seluruh dewan atau komite eksekutif.
    b. Menawarkan visi dan imajinasi di tingkat tertinggi (biasanya bekerjasama dengan MD atau CEO).
    c. Memimpin rapat umum, dalam hal: untuk memastikan pelaksanaan tata-tertib, keadilan dan kesempatan bagi semua untuk berkontribusi secara tepat; menyesuaikan alokasi waktu per item masalah;

menentukan urutan agenda; mengarahkan diskusi ke arah konsensus; menjelaskan dan menyimpulkan tindakan dan kebijakan.

d. Bertindak sebagai perwakilan organisasi dalam hubungannya dengan dunia luar.
e. Memainkan bagian terkemuka dalam menentukan komposisi dari board dan sub-komite, sehingga tercapainya keselarasan dan efektivitas.
f. Mengambil keputusan sebagaimana didelegasikan oleh BOD atau pada situasi tertentu yang dianggap perlu, yang diputuskan, dalam meeting-meeting BOD.
g. Menjalankan tanggung jawab dari direktur perusahaan sesuai dengan standar etika dan hokum.

2. CEO (*Chief Eksekutif Officer*)
   a. Bertindak sebagai perwakilan organisasi dalam hubungannya dengan dunia luar.
   b. Memainkan bagian terkemuka dalam menentukan komposisi dari board dan sub-komite, sehingga tercapainya keselarasan dan efektivitas.
   c. Mengambil keputusan sebagaimana didelegasikan oleh BOD atau pada situasi tertentu yang dianggap perlu, yang diputuskan, dalam meeting-meeting BOD.
3. CTO (*Chief Technology Officer*)

a. Mengembangkan dan menyusun strategi dan rencana TI perusahaan untuk jangka pendek, menengah, dan panjang, untuk memastikan kesesuaian dengan strategi dan rencana perusahaan dalam mencapai sasaran usaha
b. Menganalisa dan mengkaji perkembangan dan tren teknologi informasi serta pengaruhnya ke industri dan bidang usaha berkecimpung, untuk menjadi landasan bagi penetapan dan penyusunan rekomendasi pengembangan TI secara korporat.
c. Menyusun anggaran bagian TI dan mengontrol penggunaan dan realisasi dari anggaran tersebut untuk memastikan efektifitas dan efisiensinya.
d. Mengkordinir perencanaan arsitektur TI dengan membangun suatu pendekatan arsitektural TI untuk keseluruhan sistem perusahaan/ korporat, menyiapkan kapasitas perencanaan sehingga pelayanan kepada pelanggan terpelihara secara konsisten dan tidak kompromis, mempertimbangkan kreasi-nilai dalam membangun suatu arsitektur perusahaan aplikasi, dan selalu mengupdate pengetahuan tentang perkembangan TI yang mutakhir.
e. Mengarahkan dan mengkordinir penciptaan manajemen pelayanan TI yang efektif untuk seluruh cabang, unit dan korporat, dengan:
    1. Bermitra dengan provider yang melayani perusahaan untuk menjamin perolehan pelayanan yang benar dan yang dibutuhkan

oleh kegiatan usaha agar sejalan dengan proses-proses dalam perusahaan.

2. Mengkordinir pelaksanaan SLA (service level agreement) TI di seluruh perusahaan.
3. Menjamin pemberian manajemen pelayanan SLA yang tepat, demikian pula untuk pelayanan internal.
4. Menyiapkan rencana pemulihan pada upaya antisipasi terhadap tiap serangan bahaya yang sama artinya dengan menjamin sistem TI secara maksimum.
5. Mempertimbangkan biaya dan dasar-dasar dalam negosiasi kontrak.
6. Menyeleksi, merekomendasikan dan memberikan pelatihan TI yang tepat kepada staff TI di perusahaan.

f. Mengarahkan dan mengkordinir pendayagunaan software dan hardware untuk mencapai kinerja optimum di seluruh perusahaan
g. Mengkordinir perencanaan dan pelaksanaan proyek TI yang besar di seluruh perusahaan, untuk memastikan integrasi korporat dan pencapaian target penyelesaian proyek dengan tepat waktu.

4. CPDO (*Constant Proportion Debt Obligation*)
5. CIMO (*Credit Investigation and Marketing Officer*)
6. CFO (*Chief Financial Marketing*)

a. Mengkoordinir perumusan Strategi Jangka Panjang sebagai dasar perumusan Rencana Kerja dan Anggaran perusahaan (RKAP) dengan bekerja sama dengan Direksi lainnya.
b. Memberlakukan langkah-langkah yang dapat mengurangi dan menanggulangi berbagai jenis risiko finansial yang dapat dihadapi oleh perusahaan dengan berkoordinasi dengan Direksi lainnya.
c. Memastikan agar seluruh unit usaha dan wilayah kerja perusahaan mematuhi policy dan standard operating procedure (SOP) keuangan yang berlaku untuk masing-masing fungsi sesuai dengan rencana yang telah disetujui (business units oversight).
d. Membangun sinergi dan berusaha mencapai hasil bisnis yang optimal dari pelaksanaan seluruh usaha perusahaan.
e. Memastikan ketersediaan dana operasional yang dibutuhkan oleh perusahaan untuk kegiatan operasional sehari-hari, dengan melakukan koordinasi erat dengan para pimpinan unit usaha.
f. Memastikan konsolidasi keuangan yang akurat dan tepat waktu untuk keperluan pelaporan kepada Direksi dan Komisaris Perusahaan.

7. PM (*Promotion Manager*)
   a. Menyiapkan bahan atau materi promosi sesuai kebutuhan promosi suatu produk secara menyeluruh.
   b. Menganalisa spesifikasi setiap produk untuk dapat menemukan spesifikasi dan keunikan setiap produk yang akan digunakan sebagai materi promosi.

c. Mengkoordinasikan, memonitor dan menganalisa strategi marketing secara nasional setiap jangka waktu tertentu sesuai kebijakan perusahaan.
d. Menyusun usulan strategi penjualan secara nasional melalui hasil analisa pasar dan kesiapan promosi.
e. Mengarahkan dan mengkoordinasikan kegiatan-kegiatan yang diikuti sebagai sarana promosi perusahaan.
f. Menjalankan tugas-tugas terkait lainnya dalam upaya pencapaian target promosi.

8. SSM *(Sales Solution Marketing)*
   a. Menentukan harga jual, produk yang akan dilaunching, jadwal kunjungan serta system promosi untuk memastikan tercapainya target penjualan.
   b. Memonitor perolehan order serta merangkumkan forecast untuk memastikan kapasitas produksi terisi secara optimal
   c. Memonitor jumlah stock seluruh Dept. Sales & Marketing untuk memastikan umur stock perusahaan tidak melebihi target yang telah ditentukan.
   d. Menganalisa dan mengembangkan strategi marketing untuk meningkatkan jumlah pelanggan dan area sesuai dg target yang ditentukan.

e. Menganalisa dan memberikan arah pengembangan design & warna, untuk memastikan pengembangan produk sesuai dengan kebutuhan pasar.
f. Melakukan evaluasi kepuasan pelanggan dari hasil survey seluruh sales team untuk memastikan tercapainya target kepuasan pelanggan yang ditentukan.
g. Menerapkan budaya, sistem, dan peraturan intern perusahaan serta menerapkan manajemen biaya, untuk memastikan budaya perusahaan dan sistem serta peraturan dijalankan dengan optimal.

9. PRODM (*Production Manager*)
    a. Mempersiapkan program kerja dan mendapatkan persetujuan dari kepala dinas.
    b. Mengatur dan mengawasi kegiatan operasional harian di departemen.
    c. Mengadakan pertemuan rutin dengan laporan langsung untuk mengatasi masalah teknis.
    d. Menjamin semua proses permasalahan.
    e. Bernegosiasi dengan pemasok dalam produksi.
    f. Secara teratur mengunjungi pemasok di berbagai tempat.
    g. Meningkatkan pengetahuan produk dan setiap kemungkinan inovasi dalam produksi.
10. Content Manager
11. Staff
    a. Melakukan pemeliharaan dan perbaikan pada setiap unit alat kerja (komputer) sesuai dengan waktu yang diberikan.

b. Mampu menyelesaikan pekerjaan administrasi dengan baik.
c. Menjalin kerjasama yang baik dengan rekan-rekan sekerja.
d. Bertanggung jawab kepada atasan langsung & Manager IT

2.1.4 Visi dan Misi Perusahaan

a. Visi

"To Be The Biggest Mobile Portal & Community Website In Indonesia"

b. Misi

1. Memberikan informasi dunia telekomunikasi yang lengkap, relevan, akurat dan dapat dipertanggung jawabkan.
2. Menciptakan media informasi yang interaktif dan menciptakan sarana edukasi untuk para pengguna mobile di Indonesia
3. Membuat sarana untuk memenuhi kebutuhan serta keinginan pengguna layanan telekomunikasi dalam aplikasi dan tools yang bermanfaat
4. Fasilitator pengembangan bisnis industri telekomunikasi

2.2 Landasan Teori

2.2.1 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.

2.2.2 Sistem Informasi

2.2.2.1 Konsep Sistem

Sistem adalah sekumpulan unsur atau elemen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai tujuan. Menurut L. Akcof “Sistem adalah setiap kesatuan konseptual atau fisik yang terdiri dari bagian-bagian dalam keadaan saling tergantung sata sama lainya”.

2.2.2.2 Karakteristik Sistem

1. Memilki komponen

   Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, bekerja sama membentuk satu kesatuan. Komponen-komponen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap sistem tidak peduli betapapun kecilnya, selalu mengandung komponen-komponen atau subsistem-subsistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai suatu sistem yang lebih besar yang disebut supra sistem, misalnya suatu perusahaan dapat disebut dengan suatu sistem dan indutri yang merupakan sistem yang lebih besar dapat disebut dengan supra sistem. Kalau dipandang industri sebagai suatu sistem, maka perusahaan dapat dipandang sabagai suatu sistem, maka sistem akuntansi adalah subsistemnya.

2. Batas Sistem

   Boundary system ‘batas sistem’ merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainya atau merupakan dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan. Batas suatu sistem menunjukan scope ‘ruang lingkup’ dari sistem tersebut.

3. Lingkungan luar system

   Adalah apapun diluar batas dari sistem yang mempengruhi operasi sistem.

4. Interface

Interface ‘penghubung sistem’ merupakan media penghungung antara satu subsistem dengan subsistem yang lainnya.

5. Input

   Input ‘masukan sistem’ merupakan energi yang dimasukan kedalam sistem. Masukan dapat berupamaintenance input ‘masukan perawatan’ dan signal input ‘masukan sinyal’.

6. Keluaran Sistem

   Merupakan output dari input yang diproses oleh sistem.

7. Pemrosesan Sistem

   Merupakan bagian yang memproses masukan untuk menjadi keluaran yang diinginkan.

8. Sasaran system

   Kalau sistem tidak mempunyai sasaran, maka operasi sistem tidak ada gunanya.

Secara garis besari sistem dapat dibedakan menjadi 2, yaitu

1. Sistem Fisik

   Sistem fisik atau physical system ialah kumpulan elemen-elemen atau unsur-unsur yang saling berinteraksi satu sama lain secara fisik serta dapat diidentifikasi secara nyata tujuan-tujuannya.

2. Sistem Abstrak

   Sistem abstrak atau *abstract system* ialah sistem yang dibentuk akibat terselenggaranya ketergantungan ide dan dapat diidentifikasikan secara nyata, tetapi dapat diuraikan elemen-elemennya.

2.2.2.3 Konsep Informasi

Informasi ialah data yang telah diproses menjadi bentuk yang memiliki arti bagi penerima dan dapat berupa fakta, suatu nilai yang bermanfaat. Jadi ada suatu proses transformasi data menjadi suatu informasi.

Data merupakan raw material untuk suatu informasi. Perbedaan informasi dan data sangat relatif tergantung pada nilai gunanya bagi manajemen yang memerlukan. Suatu informasi bagi level manajemen tertentu bisa menjadi data bagi manajemen level diatasnya, atau sebaliknya.

Kualitas informasi tergantung pada 3 hal, yaitu informasi harus:

1. Akurat. Berarti informasi harus bebeas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat berarti juga informasi harus jelas mencerminkan maksudnya.
2. Tepat pada waktunya. Berarti informasi yang data pada penerima tidak boleh terlmbat.
3. Relevan. Berarti informasi tersebut mempunyai manfaat untuk pemakainya. Relevansi informasi untuk tiap-tiap orang satu dengan lainya mungkin berbeda.

2.2.2.4 Definisi Sistem Informasi

Sistem informasi ialah sebuah sistem yang terintregrasi atau sistem manusia-mesin, untuk menyediakan informasi untuk mendukung operasi, manajemen dalam suatu organisasi.

Sistem ini memanfaatkan perangkat keras dan perangkat lunak komputer, prosedur manual, model manajemen dan basis data.

Menurut Robert A. Leicth “Sistem infornasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manejerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

2.2.3 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 peranti 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. Pada saat perilisan perdana Android, 5 November 2007, Android bersama Open Handset Alliance menyatakan mendukung pengembangan standar terbuka pada perangkat seluler. Di lain pihak, Google merilis kode–kode Android di bawah lisensi Apache, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler. Di dunia ini terdapat dua jenis distributor sistem operasi Android. Pertama yang mendapat dukungan penuh dari Google atau Google Mail Services (GMS) dan kedua adalah yang benar–benar bebas distribusinya tanpa dukungan langsung Google atau dikenal sebagai Open Handset Distribution (OHD).

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. Pada tahun 2005 Google mengakusisi Android Inc yang pada saat itu dimotori oleh Andy Rubin, Rich Miner, Nick Sears dan Crish White. Yang kemudian pada tahun itu juga memulai membangun platform Android secara lebih intensif. Kemudian pada tanggal 12 November 2007 Google bersama Open Handset Aliance (OHA) yaitu konsorium perangkat lunak mobile terbuka, merilis Google Android SDK, setelah mengumumkannya seminggu sebelumnya. Dan sambutannya sangat luar biasa , hampir semua media berita tentang IT dan Programming memberitakan tentang dirilisnya Android SDK (Software Developtment Kit). Ini dikarenakan ide dari flatform Android sangat menarik untuk developer dan programmer di seluruh penjuru dunia. Di sisi lain, produsen pembuat handset ponsel juga segera berlomba-lomba membuat handset dengan platform Android ini. Google bersama dengan OHA merilis paket software SDK yang lengkap untuk mengembangkan aplikasi pada perangkat mobile, yaitu : Sistem Operasi, Middleware dan Aplikasi utama untuk perangkat mobile. Sebagai

Programmer atau Developer kita bisa melakukan segalanya, mulai dari membuat aplikasi pengiriman SMS hanya dengan dua baris kode, hingga mengganti even pada Home Screen perangkat Android. Selain itu, bahkan dengan mudah kita bisa membuatdan mengkostumisasi Sistem Operasinya, atau mengganti semua aplikasi default dari google.

Semua aplikasi yang dibuat untuk android akan memiliki akses setara dalam mengakses seluruh kemampuan handset, tanpa membedakan apakah itu merupakan aplikasi inti atau aplikasi pihak ketiga. Dalam kata lain dengan platform android ini, Programer atau Developer secara penuh akan bisa mengkustomisas perangkat androidnya. Android built in pada Linux kernel (Open Linux Kernel), dengan sebuah mesin virtual yang telah didesain dan untuk mengoptimalkan penggunaan sumberdaya memori dan handware pada lingkungan perangkat mobile (Mobile Environment). Dalvik adalah nama dari Android Virtual Mesin, yang merupak interpreter virtual mesin yang akan mengeksekusi file kedalam format Dalvik Executable (*.dex). Sebuah format yang dirancang untuk ruang penyimpanan yang efisien dan eksekusi memori yang terpetakan (memory-mappable execution).

Dlavik Virtual Mesun (Dalvik VM) berbasi register , dan dapat mengeksekusi kelas (class) yang telah terkompilasi pada compier bahasa java, kemudian di transformasikan kedalam native format dengan menggunakan Tool “dx” yang telah terintegrasi. Kita mungkin telah mengenal JavaVM (Java Virtual Mesin), yang saat ini bisa kita temukan pada setiap komputer desktop. Berbeda denga DalvikVM, JavaVM berbasis stack (Stack-based Virtual Machine), DalvikVM

memiliki keunggulan dengan menggunakan Registred Based, ini karena pada prosesor perangkat genggam telah dioptimasi untuk eksekusi berbasis register . Android saaat ini tidak hanya berjalan pada handphone, beberapa vendor menanamkan Android pada tablet, Internet Tablet, E-Book Reader, Laptop, dan gadget lainnya.

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 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. Beberapa versi dari android yang sudah diluncurkan diantaranya:

Andorid 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.

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.

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.

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.

Android versi 2.2 (Froyo: Frozen Youghurt)

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.

Android versi 2.3 (Gingerbread)

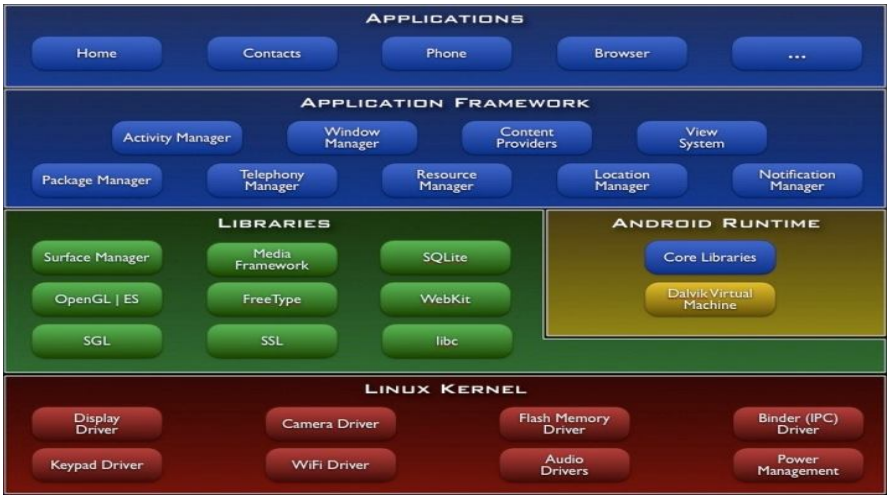
Pada 6 Desember 2010, Android versi 2.3 (Gingerbread) diluncurkan. Perubahan-perubahan umum yang didapat dari Android versi ini antara lain peningkatan kemampuan permainan (gaming), peningkatan fungsi copy paste, layar antar muka (User Interface) didesain ulang, dukungan format video VP8 dan WebM, efek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

2.2.3.1 Features

Features yang tersedia pada platform Android saat ini antara lain :

a. Framework Aplikasi yang mendukung penggantian komponen dan reusable.
b. Mesin Virtual Dalvik berjalan diatas Linux kernel dan dioptimalkan untuk perangkat mobile
c. Integrated browser berdasarkan open source engine WebKit
d. Grafis yang dioptimalkan dan didukung oleh library grafis 2D yang terkostumisasi, grafis 3D berdasarkan spesifikasi openGL ES 1,0 (Opsional akselarasi hardware)
e. SQLite untuk penyimpanan data
f. Media support yang mendukung audio, video, dan gambar (MPGE, H.264, MP3, AAC, AMR, JPG, PNG, GIF)
g. GSM Telephony (tergantung hardware)
h. Bluetooth, EDGE, 3G, dan WIFI (tergantung hardware)
i. Multi-touch : kemampuan layaknya handset modern yang dapat menggunakan dua jari atau lebih untuk berinteraksi dengan perangkat
j. Lingkungan Development yang lengkap dan kaya termasuk perangkat emulator, tools untuk debugging, profil, dan kinerja memori, dan plugin untuk Eclips IDE
k. Market : Seperti kebanyakan Handphone yang memiliki tempat penjualan aplikasi, market pada android merupakan katalog aplikasi yang dapat di download dan di install pada Handphone melalui internet

2.2.3.2 Arsitektur Android



Gambar 2. 3 Arsitektur android

2.2.3.3 Linux Kernel

Android bukan Linux, akan tetapi android dibangun diatas Linux Kernel yaitu versi 2.6 sehingga kehandalannya dapat di percaya. Untuk inti sistem service linux yang digunakan seperti keamanan, manajemen memori, proses manajemen, network, dan drive model. Seperti yang terlihat digambar, Linux Kernel menyediakan Driver Layer, Kamera, Keypad, Kamera, Wifi, Flash Memory, Audio, dan IPC (Interprocess Communication) untuk mengatur aplikasi dan keamanan. Kernel juga bertindak sebagai lapisan abstrak antara hardware dan software stacknya

2.2.3.4 Libraries

Android menyertakan libraries C/C++ yang digunakan oleh berbagai komponen dari sestem Android. Kemampuan ini disediakan kepada Developer aplikasi melalui framework aplikasi Android. beberapa inti libraries tercantum di bawah ini :

a. System C library – Variasi dari implementasi BSD-berasal pelaksanaan sistem standar C library (libe), sesuai untuk perangkat embedded berbasis linux.
b. Media libraries – PaketVideo berdasarkan OpenCORE; library mendukung pemutaran rekaman dan populer banyak format audio dan video, serta file gambar, termasuk MPEG, H.264, MP3, AAC, AMR, JPG. Dan PNG
c. Surface Manager – mengelola akses ke subsistem layar
d. LibWebCore – mesin web modern yang powerfull yang baik browser Android embeddable web.
e. SGL – mesin grafis 2D
f. 3D libraries – sebuah pelaksanaan berdasarkan openGL ES 1,0 API; perpustakaan baik menggunakan perangkat keras akselarasi 3D (apabila tersedia) atau yang disertakan, sangat optimal 3D software rasterizer
g. FreeType – bitmap dan vektor font rendering
h. SQLite – mesin database yang kuat dan ringan, dan penghubung tersedia untuk semua aplikasi.

2.2.3.5 Android-Runtime

Android terdiri satu set core libraries yang menyediakan sebagian besar fungsi yang sama dengan yang terdapat pada core libraries bahasa pemograman java. Setiap aplikasi menjalankan prosesnya sendiri dalam Android. Setiap alikasi menjalankan prosesnya sendiri dalam Android, dengan masing-masing instan dari mesin virtual Dalvik (Dalvik VM). Dalvik dirancang agar perangkat dapat menjalankan multiple VMs secara efisien. Mesin Virtual Dalvik mengeksekusi file dalam Dalvik executable (.dex), sebuah format yang dioptimalkan untuk memori yang kecil. Dalvik VM berbasis, berjalan dan dikompilasi oleh compiler bahasa Java yang telah ditransformasikan ke dalam .dex format oleh tool “dx” yang telah di sertakan, Dalvik VM bergantung pada kernel Linux untuk berfungsi seperti threading dan manajemen memori tingkat rendahnya.

2.2.3.6 Framework-Aplikasi

Pengembang memiliki akses penuh menuju API framework yang sama, dengan yang digunakan oleh aplikasi inti. Arsitektur aplikasi dirancang agar komponen dapat digunakan kembali (reuse) denga mudah. Setiap aplikasi dapat memanfaatkan kemampuan ini dan aplikasi yang lain mungkin akan memanfaatkan kemampuan ini (sesuai denganbatasan keamanan yang didefinisikan oleh framework). Mekanisme yang sama memungkinkan komponen untuk diganti oleh pengguna.

Semua pengguna merupakan rangkaian set layanan dan sistem, termasuk:

a. View yang kaya dan extensible yang dapat digunakan untuk membangun aplikasi, termasuk list, grid, kotak teks, tombol, dan bahkan sebuah embeddable web
b. Content provider yang memungkinkan aplikasi untuk mengakses data (seperti dari daftar kontak help) atau membagi data tersebut.
c. Resource Manager, yang menyediakan akses ke kode non sumber lokal seperti string, gambar, dan tata letak file
d. Notification Manager yang memungkinkan semua aplikasi menampilkan alert yang bisa dokostumisasi didalam status bar.
e. Activity Manager yang mengelola siklus hidup aplikasi dan menyediakan navigasi umum backstack.

2.2.3.7 Application

Android telah menyertakan aplikasi inti seperti email elient, SMS, kalender, peta, browser, kontak, dan lain-nya. Semua aplikasi tersebut ditulis dengan menggunakan bahasa pemrograman Java. Pada layer inilah developer atau kita menempatkan aplikasi yang dibuat. Yang istimewa adalah pada android semua aplikasi baik aplikasi inti (native) maupun aplikasi pihak ketiga berjalan pada layer aplikasi yang dibuat untuk android akan memiliki akses yang setara dalam mengakses seluruh kemampuan handset, tanpa membedakan apakah itu merupakan aplikasi inti atau aplikasi pihak ketiga. Dalam kata lain dengan platform android ini, Programmer atau developer secara penuh akan bisa mengkostumisasi perangkat androidnya.

2.2.3.8 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.

2.2.4 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.
4. 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*.

2.2.4.1 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.

2.2.4.2 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.

2.2.4.3 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

2.2.5 *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.

2.2.6 *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.

2.2.7 *Android Development Tool* (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.

2.2.8 Pemrograman

2.2.8.1 Pemrograman Berorientasi Objek

Pemograman berorientasi objek (*Obhect oriented programming – OOP*) merupakan paradigma pemograman yang berorientsikan kepada objek. Semua data dan fungsi di dalam paradigma ini dibungkus dalam kelas-kelas atau objek-objek. Bandingkan dengan logika pemrograman terstruktur. Setiap objek dapat menerima pesan, memproses data, dan mengirim pesan ke objek lainnya,

Model data berorientasi objek dikatakan dapat memberi fleksibilitas yang lebih, kemudahan mengubah program, dan digunakan luas dalam teknik piranti lunak skala besar. Lebih jauh lagi, pendukung OOP mengklaim bahwa OOP lebih mudah dipelajari bagi pemula dibanding dengan pendekatan sebelumnya, dan pendekatan OOP lebih mudah dikembangkan dan dirawat.

A. Konsep Pemrograman Berorientasi Objek

Pemrograman berorientasiobjek menekankan konsep berikut:

1. kelas, kumpulan atas definisi data dan fungsi-fungsi dalam suatu unit untuk suatu tujuan tertentu. Sebagai contoh 'class of dog' adalah suatu unit yang terdiri atas definisi-definisi data dan fungsi-fungsi yang menunjuk pada berbagai macam perilaku/turunan dari anjing. Sebuah class adalah dasar dari modularitas dan struktur dalam pemrograman berorientasi oobjek. Sebuah class secara tipikal sebaiknya dapat dikenali oleh seorang non-programmer sekalipun terkait dengan domain permasalahan yang ada, dan kode yang terdapat dalam sebuah class sebaiknya (relatif) bersifat mandiri dan independen (sebagaimana kode tersebut digunakan jika tidak

menggunakan OOP). Dengan modularitas, struktur dari sebuah program akan terkait dengan aspek-aspek dalam masalah yang akan diselesaikan melalui program tersebut. Cara seperti ini akan menyederhanakan pemetaan dari masalah ke sebuah program ataupun sebaliknya.

2. Objek, membungkus data dan fungsi bersama menjadi suatu unit dalam sebuah program komputer. Objek merupakan dasar dari modularitas dan struktur dalam sebuah program komputer berorientasi objek.
3. Abstraksi , kemampuan sebuah program untuk melewati aspek informasi yang diproses olehnya, yaitu kemampuan untuk memfokus pada inti. Setiap objek dalam sistem melayani sebagai model dari "pelaku" abstrak yang dapat melakukan kerja, laporan dan perubahan keadaannya, dan berkomunikasi dengan objek lainnya dalam sistem, tanpa mengungkapkan bagaimana kelebihan ini diterapkan. Proses, fungsi atau metode dapat juga dibuat abstrak, dan beberapa teknik digunakan untuk mengembangkan sebuah pengabstrakan.
4. Enkapsulasi, memastikan pengguna sebuah objek tidak dapat mengganti keadaan dalam dari sebuah objek dengan cara yang tidak layak; hanya metode dalam objek tersebut yang diberi ijin untuk mengakses keadaannya. Setiap objek mengakses interface yang menyebutkan bagaimana objek lainnya dapat berinteraksi dengannya. Objek lainnya tidak akan mengetahui dan tergantung kepada representasi dalam objek tersebut.

5. Polimorfisme melalui pengiriman pesan. Tidak bergantung kepada pemanggilan subrutin, bahasa orientasi objek dapat mengirim pesan; metode tertentu yang berhubungan dengan sebuah pengiriman pesan tergantung kepada objek tertentu di mana pesa tersebut dikirim. Contohnya, bila sebuah burung menerima pesan "gerak cepat", dia akan menggerakan sayapnya dan terbang. Bila seekor singa menerima pesan yang sama, dia akan menggerakkan kakinya dan berlari. Keduanya menjawab sebuah pesan yang sama, namun yang sesuai dengan kemampuan hewan tersebut. Ini disebut polimorfisme karena sebuah variabel tungal dalam program dapat memegang berbagai jenis objek yang berbeda selagi program berjalan, dan teks program yang sama dapat memanggil beberapa metode yang berbeda di saat yang berbeda dalam pemanggilan yang sama. Hal ini berlawanan denganbahasa fungsional yang mencapai polimorfisme melalui penggunaan fungsi kelas-pertama.
6. Inheritas- Mengatur polimorfisme dan enkapsulasi dengan mengijinkan objek didefinisikan dan diciptakan dengan jenis khusus dari objek yang sudah ada - objek-objek ini dapat membagi (dan memperluas) perilaku mereka tanpa haru mengimplementasi ulang perilaku tersebut (bahasa berbasis-objek tidak selalu memiliki inheritas.)
7. Dengan menggunakan OOP maka dalam melakukan pemecahan suatu masalah kita tidak melihat bagaimana cara menyelesaikan suatu masalah tersebut (terstruktur) tetapi objek-objek apa yang dapat melakukan pemecahan masalah tersebut. Sebagai contoh anggap kita memiliki sebuah

departemen yang memiliki manager, sekretaris, petugas administrasi data dan lainnya. Misal manager tersebut ingin memperoleh data dari bag administrasi maka manager tersebut tidak harus mengambilnya langsung tetapi dapat menyuruh petugas bag administrasi untuk mengambilnya. Pada kasus tersebut seorang manager tidak harus mengetahui bagaimana cara mengambil data tersebut tetapi manager bisa mendapatkan data tersebut melalui objek petugas adminiistrasi. Jadi untuk menyelesaikan suatu masalah dengan kolaborasi antar objek-objek yang ada karena setiap objek memiliki deskripsi tugasnya sendiri.

2.2.9 Java

Java adalah bahasa pemograman yang dapat dijalankan di berbagai berbagai komputer termasuk telepon genggam. Dikembangkan oleh Sun Microsystems dan dirilis tahun 1995. Java berbeda dengan JavaScript. JavaScript adalah bahasa scripting yang digunakan oleh web browser.



Gambar 2. 4 Logo JAVA

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

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

Kelebihan:

1. Multiplatform. Kelebihan utama dari Java ialah dapat dijalankan di beberapa platform / sistem operasi komputer, sesuai dengan prinsip tulis sekali, jalankan di mana saja. Dengan kelebihan ini pemrogram cukup menulis sebuah program Java dan dikompilasi (diubah, dari bahasa yang dimengerti manusia menjadi bahasa mesin / bytecode) sekali lalu hasilnya dapat dijalankan di atas beberapa platform tanpa perubahan. Kelebihan ini memungkinkan sebuah program berbasis java dikerjakan diatas operating system Linux tetapi dijalankan dengan baik di atas Microsoft Windows.

Platform yang didukung sampai saat ini adalah Microsoft Windows, Linux, Mac OS dan Sun Solaris. Penyebanya adalah setiap sistem operasi menggunakan programnya sendiri-sendiri (yang dapat diunduh dari situs Java) untuk meninterpretasikan bytecode tersebut.

2. Objek Oriented Programming (OOP) yang artinya semua aspek yang terdapat di Java adalah objek. Java merupakan salah satu bahasa pemrograman berbasis objek secara murni. Semua tipe data diturunkan dari kelas dasar yang disebut objek. Hal ini sangat memudahkan pemrogram untuk mendesain, membuat, mengembangkan dan mengalokasi kesalahan sebuah program dengan basis Java secara cepat, tepat, mudah dan terorganisir. Kelebihan ini menjadikan Java sebagai salah satu bahasa pemograman termudah, bahkan untuk fungsi fungsi yang advance seperti komunikasi antara komputer sekalipun.
3. Perpustakaan kelas Yang Lengkap. Java terkenal dengan kelengkapan library/perpustakaan (kumpulan program-program yang disertakan dalam pemrograman java) yang sangat memudahkan dalam penggunaan oleh para pemrogram untuk membangun aplikasinya. Kelengkapan perpustakaan ini ditambah dengan keberadaan komunitas Java yang besar yang terus menerus membuat perpustakaan-perpustakaan baru untuk melingkupi seluruh kebutuhan pembangunan aplikasi.
4. Bergaya C++, memiliki sintaks seperti bahasa pemrograman C++ sehingga menarik banyak pemrogram C++ untuk pindah ke Java. Saat ini pengguna Java sangat banyak, sebagian besar adalah pemrogram C++ yang pindah

ke Java. Universitas-universitas di Amerika Serikat juga mulai berpindah dengan mengajarkan Java kepada murid-murid yang baru karena lebih mudah dipahami oleh murid dan dapat berguna juga bagi mereka yang bukan mengambil jurusan komputer.

5. Pengumpulan sampah otomatis, memiliki fasilitas pengaturan penggunaan memori sehingga para pemrogram tidak perlu melakukan pengaturan memori secara langsung (seperti halnya dalam bahasa C++ yang dipakai secara luas).

Kekurangan :

1. Tulis sekali, perbaiki di mana saja. Masih ada beberapa hal yang tidak kompatibel antara platform satu dengan platform lain. Untuk J2SE, misalnya SWT-AWT bridge yang sampai sekarang tidak berfungsi pada Mac OS X.
2. Mudah didekompilasi. Dekompilasi adalah proses membalikkan dari kode jadi menjadi kode sumber. Ini dimungkinkan karena kode jadi Java merupakan bytecode yang menyimpan banyak atribut bahasa tingkat tinggi, seperti nama-nama kelas, metode, dan tipe data. Hal yang sama juga terjadi pada Microsoft .NET Platform. Dengan demikian, algoritma yang digunakan program akan lebih sulit disembunyikan dan mudah dibajak/direverse-engineer.
3. Penggunaan memori yang banyak. Penggunaan memori untuk program berbasis Java jauh lebih besar daripada bahasa tingkat tinggi generasi sebelumnya seperti C/C++ danPascal (lebih spesifik lagi, Delphi dan

Objek Pascal). Biasanya ini bukan merupakan masalah bagi pihak yang menggunakan teknologi terbaru (karena trend memori terpasang makin murah), tetapi menjadi masalah bagi mereka yang masih harus berkutat dengan mesin komputer berumur lebih dari 4 tahun.

2.2.8.2 *Unified Modelling Language* (UML)

Unified Modeling Language (UML) adalah bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan, dan membangun sistem perangkat lunak.

Unified Modeling Language (UML) adalah himpunan struktur danteknik untuk pemodelan desain program berorientasi objek (OOP) serta aplikasinya.UML adalah metodologi untuk mengembangkansistem OOP dan sekelompok perangkat tool untuk mendukung pengembangan sistem tersebut UML mulai diperkenalkan oleh Objek Management Group, sebuah organisasi yang telah mengembangkan model, teknologi, dan standar OOP sejak tahun 1980-an. Sekarang UML sudah mulai banyak digunakan oleh para praktisi OOP. UML merupakan dasar bagi perangkat (tool) desain berorientasi objek dari IBM

UML adalah suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan suatu sistem informasi. UML dikembangkan sebagai suatu alat untuk analisis dan desain berorientasi objek oleh Grady Booch, Jim Rumbaugh, dan Ivar Jacobson.Namun demikian UML dapat digunakan untuk memahami dan mendokumentasikan setiap sistem informasi.Penggunaan UML dalam industri terus meningkat. Ini merupakan

standar terbuka yang menjadikannya sebagai bahasa pemodelan yang umum dalam industri peranti lunak dan pengembangan sistem.

A. Diagram UML

UML menyediakan 10 macam diagram untuk memodelkan aplikasi berorientasi objek, yaitu:

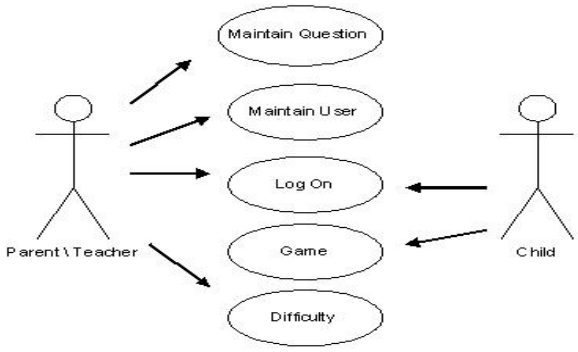
1. Use Case Diagram untuk memodelkan proses bisnis.
2. Conceptual Diagram untuk memodelkan konsep-konsep yang ada di dalam aplikasi.
3. Sequence diagram untuk memodelkan pengiriman pesan (message) antar objek.
4. Collaboration Diagram untuk memodelkan interaksi antar objek.
5. State Diagram untuk memodelkan perilaku objek di dalam sistem.
6. Activity diagram untuk memodelkan perilaku userdan objek di dalam sistem.
7. Class Diagram untuk memodelkan struktur kelas.
8. Objek Diagram untuk memodelkan struktur objek.
9. Component Diagram untuk memodelkan komponen objek.
10. Deployment Diagram untuk memodelkan distribusi aplikasi.

Berikut akan dijelaskan 4 macam diagram yang paling sering digunakan dalam pembangunan aplikasi berorientasi objek, yaitu *use case diagram, sequence diagram, collaboration diagram, dan class diagram*.

B. Use Case Diagram

Use case diagram digunakan untuk memodelkan bisnis proses berdasarkan perspektif pengguna sistem. Use case diagram terdiri atas diagram untuk use case dan actor. Actormerepresentasikan orang yang akan mengoperasikan atau orang yang berinteraksi dengan sistem aplikasi.

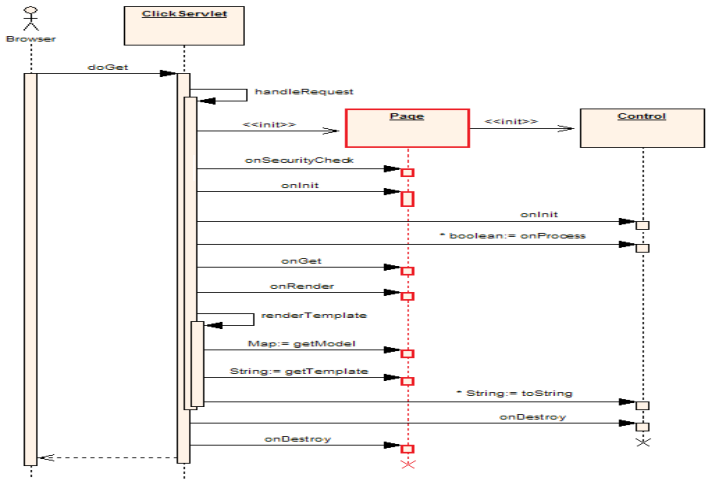
Use case merepresentasikan operasi-operasi yang dilakukan oleh actor.Use case digambarkan berbentuk elips dengan nama operasi dituliskan didalamnya. Actor yang melakukan operasi dihubungkan dengan garis lurus ke use case.



Gambar 2. 5 Use Case Diagram

C. Sequence diagram

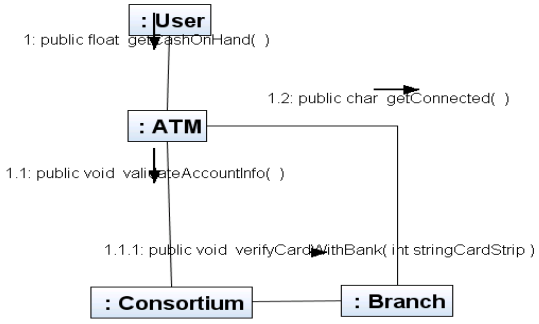
Sequence diagram menjelaskan secara detil urutan proses yang dilakukan dalam sistem untuk mencapai tujuan dari use case. Interaksi yang terjadi antar class, operasi apa saja yang terlibat, urutan antar operasi, dan informasi yang diperlukan oleh masing-masing operasi.



Gambar 2. 6 Sequence Diagram

D. Collaboration Diagram

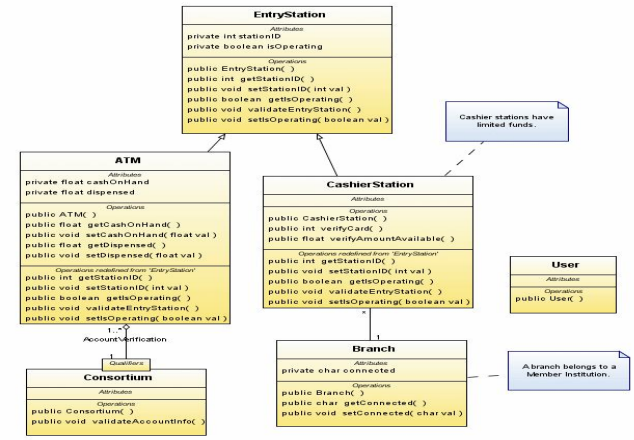
Collaboration diagram dipakai untuk memodelkan interaksi antar objek di dalam sistem.Berbeda dengan sequence diagram yang lebih menonjolkan kronologis dari operasi-operasi yang dilakukan, collaboration diagram lebih fokus pada pemahaman atas keseluruhan operasi yang dilakukan oleh objek.



Gambar 2. 7 Collaboration Diagram

E. Class Diagram

Class diagram merupakan diagram yang selalu ada di permodelan sistem berorientasi objek. Class diagram menunjukkan hubungan antar class dalam sistem yang sedang dibangun dan bagaimana mereka saling berkolaborasi untuk mencapai suatu tujuan.



Gambar 2. 8 Class Diagram