BAB II
LANDASAN TEORI

2.1. Aplikasi

Definisi aplikasi merupakan kumpulan dari prosedur-prosedur yang digunakan untuk mengolah data menjadi informasi. Misalnya penjumlahan, klasifikasi, rotasi, koreksi geometri, query, *overlay*, *buffer*, *jointable* dan sebagainya.

Menurut Daryanto (2004:347), aplikasi adalah *software* atau perangkat lunak yang dibuat untuk mengerjakan menyelesaikan masalah-masalah khusus. Sedangkan menurut Jogiyanto (2004:4), aplikasi merupakan program yang berisikan perintah-perintah untuk melakukan pengolahan data. Jadi aplikasi secara umum adalah suatu proses dari cara manual yang ditransformasikan ke komputer dengan membuat sistem atau program.

Program inilah yang mengendalikan semua aktifitas yang ada pada proses. Program berisi konstruksi logika yang dibuat oleh manusia, dan sudah diterjemahkan ke dalam bahasa mesin sesuai dengan format yang ada pada *instructionset*.

2.2. Klasifikasi Aplikasi

Aplikasi dapat digolongkan menjadi beberapa kelas, antara lain :

1. Perangkat lunak perusahaan (*enterprise*)

2. Perangkat lunak infrastruktur perusahaan
3. Perangkat lunak informasi kerja
4. Perangkat lunak media dan hiburan
5. Perangkat lunak pendidikan
6. Perangkat lunak pengembangan media
7. Perangkat lunak rekayasa produk

2.3 Konsep *E-learning*

Istilah *e-learning* sering digunakan ketika kegiatan pendidikan menggunakan media computer atau internet. Banyak pula penggunaan terminologi yang memiliki arti yang hampir sama dengan e-*learning*, seperti *Web Based Learning, online learning, computer-based training / learning, distance learning computer-aided instruction,* adalah terminologi yang sering digunakan untuk menggantikan *e-learning.* Terminologi *e-learning* sendiri dapat mengacu pada semua kegiatan pelatihan yang menggunakan media elektronik atau teknologi informasi. Banyak perubahan yang sangat cepat tentang *e-learning*, sebelum kata '*e-learning*' menjadi popular banyak kata-kata pembelajaran yang telah digunakan dan masih tetap digunakan seperti terlihat dibawah ini :

1. Pembelajaran jarak jauh (*open distance learning*)
2. Pengajaran berbasis Web (*web based training*)
3. Pengajaran berbantuan komputer (*computer based training*)
4. Pembelajaran berbasis teknologi (*technology based learning*)
5. Pembelajaran secara online (*online learning*)

Pembelajaran baik secara formal maupun informal yang dilakukan melalui media elektronik, seperti *Internet, Intranet, Ektranet,* CDROM, video tape, DVD, TV, Handphone, PDA, dll. Bentuk-bentuk seperti diatas tidaklah semuanya sama. *E-learning* lebih luas dibandingkan dengan *online learning*. *Online learning* hanya menggunakan *Internet* / *intranet* / LAN / WAN tidak termasuk menggunakan CDROM.

Ada bermacam-macam penggunaan *e-learning* saat ini, terdapat pembagian atau pembedaan *e-learning*. Pada dasarnya *e-learning* mempunyai dua tipe, yaitu *synchronous* dan *asynchronous*.

1. *Synchronous Training*

   Synchronous Training adalah tipe pelatihan, dimana proses pembelajaran terjadi pada saat yang sama ketika pengajar sedang mengajar dan murid sedang belajar. Terjadi kemungkinan interaksi langsung antara guru dan murid, baik melalui internet maupun *intranet*. *Synchronous Training* mengharuskan guru dan semua murid mengakses *internet* bersamaan. Pengajar memberikan makalah dengan slide presentasi dan peserta *web conference* dapat mendengarkan presentasi melalui hubungan *internet*. Peserta pun dapat mengajukan pertanyaan atau komentar melalui *chat window*. *Synchronous Training* sifatnya mirip pelatihan di ruang kelas. Namun, kelasnya bersifat maya (*virtual*) dan peserta tersebar di seluruh dunia dan terhubung melalui *internet*.

2. *Asynchronous Training*

   Asynchronous Training ini lebih popular di dunia *e-learning* karena memberikan keuntungan lebih bagi peserta pelatihan karena dapat mengakses pelatihan kapanpun dan dimanapun. Pelatihan ini berupa paket pelajaran yang dapat dijalankan di komputer manapun dan tidak melibatkan interaksi dengan pengajar atau pengajar lain. Oleh karena itu, pelajar dapat memulai pelajaran dan menyelesaikannya setiap saat. Paket pelajaran berbentuk bacaan dengan animasi, simulasi, visualisasi, permainan edukatif, maupun latihan atau tes dengan jawabannya. Ada beberapa keuntungan dan keterbatasan dalam penggunaan *e-learning*. Keuntungan dan keterbatasan adalah sebagai berikut

2.4 Keuntungan dari *E-learning*

1. Biaya

   Kelebihan pertama *e-learning* adalah mampu mengurangi biaya pelatihan. Dengan adanya *e-learning,* tidak perlu mengeluarkan biaya untuk menyewa pelatih dan ruang kelas serta transportasi peserta latihan atau pelatih.

2. Fleksibilitas Waktu

   E-learning membuat pelajar dapat menyesuaikan waktu belajar, dapat mengakses *e-learning* disaat waktu senggang. Ketika waktu sudah tidak memungkinkan atau ada hal yang lebih mendesak,

pengguna akses dapat meninggalkan pelajaran *e-learning* saat itu juga.

3. Fleksibilitas Tempat

   Dengan jangkauan *internet* yang sangat luas, akan memudahkan pelajar untuk mengakses *e-learning* tanpa perlu menempuh perjalanan jauh ke kelas. Selama komputer terhubung dengan komputer yang menjadi server *e-learning*, pengguna akases dapat mengaksesnya dengan mudah.

4. Fleksibilitas Kecepatan Pembelajaran

   E-learning dapat disesuaikan dengan kecepatan belajar masing-masing siswa. Siswa mengatur sendiri kecepatan pelajaran yang diikuti. Apabila belum mengerti, dapat dipelajari modul tertentu dan mengulanginya nanti. Pelajar pun dapat memilih modul yang ingin dipelajari dan dapat melewati modul pelajaran yang dianggap tidak sesuai dan mengonsentrasikan diri ke bagian lain.

5. Standarisasi Pengajaran

   Pelajaran e-*learning* selalu memiliki kualitas yang sama setiap kali diakses dan tidak tergantung suasana hati pengajar.

6. Efektivitas Pengajaran

   E-learning yang didesain dengan instruktual design mutahir akan membuat pelajar lebih mengerti isi pelajaran. Penyampaian pelajaran *e-learning* dapat berupa simulasi, visualisasi dan kasus-kasus, menggunakan bentuk permainan dan teknologi animasi.

Bentuk-bentuk pelajaran tersebut akan membantu proses pembelajaran dan mempertahankan minat belajar.

7. Kecepatan Distribusi

   E-learning dapat cepat menjangkau seseorang yang berada di luar wilayah pusat. Tim desain pelatihan hanya perlu mempersiapkan bahan pelatihan secepatnya dan menginstal hasilnya di server pusat *e-learning*. Jadi, semua komputer yang tergabung ke server dapat langsung mengakses.

8. Ketersediaan *On-Demand*

   E-learning dapat sewaktu-waktu diakses dan dapat dianggap sebagai "buku saku" yang membantu pekerjaan setiap saat. Dengan menggunakan suatu program *e-learning*, seseorang dapat segera mempraktikkan pelajaran dan menyelesaikan tugas dengan baik.

2.5 Keterbatasan dari *E-learning*

1. Budaya

   Penggunaan *e-learning* menuntut budaya *self learning*, dimana seseorang memotivasi diri sendiri agar mau belajar. Sehingga ada beberapa orang merasa tidak nyaman mengikuti pelatihan melalui komputer. Untuk itu, pada saat ingin mendesain *e-learning*, terlebih dahulu harus melihat budaya dan kebiasaan penggunaan teknologi pelajar. Apabila pengguna akses tidak terbiasa

menggunakan komputer, implementasi *e- learning* akan memakan waktu lebih lama.

2. Investasi

   Walaupun *e-learning* menghemat banyak biaya, tetapi harus mengeluarkan investasi awal yang cukup besar untuk mulai mengimplementasikan *e-learning*. Investasi dapat berupa biaya desain dan pembuatan program *Learning Management System*, paket pembelajaran dan biaya-biaya lain seperti promosi dan *Change Managemet System*. Apabila insfrastruktur yang dimiliki belum memadai, organisasi harus mengeluarkan sejumlah dana untuk membeli komputer, jaringan, server, dan lain sebagainya.

3. Teknologi

   Teknologi yang digunakan beragam, ada kemungkinan teknologi tersebut tidak sejalan dengan yang ada dan terjadi konflik teknologi sehingga *e-learning* tidak berjalan baik. Oleh karena itu, kompabilitas teknologi yang digunakan harus diteliti sebelum memutuskan menggunakan suatu paket *e-learning*.

4. Infrastruktur

   Internet belum menjangkau semua kota di Indonesia. Layanan broadband baru ada di kota-kota besar. Akibatnya, belum semua orang atau wilayah belum dapat merasakan *e-learning* dengan *internet*.

5. Materi

   Walaupun *e-learning* menawarkan berbagai fungsi, ada beberapa materi yang tidak dapat diajarkan melalui *e-learning*.

2.6 Pengertian Kimia

Kimia (dari bahasa Arab: **كيمياء**, transliterasi: kimiya = perubahan benda / zat atau bahasa Yunani: χημεία, transliterasi: khemeia) adalah ilmu yang mempelajari mengenai komposisi, struktur, dan sifat zat atau materi dari skala atom hingga molekul serta perubahan atau transformasi serta interaksi mereka untuk membentuk materi yang ditemukan sehari-hari. Kimia juga mempelajari pemahaman sifat dan interaksi atom individu dengan tujuan untuk menerapkan pengetahuan tersebut pada tingkat makroskopik. Menurut kimia modern, sifat fisik materi umumnya ditentukan oleh struktur pada tingkat atom yang pada gilirannya ditentukan oleh gaya antaratom dan ikatan kimia.

Kimia sering disebut sebagai "ilmu pusat" karena menghubungkan berbagai ilmu lain, seperti fisika, ilmu bahan, nanoteknologi, biologi, farmasi, kedokteran, bioinformatika, dan geologi. Koneksi ini timbul melalui berbagai subdisiplin yang memanfaatkan konsep-konsep dari berbagai disiplin ilmu. Sebagai contoh, kimia fisik melibatkan penerapan prinsip-prinsip fisika terhadap materi pada tingkat atom dan molekul.

Kimia berhubungan dengan interaksi materi yang dapat melibatkan dua zat atau antara materi dan energi, terutama dalam hubungannya dengan hukum pertama termodinamika. Kimia tradisional melibatkan interaksi antara zat kimia

dalam reaksi kimia, yang mengubah satu atau lebih zat menjadi satu atau lebih zat lain. Kadang reaksi ini digerakkan oleh pertimbangan entalpi, seperti ketika dua zat berentalpi tinggi seperti hidrogen dan oksigen elemental bereaksi membentuk air, zat dengan entalpi lebih rendah. Reaksi kimia dapat difasilitasi dengan suatu katalis, yang umumnya merupakan zat kimia lain yang terlibat dalam media reaksi tapi tidak dikonsumsi (contohnya adalah asam sulfat yang mengkatalisasi elektrolisis air) atau fenomena immaterial (seperti radiasi elektromagnet dalam reaksi fotokimia). Kimia tradisional juga menangani analisis zat kimia, baik di dalam maupun di luar suatu reaksi, seperti dalam spektroskopi.

Semua materi normal terdiri dari atom atau komponen-komponen subatom yang membentuk atom; proton, elektron, dan neutron. Atom dapat dikombinasikan untuk menghasilkan bentuk materi yang lebih kompleks seperti ion, molekul, atau kristal. Struktur dunia yang kita jalani sehari-hari dan sifat materi yang berinteraksi dengan kita ditentukan oleh sifat zat-zat kimia dan interaksi antar mereka. Baja lebih keras dari besi karena atom-atomnya terikat dalam struktur kristal yang lebih kaku. Kayu terbakar atau mengalami oksidasi cepat karena ia dapat bereaksi secara spontan dengan oksigen pada suatu reaksi kimia jika berada di atas suatu suhu tertentu.

Zat cenderung diklasifikasikan berdasarkan energi, fase, atau komposisi kimianya. Materi dapat digolongkan dalam 4 fase, urutan dari yang memiliki energi paling rendah adalah padat, cair, gas, dan plasma. Dari keempat jenis fase ini, fase plasma hanya dapat ditemui di luar angkasa yang berupa bintang, karena kebutuhan energinya yang teramat besar. Zat padat memiliki struktur tetap pada

suhu kamar yang dapat melawan gravitasi atau gaya lemah lain yang mencoba mengubahnya. Zat cair memiliki ikatan yang terbatas, tanpa struktur, dan akan mengalir bersama gravitasi. Gas tidak memiliki ikatan dan bertindak sebagai partikel bebas. Sementara itu, plasma hanya terdiri dari ion-ion yang bergerak bebas; pasokan energi yang berlebih mencegah ion-ion ini bersatu menjadi partikel unsur. Satu cara untuk membedakan ketiga fase pertama adalah dengan volume dan bentuknya: kasarnya, zat padat memeliki volume dan bentuk yang tetap, zat cair memiliki volume tetap tapi tanpa bentuk yang tetap, sedangkan gas tidak memiliki baik volume ataupun bentuk yang tetap. Air yang dipanaskan akan berubah fase menjadi uap air. Air (H2O) berbentuk cairan dalam suhu kamar karena molekul-molekulnya terikat oleh gaya antarmolekul yang disebut ikatan Hidrogen. Di sisi lain, hidrogen sulfida (H2S) berbentuk gas pada suhu kamar dan tekanan standar, karena molekul-molekulnya terikat dengan interaksi dwikutub (dipole) yang lebih lemah. Ikatan hidrogen pada air memiliki cukup energi untuk mempertahankan molekul air untuk tidak terpisah satu sama lain, tapi tidak untuk mengalir, yang menjadikannya berwujud cairan dalam suhu antara 0 °C sampai 100 °C pada permukaan laut. Menurunkan suhu atau energi lebih lanjut mengizinkan organisasi bentuk yang lebih erat, menghasilkan suatu zat padat, dan melepaskan energi. Peningkatan energi akan mencairkan es walaupun suhu tidak akan berubah sampai semua es cair. Peningkatan suhu air pada gilirannya akan menyebabkannya mendidih (lihat panas penguapan) sewaktu terdapat cukup energi untuk mengatasi gaya tarik antarmolekul dan selanjutnya memungkinkan molekul untuk bergerak menjauhi satu sama lain.

Ilmuwan yang mempelajari kimia sering disebut kimiawan. Sebagian besar kimiawan melakukan spesialisasi dalam satu atau lebih subdisiplin. Kimia yang diajarkan pada sekolah menengah sering disebut "kimia umum" dan ditujukan sebagai pengantar terhadap banyak konsep-konsep dasar dan untuk memberikan pelajar alat untuk melanjutkan ke subjek lanjutannya. Banyak konsep yang dipresentasikan pada tingkat ini sering dianggap tak lengkap dan tidak akurat secara teknis. Walaupun demikian, hal tersebut merupakan alat yang luar biasa. Kimiawan secara reguler menggunakan alat dan penjelasan yang sederhana dan elegan ini dalam karya mereka, karena terbukti mampu secara akurat membuat model reaktivitas kimia yang sangat bervariasi. Adapun Kimia menurut Brady (1994 : 3), ilmu kimia merupakan ilmu mengenal bahan kimia. Bahan kimia bukanlah zat abstrak yang perlu ditakuti oleh manusia biasa. Bahan ini mencakup benda yang ada di sekitar kita. Selanjutnya ilmu kimia dapat didefinisikan sebagai ilmu murni yang mempelajari bahan-bahan yang ada di alam semesta, interaksi diantaranya dan perubahan energy yang berhubungan atau disebabkan oleh adanya perubahan-perubahan alam. Sedangkan menurut Irfan Anshory (2000: 3), ilmu kimia adalah cabang ilmu pengetahuan alam yang mempelajari struktur materi, sifat-sifat materi, perubahan suatu materi menjadi materi lain, serta energy yang menyertai perubahan materi.

2.6.1 Sejarah Kimia

Sejarah kimia dimulai lebih dari 4000 tahun yang lalu dimana bangsa Mesir mengawalidengan the art of synthetic "wet" chemistry. 1000 tahun SM, masyarakat purba telah menggunakan teknologi yang akan menjadi dasar

terbentuknya berbagai macam cabang ilmu kimia. Ekstrasi logam dari bijihnya, membuat keramik dan kaca, fermentasi bir dan anggur, membuat pewarna untuk kosmetik dan lukisan, mengekstraksi bahan kimia dari tumbuhan untuk obat-obatan dan parfum, membuat keju, pewarna, pakaian, membuat paduan logam seperti perunggu. Mereka tidak berusaha untuk memahami hakikat dan sifat materi yang mereka gunakan serta perubahannya, sehingga pada zaman tersebut ilmu kimia belum lahir. Tetapi dengan percobaan dan catatan hasilnya merupakan sebuah langkah menuju ilmu pengetahuan. Para ahli filsafat Yunani purba sudah mempunyai pemikiran bahwa materi tersusun dari partikel-partikel yang jauh lebih kecil yang tidak dapat dibagi-bagi lagi (atomos). Namun konsep tersebut hanyalah pemikiran yang tidak ditunjang oleh eksperimen, sehingga belum pantas disebut sebagai teori kimia. Ilmu kimia sebagai ilmu yang melibatkan kegiatan ilmiah dilahirkan oleh para ilmuwan muslim bangsa Arab dan Persia pada abad ke-8. Salah seorang bapak ilmu kimia yang terkemuka adalah Jabiri bn Hayyan (700-778), yang lebih dikenal di Eropa dengan nama Latinnya, Geber. Ilmu yang bari itu diberi nama al-kimiya (bahasa Arab yangberarti “perubahan materi”). Dari kata al-kimiya inilah segala bangsa di muka bumi ini meminjam istilah: alchemi (Latin), chemistry (Inggris), chimie (Perancis), chemie (Jerman), chimica (Italia) dan kimia (Indonesia). Sejarah kimia dapat dianggap dimulai dengan pembedaan kimia dengan alkimia oleh Robert Boyle (1627–1691) melalui karyanya The Sceptical Chymist (1661). Baik alkimia maupun kimia mempelajari sifat materi dan perubahan-perubahannya tapi, kebalikan dengan alkimiawan, kimiawan menerapkan metode ilmiah. Pada tahun 1789 terjadilah dua jenis revolusi besar di

Perancis yang mempunyai dampak bagi perkembangan sejarah dunia. Pertama, revolusi di bidang politik tatkala penjara Bastille diserbu rakyat dan hal ini mengawali tumbuhnya demokrasi di Eropa. Kedua, revolusi di bidang ilmu tatkala Antoine Laurent Lavoisier (1743-1794) menerbitkan bukunya, Traite Elementaire de Chimie, hal ini mengawali tumbuhnya kimia modern. Dalam bukunya Lavoisier mengembangkan hukum kekekalan massa. Penemuan unsur kimia memiliki sejarah yang panjang yang mencapai puncaknya dengan diciptakannya tabel periodik unsur kimia oleh Dmitri Mendeleyev pada tahun1869.

Kimia modern dimulai oleh kimiawan Perancis Antoine Laurent Lavoisier (1743-1794). Ia menemukan hukum kekekalan massa dalam reaksi kimia, dan mengungkap peran oksigen dalam pembakaran. Berdasarkan prinsip ini, kimia maju di arah yang benar. Sebenarnya oksigen ditemukan secara independen oleh dua kimiawan, kimiawan Inggris Joseph Priestley (1733-1804) dan kimiawan Swedia Carl Wilhelm Scheele (1742-1786), di penghujung abad ke-18. Jadi, hanya sekitar dua ratus tahun sebelum kimia modern lahir. Dengan demikian, kimia merupakan ilmu pengetahuan yang relatif muda bila dibandingkan dengan fisika dan matematika, keduanya telah berkembang beberapa ribu tahun.Lebih lanjut, sebelum abad ke-18, metalurgi dan farmasi sebenarnya didasarkan atas pengalaman saja dan bukan teori. Jadi, nampaknya tidak mungkin titik-titik awal ini yang kemudian berkembang menjadi kimia modern. Berdasarkan hal-hal ini dan sifat kimia modern yang terorganisir baik dan sistematik metodologinya, akar sebenarnya kimia modern mungkin dapat ditemui di filosofi Yunani kuno. Jalan dari filosofi Yunani kuno ke teori atom modern tidak selalu mulus. Di Yunani

kuno, ada perselisihan yang tajam antara teori atom dan penolakan keberadaan atom. Sebenarnya, teoriatom tetap tidak ortodoks dalam dunia kimia dan sains. Orang-orang terpelajar tidak tertarik padateori atom sampai abad ke-18. Di awal abad ke-19, kimiawan Inggris John Dalton (1766-1844) melahirkan ulang teori atom Yunani kuno. Bahkan setelah kelahirannya kembali ini, tidak semua ilmuwan menerima teori atom. Tidak sampai awal abad 20 teori atom akhirnya dibuktikan sebagai fakta, bukan hanya hipotesis. Hal ini dicapai dengan percobaan yang terampil oleh kimiawan Perancis Jean Baptiste Perrin (1870-1942). Sebagaimana dicatat sebelumnya, kimia adalah ilmu yang relatif muda. Akibatnya, banyak yang masih harus dikerjakan sebelum kimia dapat mengklaim untuk mempelajari materi, dan melalui pemahaman materi ini memahami alam ini.

2.7 Android

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

2.7.1 Sejarah Android

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

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

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

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

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

1. Android versi 1.1

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

2. Android versi 1.5 (Cupcake)

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

3. Android versi 1.6 (Donut)

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

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

4. Android versi 2.0/2.1 (Eclair)

Pada 3 Desember 2009 kembali diluncurkan ponsel Android dengan versi 2.0/2.1 (Eclair), perubahan yang dilakukan adalah pengoptimalan hardware, peningkatan *Google Maps* 3.1.2, perubahan UI dengan browser baru dan dukungan HTML5, daftar kontak yang baru, dukungan *flash* untuk kamera 3,2 MP, *digital Zoom*, dan Bluetooth 2.1. Untuk bergerak cepat dalam persaingan perangkat generasi berikut, Google melakukan investasi dengan mengadakan kompetisi aplikasi *Mobile* terbaik (*killer apps* - aplikasi unggulan). Kompetisi ini berhadiah $25,000 bagi setiap pengembang aplikasi terpilih. Kompetisi diadakan selama dua tahap yang tiap tahapnya dipilih 50 aplikasi terbaik.

Dengan semakin berkembangnya dan semakin bertambahnya jumlah handset Android, semakin banyak pihak ketiga yang berminat untuk menyalurkan aplikasi mereka kepada sistem operasi Android. Aplikasi terkenal yang diubah ke dalam sistem operasi Android adalah Shazam, Backgrounds, dan WeatherBug.

Sistem operasi Android dalam situs Internet juga dianggap penting untuk menciptakan aplikasi Android asli, contohnya oleh MySpace dan Facebook.

5. Android versi 2.2 (Froyo: Frozen Yoghurt)

Pada 20 Mei 2010, Android versi 2.2 (Froyo) diluncurkan. Perubahan – perubahan umumnya terhadap versi-versi sebelumnya antara lain dukungan

Adobe Flash 10.1, kecepatan kinerja dan aplikasi 2 sampai 5 kali lebih cepat, intergrasi V8 JavaScript *engine* yang dipakai Google Chrome yang mempercepat kemampuan *rendering* pada browser, pemasangan aplikasi dalam SD Card, kemampuan *WiFi Hotspot portabel*, dan kemampuan auto update dalam aplikasi Android Market.

6. Android versi 2.3 (Gingerbread)

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

7. Android versi 3.0 (Honeycomb)

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

8. Android versi 4.0 (Ice Cream Sandwich)

Versi dirilis pada 19 Oktober 2011. Smartphone yang pertama kali mengunakan OS Android ini adalah Samsung Galaxy Nexus. Secara teori semua perangkat seluler yang menggunakan versi Android sebelumnya, Gingerbread,

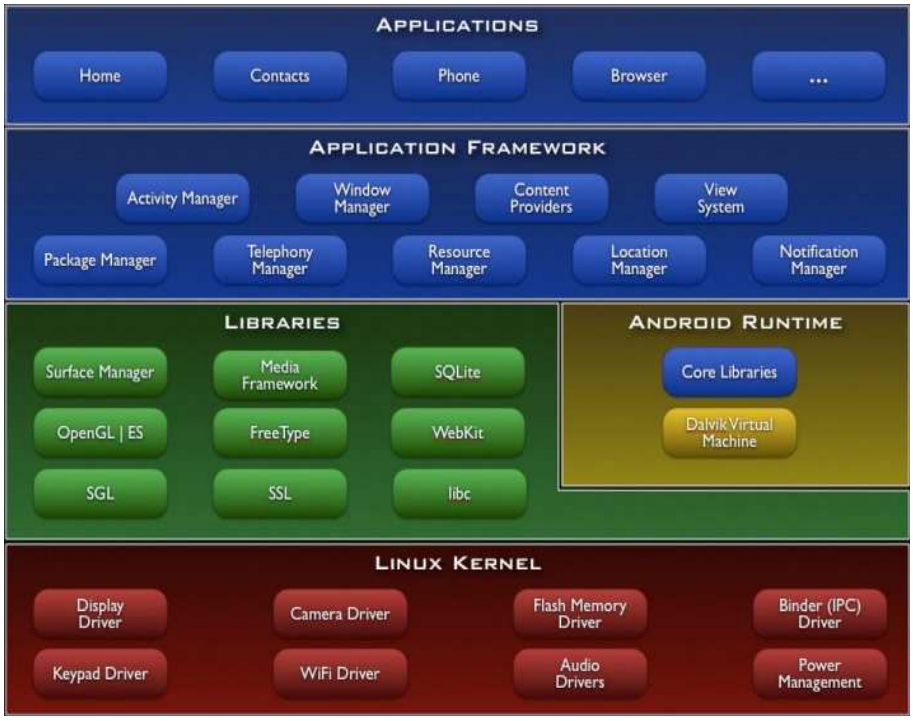
dapat di-update ke Android Ice Cream Sandwich. Namun sayangnnya sampai saat ini kebanyak smartphone yang menggunakan Android ICS merupakan smartphone kelas high-end yang dijual dengan harga cukup mahal. Mungkin karena alasan inilah distribusi versi Android satu ini tidak lebih dari 8% sampai pertangahn tahun 2012.

9. Android versi 4.1 (Jelly Bean)

Versi Android Jelly Bean adalah versi Android yang terbaru. Android Jelly Bean yang diluncurkan pada acara Google I/O lalu membawa sejumlah keunggulan dan fitur baru. Penambahan baru diantaranya meningkatkan input keyboard, desain baru fitur pencarian, UI yang baru dan pencarian melalui Voice Search yang lebih cepat. Tak ketinggalan Google Now juga menjadi bagian yang diperbarui. Google Now memberikan informasi yang tepat pada waktu yang tepat pula. Salah satu kemampuannya adalah dapat mengetahui informasi cuaca, lalu-lintas, ataupun hasil pertandingan olahraga. Sistem operasi Android Jelly Bean 4.1 muncul pertama kali dalam produk tablet Asus, yakni Google Nexus 7.

2.7.2 Anatomi Aplikasi Android

Dalam paket sistem operasi android tediri dari beberapa unsur seperti tampak pada gambar 2.1. Secara sederhana arsitektur android merupakan sebuah kernel Linux dan sekumpulan pustaka C / C++ dalam suatu *Framework* yang menyediakan dan mengatur alur proses aplikasi.



Gambar 2.1 *Detail* Anatomi *Android*

2.7.2.1 *Linux Kernel*

Android dibangun di atas kernel Linux 2.6. Namun secara keseluruhan android bukanlah linux, karena dalam android tidak terdapat paket standar yang dimiliki oleh linux lainnya. Linux merupakan sistem operasi terbuka yang handal dalam manajemen memori dan proses. Oleh karenanya pada android hanya terdapat beberapa servis yang diperlukan seperti keamanan, manajemen memori, manajemen proses, jaringan dan driver. Kernel linux menyediakan driver layar, kamera, *keypad, WiFi, Flash Memory, audio*, dan IPC (*Interprocess Communication*) untuk mengatur aplikasi dan lubang keamanan.

2.7.2.2 *Libraries*

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

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

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

2.7.2.3 *Android Runtime*

Pada android tertanam paket pustaka inti yang menyediakan sebagian besar fungsi android. Inilah yang membedakan Android dibandingkan dengan sistem operasi lain yang juga mengimplementasikan Linux. *Android Runtime* merupakan mesin virtual yang membuat aplikasi android menjadi lebih tangguh dengan paket pustaka yang telah ada. Dalam Android Runtime terdapat 2 bagian utama, diantaranya:

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

2.7.2.4. *Application Framework*

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

1. *Activity Manager*, berfungsi untuk mengontrol siklus hidup aplikasi dan menjaga keadaan *"Backstack"* untuk navigasi penggunaan.
2. *Content Providers*, berfungsi untuk merangkum data yang memungkinkan digunakan oleh aplikasi lainnya, seperti daftar nama.

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

2.7.2.5. *Application Layer*

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

1. *Activity Manager*, berfungsi untuk mengontrol siklus hidup aplikasi dan menjaga keadaan ”*Backstack“* untuk navigasi penggunaan.
2. *Content Providers*, berfungsi untuk merangkum data yang memungkinkan digunakan oleh aplikasi lainnya, seperti daftar nama.
3. *Resuource Manager*, untuk mengatur sumber daya yang ada dalam program. Serta menyediakan akses sumber daya diluar kode program, seperti karakter, grafik, dan *file layout*.
4. *Location Manager,* berfungsi untuk memberikan informasi detail mengenai lokasi perangkat android berada.

5. *Notification Manager,* mencakup berbagai macam peringatan seperti, pesan masuk, janji, dan lain sebagainya yang akan ditampilkan pada *status bar*.

2.7.3. Komponen Aplikasi *Android*

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

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

Sebaliknya, aplikasi memiliki komponen penting yang memungkinkan system untuk memanggil dan menjalankan ketika dibutuhkan.

2.7.3.1 *Activities*

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

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

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

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

dalam keadaan *paused* ini, untuk menunjang sumber daya bagi *activity* yang sedang aktif.

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

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

2.7.3.2 *Services*

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

2.7.3.3 *Intents*

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

2.7.3.4 *Broadcast Receiver*

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

2.7.3.5 *Content Providers*

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

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

2.7.4 Tipe Aplikasi Android

Terdapat tiga kategori aplikasi pada android :

1. *Foreground Activity*

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

2. *Background Service*

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

3. *Intermittent Activity*

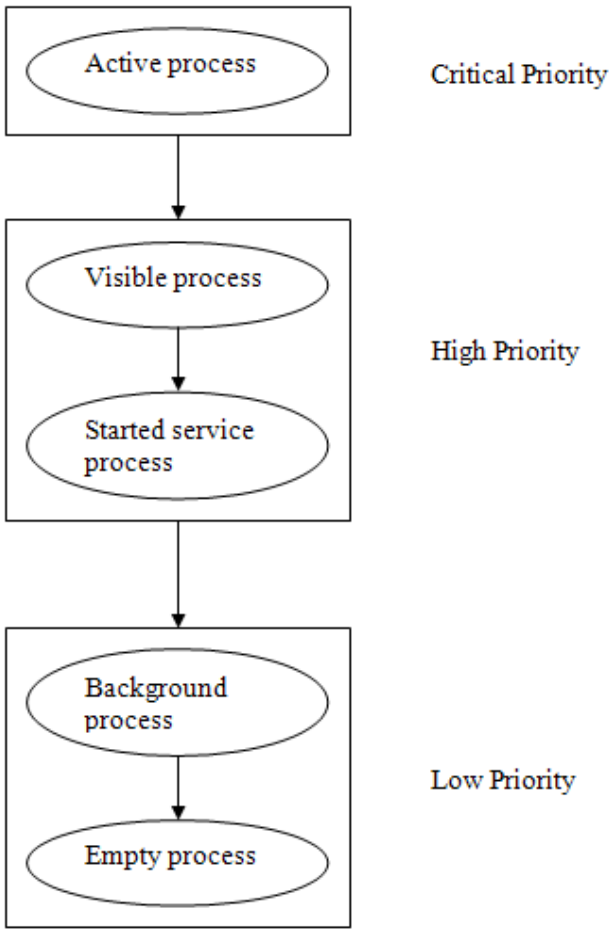
Aplikasi yang masih membutuhkan beberapa masukkan dari pengguna, namun sebagian sangat efektif jika dijalankan di *background* dan jika diperlukan akan memberi tahu pengguna tentang kondisi tertentu. Contohnya pemutar musik.

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

2.7.5 Siklus Hidup Android

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

Android menjalankan setiap aplikasi dalam proses secara terpisah, yang masing-masing memiliki mesin virtual pengolah sendiri, dengan ini melindungi penggunaan memori pada aplikasi. Selain itu juga android dapat mengontrol aplikasi mana yang layak menjadi prioritas utama. Karenanya android sangat sensitive dengan siklus hidup aplikasi dan komponen-komponennya. Perlu adanya penanganan terhadap setiap kondisi agar aplikasi menjadi stabil. Gambar 2.2 menunjukkan prioritas dari aplikasi.



Gambar 2.2 Prioritas Aplikasi Berdasarkan *Activity*

2.7.6 Kelebihan Android

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

1. Keterbukaan, Bebas pengembangan tanpa dikenakan biaya terhadap system 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 system 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.

2.8 *Eclipse*

Eclipse adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua *platform* (*platformindependent*).

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 bias digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, test perangkat lunak, pengembangan web, dan lain sebagainya.

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

2.8.1 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:

a. *Core platform*

b. OSGi

c. SWT (*Standard Widget Toolkit*)

d. JFace

e. *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[2]. 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.

2.8.2 Android SDK

Android SDK adalah *tools* API (*Application Programming Interface*) yang diperlukan untuk mulai mengembangkan aplikasi pada *platform* android menggunakan bahasa pemrograman Java. Android merupakan subset perangkat lunak untuk ponsel yang meliputi sistem operasi, *middleware* dan aplikasi kunci yang di *release* oleh Google. Saat ini disediakan Android SDK (*Software Development Kit*) sebagai alat bantu dan API untuk mulai mengembangkan aplikasi pada *platform* android menggunakan bahasa pemrograman Java. Sebagai *platform* aplikasi-netral, android member anda kesempatan unutk membuat aplikasi yang kita butuhkan yang bukan merupakan aplikasi bawaan Hadphone/*Smartphone*. Beberapa fitur-fitur android yang paling penting adalah :

a. *Framework* : aplikasi yang mendukung pengganti komponen dan reusable.
b. *Dalvik Virtual Machine* dioptimalkan untuk perangkat *mobile*
c. Integrated *Browser* verdasarkan engine *open source* WebKit.

d. Grafis yang dioptimalkan dan didukung oleh *libraries* grafis 2D, grafis 3D berdasarkan spesifikasi opengl ES 1,0 (Opsional Ekselerasi *hardware*)

e. SQLite untuk penyimpanan data.

f. Media *Support* yang mendukung audio, video, dan gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PING, GIF), GSM Telephony (tergantung *Hardware*)

g. *Bluetooth*, EDGE, 3G, dan WiFi (tergantung *hardware*)

h. Kamera, GPS, Kompas, dan *Accelerometer* (tergantung *hardware*)

i. Lingkungan *Development* yang lengkap dan termasuk pernagkat *emulator*, *tools* untuk *debugging*, profil dan kinerja memori, dan *plugin* untuk IDE Eclipse.

Untuk source SDK Android ini dapat dilihat dan didownload langsung di situs resmi pengembang SDK Android di http://www.developer.android.com.

2.9 *MySQL*

MySQL adalah salah satu jenis database *server* yang sangat terkenal, kepopulerannya disebabkan MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses databasenya. Selain itu , ia bersifat *Open source* (Anda tidak perlu membayar untuk menggunakannya) pada pelbagai *platform* (kecuali untuk jenis enterprise, yang bersifat komersial). Perangkat lunak MySQL sendiri bisa di *download* dari http://www.mysql.com MySQL termasuk jenis RDBMS (*Relational Database Management*). Itulah sebabnya, istilah seperti tabel, baris, dan kolom digunakan pada MySQL. Pada MySQL, sebuah database mengandung

satu atau sejumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau beberapa kolom.

2.10 *Unified Modeling Language* (UML)

UML menurut Booch (1999: 14) dalam Jurnal Gintoro, Andreyus, Emilia dan Richard William (2010 , Hal : B-30) adalah sebuah bahasa yang telah menjadi standar dalam industri untuk menvisualisasi, menspesifikasi, merancang dan mendokumentasi sistem piranti lunak.

Unified Modelling Language (UML) menawarkan sebuah standar untuk merancang model sebuah sistem. Tujuan UML adalah :

1. Memberikan model yang siap pakai, bahasa pemodelan visual yang ekspresif untuk mengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.
2. Memberikan bahasa pemodelan yang bebas dari berbagai bahasa pemrograman dan proses rekayasa.
3. Menyatukan praktik-praktik terbaik yang terdapat dalam pemodelan.

UML menyediakan beberapa *notasi dan artifact standar* yang bisa digunakan sebagai alat komunikasi bagi para pelaku dalam proses analisis dan desain. Artifact didalam UML didefinisikan sebagai informasi dalam bentuk yang digunakan atau dihasilkan dalam proses pengembangan perangkat. Contohnya adalah source code yang dihasilkan oleh proses pemrograman.

2.10.1 *Use Case Diagram*

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. *Use Case* dilakukan oleh satu *actor* yang digambarkan dengan simbol orang yang dihubungkan dengan garis yang menunjukan hubungan komunikasi. Setiap *Use Case* harus diberi nama yang menyatakan apa hal yang dicapai dari hasil interaksi dengan *Actor.* Nama *Use Case* boleh terdiri dari beberapa kata dan tidak boleh ada *Use Case* yang memiliki nama yang sama.

2.10.2 *Class Diagram*

Diagram kelas menggambarkan tipe-tipe objek dalam *system* dan berbagai jenis hubungan atau relasi statis yang ada diantara mereka. Diagram ini memberikan gambaran umum dari sistem. Seperti tipe-tipe dari objek dengan menunjukan kelasnya dan *relationship* yang diantara mereka, serat penjelasan detail tiap-tiap kelas ke dalam model suatu *system. Class diagram* bersifat *static* (tidak berubah) yang akan menunjukan apa itu interaksi tapi tidak menjelaskan apa yang terjadi ketika mereka melakukan interaksi.

2.10.3 *Activity Diagram*

Activity diagram memodelkan alur kerja *(work flow).* Sebuah proses bisnis dan urutan aktifitas dalam suatu proses. Diagram ini sangat mirip dengan sebuah *flowchart* karena kita dapat memodelkan sebuah alur kerja dari aktifitas keaktifitas lainnya atau dari suatu aktifitas kedalam keadaan sesaat (*state).*

2.10.4 *Sequence Diagram*

Diagram sequence merupakan gambaran interaksi antar objek di dalam dan di sekitar sistem berupa *massage* yang digambarkan terhadap waktu. Diagram ini secara khusus berasosiasi dengan *use Case. Diagram sequence* juga digunakan untuk menggambarkan skenario atau rangkain langkah-langkah apa yang seharusnya terjadi sebagai respons dari sebuah *event* untuk menghasilkan sesuatu didalam *Use case* sebagai *output.* Untuk *massage* digambarkan sebagai garis berpanah dari satu objek ke objek lainnya. Pada fase desain berikutnya, *message* dipetakan menjadi operasi/metode dari *class. Activation bar* menunjukan lamanya eksekusi sebuat proses, biasanya diawali dengan sebuah *message*. Berikut ini adalah contoh dari *diagram sequence.*

2.10.5 *Deployment Diagram*

Diagram *deployment* atau *deployment diagram* menunjukan konfigurasi komponen dalam proses eksekusi aplikasi. Diagram *deployment* juga dapat digumakan untuk memodelkan hal-hal berikut

1. Sistem tambahan (*embedded system*) yang menggambarkan rancangan *device*, *node,* dan *hardware.*
2. Sistem *client*/*server*
3. Sistem terdistribusi murni
4. Rekayasa ulang aplikasi

(Sumber :Buku Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek). Pengarang Rosa A.S – M.Shalahuddin Tahun 2011).