BAB II

LANDASAN TEORI

2.1 Aplikasi

Aplikasi menurut Jogiyanto *(2001:12)* adalah penggunaan dalam suatu komputer,instruksi *(instructiom)* atau pernyataan *(statement)* yang disusun sedemikian rupa sehingga komputer dapat memproses *input* menjadi *output*. Menurut *Kamus Besar Bahasa Indonesia (2001 : 52)*, "Aplikasi adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atauketentuan bahasa pemrograman tertentu". Aplikasi adalah suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari pengguna. Aplikasi merupakan rangkaian kegiatan atau perintah untuk dieksekusi oleh komputer. Program merupakan kumpulan *instruction set* yang akan dijalankan oleh pemroses, yaitu berupa software. Bagaimana sebuah sistem komputer berpikir diatur oleh program ini.

Program inilah yang mengendalikan semua aktifitas yang ada pada pemroses. Program berisi konstruksi logika yang dibuat oleh manusia, dan sudah diterjemahkan ke dalam bahasa mesin sesuai dengan format yang ada pada *instructionset*. Program aplikasi merupakan program siap pakai. Program yang direka untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain. Contoh-contoh aplikasiialah program pemproses kata dan *Web Browser*. Aplikasi akan menggunakan sistemoperasi (OS) komputer dan aplikasi yang lainnya yang mendukung.Istilah ini mulai

perlahan masuk ke dalam istilah Teknologi Informasi semenjak tahun 1993, yang biasanya juga disingkat dengan app. Secara historis, aplikasi adalah software yang dikembangkan oleh sebuah perusahaan. App adalah *sofware* yang dibeli perusahaan dari tempat pembuatnya. Industri PC tampaknya menciptakan istilah ini untuk merefleksikan medan pertempuran persaingan yang baru, yang paralel dengan yang terjadi antar sistem operasi yang dimunculkan.

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

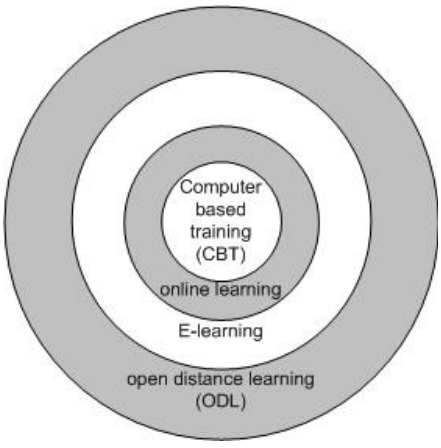
2.3 Konsep E-learning

Istilah *E-learning* sering digunakan semua kegiatan pendidikan yang menggunakan media komputer 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 dengan 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. Ruang lingkup bentuk bentuk diatas digambarkan pada Gambar II-1.



Gambar 2.1 Ruang Lingkup e-learning

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. SynchronousTraining

SynchronousTraining 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 Syst*em, 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. Fisika

Fisika (bahasa Yunani: φυσικός (fysikós), "alamiah", dan φύσις (fýsis), "alam") adalah sains atau ilmu tentang alam dalam makna yang terluas. Fisika mempelajari gejala alam yang tidak hidup atau materi dalam lingkup ruang dan waktu. Para fisikawan atau ahli fisika mempelajari perilaku dan sifat materi dalam bidang yang sangat beragam, mulai dari partikel submikroskopis yang membentuk segala materi (fisika partikel) hingga perilaku materi alam semesta sebagai satu kesatuan kosmos.

Beberapa sifat yang dipelajari dalam fisika merupakan sifat yang ada dalam semua sistem materi yang ada, seperti hukum kekekalan energi. Sifat semacam ini sering disebut sebagai hukum fisika. Fisika sering disebut sebagai "ilmu paling mendasar", karena setiap ilmu alam lainnya (biologi, kimia, geologi, dan lain-lain) mempelajari jenis sistem materi tertentu yang mematuhi hukum fisika. Misalnya, kimia adalah ilmu tentang molekul dan zat kimia yang dibentuknya. Sifat suatu zat kimia ditentukan oleh sifat molekul yang

membentuknya, yang dapat dijelaskan oleh ilmu fisika seperti mekanika kuantum, termodinamika, dan elektromagnetika.

Fisika juga berkaitan erat dengan matematika. Teori fisika banyak dinyatakan dalam notasi matematis, dan matematika yang digunakan biasanya lebih rumit daripada matematika yang digunakan dalam bidang sains lainnya. Perbedaan antara fisika dan matematika adalah: fisika berkaitan dengan pemerian dunia material, sedangkan matematika berkaitan dengan pola-pola abstrak yang tak selalu berhubungan dengan dunia material. Namun, perbedaan ini tidak selalu tampak jelas. Ada wilayah luas penelitan yang beririsan antara fisika dan matematika, yakni fisika matematis, yang mengembangkan struktur matematis bagi teori-teori fisika.

2.6.1 Sejarah Fisika

Sejak zaman purbakala, orang telah mencoba untuk mengerti sifat dari benda: mengapa objek yang tidak ditopang jatuh ke tanah, mengapa material yang berbeda memiliki properti yang berbeda, dan seterusnya. Lainnya adalah sifat dari jagad raya, seperti bentukBumi dan sifat dari objek celestial seperti Matahari dan Bulan.

Beberapa teori diusulkan dan banyak yang salah. Teori tersebut banyak tergantung dari istilah filosofi, dan tidak pernah dipastikan oleh eksperimen sistematik seperti yang populer sekarang ini. Ada pengecualian dan anakronisme: contohnya, pemikir Yunani Archimedes menurunkan banyak deskripsi kuantitatif yang benar dari mekanik dan hidrostatik.

Pada awal abad 17, Galileo membuka penggunaan eksperimen untuk memastikan kebenaran teori fisika, yang merupakan kunci darimetode sains. Galileo memformulasikan dan berhasil mengetes beberapa hasil dari dinamika mekanik, terutama Hukum Inert.

Pada 1687, Isaac Newton menerbitkan *Philosophiæ Naturalis Principia Mathematica* ("prinsip matematika dari filsafat alam", dikenal sebagai *Principia*), memberikan penjelasan yang jelas dan teori fisika yang sukses.

Hukum gerak Newton, yang merupakan sumber mekanika klasik; dan Hukum Gravitasi Newton, yang menjelaskan gaya dasar gravitasi. Kedua teori ini cocok dalam eksperimen. *Principia* juga memuat beberapa teori dinamika fluida.

Mekanika klasik dikembangkan besar-besaran oleh Joseph-Louis de Lagrange, William Rowan Hamilton, dan lainnya, yang menciptakan formula, prinsip, dan hasil baru. Hukum Gravitasi memulai bidang astrofisika, yang menggambarkan fenomena astronomimenggunakan teori fisika.

Dari sejak abad 18 dan seterusnya, termodinamika dikembangkan oleh Robert Boyle, Thomas Young, dan banyak lainnya. Pada 1733,Daniel Bernoulli menggunakan argumen statistika dalam mekanika klasik untuk menurunkan hasil termodinamika, memulai bidangmekanika statistik.

Pada 1798, Benjamin Thompson mempertunjukkan konversi kerja mekanika ke dalam panas, dan pada 1847 James Joule menyatakan hukum konservasi energi, dalam bentuk panasa juga dalam energi mekanika.

Sifat listrik dan magnetisme dipelajari oleh Michael Faraday, George Simon Ohm, dan lainnya. Pada 1855, James Clerk Maxwell menyatukan kedua fenomena menjadi satu teori elektromagnetisme, dijelaskan oleh persamaan Maxwell. Perkiraan dari teori ini adalahcahaya adalah gelombang elektromagnetik.

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 peranti bergerak

2.7.1. Sejarah Android

Android Inc. berdiri secara independen yang didirikan pada tahun 2003 oleh Andy Rubin, Rich Miner, Chris White, dan Nicks Sears. Android Inc. bermarkas di Palo Alto, California. Perusahaan berusaha untuk mengembangkan sebuah *operating system mobile* yang lebih canggih dari operating system mobile kala itu. Kemudian pada Agustus 2005 *Google* mngajukan penawaran untuk mengakuisisi perusahaan tersebut

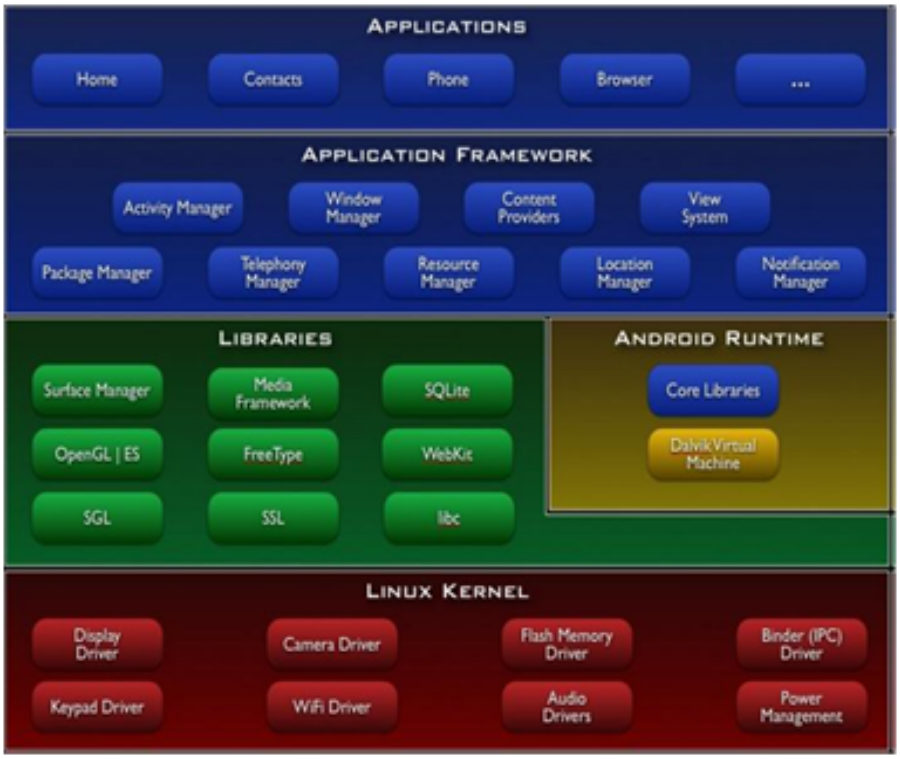
Dimulai pada tahun 2005, *Android Inc*. dibawah naungan *Google Inc*. berusaha membuat sebuah operating system mobile baru. Sejak sat itulah mulai beredar rumor bahwa Google akan melakukan ekspansi bisnis ke industri seluler. Akhirnya pada bulan September 2007 Google mngajukan hak paten atas produknya yang kemudian dinamai dengan *Nexus One.*

Kemudian pada akhir tahun 2008, dibentuk sebuah tim kerja sama yang dinamai *Open Handset Alliance (OHA). OHA* ini terdiri dari beberapa produsen perangkat telekomunikasi ternama dunia, antara lain ASUS, Toshiba, Sony Ericsson (sekarang Sony), Garmin, Vodafone, dan Softbank. OHA bekerja sama untuk mengembangkan sebuah kernel Linux yang akan dijadkan sebuah program untuk perangkat seluler. Hingga akhirnya OHA berhasil dan mengumumkan produk *operating system mobile* yang diberi nama Android. Ponsel yang mendapat kehormatan untuk mencoba pertama kali sistem operasi Anroid adalah *HTC Dream. HTC Dream* dirilis pada bulan Oktober tahun 2008. Sejak saat itu banyak perusahaan perangkat seluler di dunia ikut menggunakan Android sebagai *operating system* ponsel mereka.

Tak disangka-sangka ternyata operating system *Android* dengan cepat menjadi *operating system* yang digemari masyarakat. Hal ini tidak terlepas dari berbagai kelebihan *Android* yang diataranya mengusung kemampuan *multitasking*

2.7.2. Arsitektur *Android*

Google menggambarkan *Android* seperti sebuah tumpukan *software*. Setiap lapisan dari tumpukan ini terdiri dari beberapa program yang mendukung fungsi-fungsi spesifik dari sistem operasi. Inilah susunan Android dilihat dari lapisan dasar hingga lapisan paling atas!



Gambar 2.2 Arsitektur Android

Di lapisan teratas ada aplikasi itu sendiri. Di lapisan inilah kamu akan menemukan fungsi-fungsi dasar smartphone seperti menelepon dan mengirim pesan singkat, menjalankan *web browser,* mengakses daftar kontak, dan lain-lain. Inilah lapisan yang akan sering diakses oleh para pengguna. Mereka mengakses fungsi-fungsi dasar tersebut melalui *user interface.*

2.7.3. Fitur *Android*

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 diperlukan untuk mulai mengembangkan aplikasi pada platform *Android* menggunakan bahasa pemrograman *Java.*

1. *Application framework* mendukung pemakaian komponen-komponen yang mudah diganti dan digunakan kembali
2. *Dalvik virtual machine* dioptimalkan untuk perangkat mobile
3. *Integrated browser* dirancang berdasarkanengine *browser open source WebKit*
4. *Optimized graphics* didukung oleh *custom 2D graphics library; 3D graphics* berdasarkan OpenGL ES 1.0 *specification* (dukungan *hardware acceleration optional*)
5. *SQLite* untuk penyimpanan database terstruktur
6. Media support : unuk mendukung berbagai audio, *video*, dan *still image formats (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF)*
7. *GSM Telephony, Bluetooth, EDGE, 3G, and WiFi (hardware dependant)*
8. Kamera, GPS, *compass, and accelerometer (hardware dependant)*
9. *Rich development environment* meliputi *device emulator, tools* untuk *debugging, memory* dan performance profiling, serta plugin untuk *Eclipse IDE.*

2.7.4 Kelebihan Dan Kekurangan *OS Android*

Tentunya setiap OS memiliki kelebihan dan kekurangan masing-masing. Karena tidak ada yang *software* yang berjalan 100% sempurna.

2.7.4.1. Kelebihan OS *Android*

Karena *Android* sendiri adalah OS yang berbasis *Linux* jadi dapat dengan mudah dikembangkan oleh siapa saja

1. Akses yang mudah ke *Android App Market*. Di Android App Market bisa *download game* dan aplikasi yang mendukung *Android.* Fasilitas ini mirip seperti *Apple Store* di *Apple iOS*
2. Mudah dijangkau masyarakat. Tidak seperti *iOS* yang khusus pada *Apple Phone* saja, *Android* terdapat di banyak *platfrom-platfrom ponsel*. Mulai dari *Nokia, Samsung, SE,* sampai *HTC.*
3. Fasilitas penuh *USB*. Kamu bisa mengganti baterai, *mass storage, disk drive,* dan *USB tethering.*
4. *Easy Notification*. Kamu tidak akan terlewat satupun SMS, *Miss Call,* dan *Email,* bahkan artikel terbaru dari *RSS Reader* yang masuk.
5. Mendukung Semua Layanan *Google*. Karena diciptakan oleh *google, Android* sangat mendukung layanan-layanan *Google.* Mulai dari *Gmail, Google+, Google Map, Google Talk*, sampai *Google reader.*
6. *Install ROM* modifikasi. Kita kadang mendapati *ROM* yang tidak resmi. Maksudnya adalah versi yang telah rilis tidak sesuai dengan spesifikasi ponsel kita, jalan terakhir kita adalah modifikasi. Jangan khawatir ada banyak *custom ROM* yang bisa Anda pakai di ponsel *Android,* dan dijamin tidak akan membahayakan perangkat anda

2.7.4.2. Kelemahan OS *Android*

Selain Kelebihan, Android juga punya beberapa kelemahan diantaranya :

1. Selalu *Online.* *Android* sangat membutuhkan *internet* untuk menunjang kinerjanya, setidaknya harus ada koneksi dengan jaringan GPRS saja di tempat kamu.
2. Terkadang Perusahaan perangkat lambat dalam mengeluarkan versi resmi dari *Android* milik anda. Meskipun kadang tidak ada perbedaan mencolok dalam hal *UI(User Interface).*
3. *Android Market App* terkadang kurang mendapat dari pengelolanya. Jadi, kadang masih terdapat *malware.*
4. Sebagai penyedia layanan langsung, terkadang pengguna sangat sulit sekali terhubung dengan pihak *Google.*
5. Karena aplikasinya yang mudah dan grais, terkadang terdapat iklan di atas atau bawah tampilan aplikasi. Tetapi tidak begitu menggangu.
6. Boros Baterai**.** Dikarenakan *Android* menjalankan banyak program di bawahnya, *Android* menjadi lebih boros daripada *OS* lainnya.

2.7.5. Perkembangan OS *Android* dari Masa ke Masa

mengenai sejarah *OS Android* yang membahas mengenai latar belakang terciptanya sebuah sistem operasi canggih berbasiskan pada open *source Linux*. Sejak awal diakuisisinya *Android Inc*. oleh *Google* telah tercipta beberapa versi *Android* yang diberinama secara abjad dari versi pertama yang dinamai *Android*

*be*ta, kemudian versi 1.5 mulai dinami dengan nama makanan pencuci mulut (*dessert*) yaitu *Cupcake* hingga yang terbaru versi 5.0 saat ini yaitu *Jelly Bean.*

2.7.5.1. Android Beta

Pertama kali dirilis pada 5 November 2007, kemudian pada 12 November 2007 *Software Development Kit (SDK)* dirilis oleh *Google.*

2.7.5.2. *Android 1.0 (Astro)*

Pertama kali dirilis pada 23 Spetember 2008. Sebenarnya *Android* versi pertama ini akan dinamai dengan nama *"Astro"* tapi karena alasan hak cipta dan *trademark* nama*"Astro"* tidak jadi disematkan pada versi pertama dari *OS Android* ini. *HTC Dream* adalah ponsel pertama yang menggunakan OS *a*ni.

Pada *android* versi pertama ini sudah mulai diperkenalkan *Android Market,* tempat untuk mendownload berbagai macam aplikasi dan update untuk *Android*. Kemudian terdapat fitur *syncronisasi* dengan berbagai layanan *Google* lainnya seperti *Google Contact, Calender, Maps, Talk,* dan lainnya.

2.7.5.3. *Android 1.1 (Bender)*

Pertama kali dirilis pada 9 Februari 2009. Versi *Android* kedua ini juga mengalami masalah penamaan yang sama dengan versi pertamanya. Pada awalnya *Android* ini akan diberi nama *"Bender"* akan tetapi karena alasan melanggar *trademark,* nama *"Bender"* tidak jadi disematkan pada versi *Android* ini. Awalnya versi *OS Android* ini dirilis untuk perangkat *T-Mobile G1* saja. Versi ini

merupakan *update* untuk memperbaiki beberapa bugs, mengganti *API* dan menambahkan beberapa fitur.

2.7.5.4. *Android 1.5 Cupcake*

Pertama kali dirilis pada 30 April 2009. Nah, mulai versi *Androi*d ini penamaan menggunakan nama makan pencuci mulut *(dessert)* mulai digunakan, karena ini merupakan versi yang ketiga maka penamaan diawali dengan huruf "C" dan jadilah *"Cupcake"* menjadi nama resmi dari versi *OS Android* ketiga ini. *OS* ini berbasiskan pada *kernel Linux* 2.6.27 dan menambahkan beberapa update serta UI baru dari versi *Android* sebelumnya. Mulai terdapat *"widget"* yang dapat dibesar kecilkan. Kemudian ditambah kemampuan untuk meng-upload *video* dan gambar ke *Youtube* dan *Picasa.*

2.7.5.5. *Android 1.6 Donut*

Dirilis pertama kali pada 15 September 2009. Terdapat peningkatan pada fitur pencarian dan *UI* yang lebih *user friendly*. Pada versi ini juga sudah mendukung teknologi *CDMA/EVDO*, 802.1x, VPNs. Kemudian support layar dengan resolusi *WVGA*..

2.7.5.6. *Android 2.0/2.1 Enclair*

Dirilis pertama kali pada 9 Desember 2009. Terjadi penambahan fitur untuk pengoptimalan *hardware*, peningkatan *Google Maps* 3.1.2, perubahan UI dengan *browser* baru dan dukungan *HTML*5, daftar kontak yang baru, dukungan

flash untuk kamera 3,2 MP, *digital Zoom*, dan *Bluetooth* 2.1. Beberapa versi *update*nya antara *Android* v.2.0 kemudian v2.0.2 dan terakhir v.2.1.

2.7.5.7. *Android 2.2 Frozen Yoghurt (Froyo)*

Dirilis pertamakali pada 20 Mei 2010 pada smartphone Google Nexus One. Pada versi ini sudah support terhadap Adobe *Flash Player* 10.1. Peningkatan pada kecepatan membuka dan menutup aplikasi, serta penggunaan *SD Card* sebagai tempat penyimpanan aplikasi. Ketika *Android Froyo* hadir mulai muncul banyak diskusi yang membahas mengenai persaingan antara *Android* dengan *iOS* yang akan semakin ketat di masa yang akan datang. Beberapa versi *update* yang dirilis antara lain *Android* v.2.2.1 hingga v.2.2.3.

2.7.5.8. *Android 2.3 Gingerbread*

Pertama kali diperkenalkan pada 6 Desember 2010. Terjadi banyak peningkatan pada versi *Android* yang satu ini dibandingkan dengan versi sebelumnya. Dirancang untuk memaksimalakan kemampuan aplikasi dan game. Serta mulai digunakannya *Near Field Communication (NFC)*. Perbaikan terhadap dukungan layar resolusi *WXGA* dan diatasnya. Beberapa versi *update* yang dirilis antara lain v.2.3.3 hingga v.2.3.7. Sampai saat ini *Android Gingerbread* merupakan versi *Android* yang memiliki pengguna terbanyak dibandingkan dengan seri *Android* lainnya, yaitu mencapai 65% dari seluruh versi *Android* yang dirilis.

2.7.5.9. *Android 3.0/3.1 Honeycomb*

Pertama kali diperkenalkan pada 22 Februari 2011 dan *Motorola Xoom* adalah yang pertama kali menggunakannya. *Android* versi ini merupakan *OS* yang didesain khusus untuk pengoptimalan pengunaan pada tablet *PC.*

2.7.5.10. *Android 4.0 Ice Cream Sandwich*

Sampai tulisan ini ditulis ICS merupakan versi *Android* yang paling anyar. Pertama kali 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 ini. Berdasarkan versi-nya. Itulah perjalanan perkembangan *Operating System Android* dari masa ke masa. Meski belum genap 4 tahun OS *Android* hadir di pasar ponsel seluler namun saat ini *operating system* yang dinaungi oleh *Google* ini semakin mendekati posisi iOS sebagai penguasa sistem operasi *mobile* di dunia.

2.7.5.11. *Android versi 5.0 (Jelly Bean)*

Android Jelly Bean ini diperuntukkan untuk komputer tablet dan memungkinkan untuk digunakan pada sistem operasi PC atau Komputer.

Sehingga rumornya kemunculan *Android Jelly Bean* ini untuk menyaingi rilis terbaru *Windows* 8 yang juga akan segera dirilis. Karena kita ketahui bersama perbincangan versi Android sebelumnya yaitu Android Ice Cream Sandwhich pun masih hangat di telinga.

Kini Android Jelly Bean versi terbaru *Android* 5.0 ini mulai muncul dengan mengusung berbagai keunggulan yang ditawarkan diantaranya :

1. *Full chrome browser*, Fitur yang menawarkan berselancar di dunia maya dengan komputer tablet atau smartphone selayaknya berselancar dengan chrome di PC atau komputer karena pada *Android Jelly Bean Chrome Browser* ini sudah disempurnakan hingga sesuai dengan layanan di PC/Komputer dengan menggunakan *Smartphon*e yang tentunya sudah disesuaikan dengan layar Smartphone android anda.

2. *Dual boot*, Keunggulan yang luar biasa dengan fitur dual boot yang memungkinkan kita menggunakan 2 sistem operasi dalam satu perangkat komputer kita.

3. *File Manager* yang lebih baik, Pada Android versi sebelumnya terdapat file *manager* atau pengelola data yang bisa di download pada *Android market* pada *Android 5.0 Jelly B*ean ini pengguna dimudahkan dengan tersedianya langsung file *manager* saat menggunakan *Android Jelly Bean* tanpa harus *download* dan *Install file* manager pihak ketiga di *Android market*.

4. *Keyboard Virtual*, Terdapat *fitur keyboard virtual* yang lebih baik dan telah diperbarui. keyboard Virtual ini memudahkan proses penulisan dengan cara penggantian karakter dan hurud (QWERTY) ke angka dan simbol dengan cara menekan tombol sedikit agak lama sampai muncul perubahan keyboard virtual tersebut.

2.8. Java

Bahasa Java dikembangkanoleh *Sun Microsystem* tahun 1991 sebagai bagian dari suatu proyek penelitian untuk mengembangkan software bagi konsumer barang – barang elektronik seperti televise, *VCR, toaster* dan mesin – mesin lainnya yang dapat dibeli di swalayan. Tujuan penciptaan Java pada waktu itu adalah menjadi suatu program yang berukuran kecil, efisien, dan *portable* di segala jenis *hardware*. Tujuan yang sama ini membuat Java menjadi satu bahasa yang ideal untuk mendistribusikan program – program yang dapat dijalankan melalui *www* dan juga suatu bahasa pemrograman untuk segala tujuan untuk mengembangkan program – program yang dapat digunakan dengan mudah dan *portable* di berbagai *platform* yang berbeda.

Sekarang, *Sun* telah mengeluarkan berbagai program Java yang dapat digunakan seperti *Java API,* atau JDK atau *JAVA Developer Kit* . Selain itu, banyak juga program-program lain yang dapat digunakan untuk membuat program Java, seperti *Eclipse, NetBeans, JBuilder, JCreator, J++,* dan sebagainya.

Saat ini distribusi Java dan kelas pendukungnya dibagi dalam tiga bagian yang masing-masing memiliki konsentrasi tersendiri yaitu:

1. *Java 2 Standart Edition* (J2SE), untuk aplikasi *desktop*
2. *Java 2 Enterprise Edition* (J2EE), untuk aplikasi *server*
3. *Java 2 Micro Edition* (J2ME), untuk piranti dengan kemampuan terbatas.

2.9. Eclipse IDE

Eclipse adalah sebuat komunitas proyek open source yang difokuskan pada membangun sebuah platform pengembangan yang dapat diperluas, runtime dan kerangka aplikasi untuk membangun, menyebarkan dan mengelola perangkat lunak di seluruh daur hidup perangkat lunak.

Komunitas Eclipse memiliki lebih dari 60 proyek open source. Proyek-proyek ini secara konsep terbagi menjadi 7 kategori :

1. Enterprise Development
2. Embedded and Device Development
3. Rich Client Platform
4. Rich Internet Applications
5. Application Frameworks
6. Application Lifecycle Management (ALM)
7. Service Oriented Architecture (SOA)

Komunitas Eclipse juga didukung oleh ekosistem besar dan dinamis dari penyedia solusi teknologi informasi utama, start-up inovatif, universitas dan lembaga penelitian, dan individu yang memperluas, mendukung, dan melengkapi platform Eclipse.

2.9.1. 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.

2.9.2. 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.

2.9.3 ADT plugin for Eclipse

Android Development Tools (ADT) adalah plugin untuk Eclipse IDE yang dirancang khusus untuk memberikan *integrated environment* yang kuat untuk membuat aplikasi Android. ADT memberikan kemampuan kepada Eclipse untuk membuat projek baru Android secara cepat, membuat aplikasi *User Interface*, menambahkan komponen berdasarkan Android *Framework* API, melakukan *debugging* aplikasi yang di buat dengan menggunakan Android SDK *tools* dan bahkan melakukan distribusi aplikasi yang dibuat.

Pembuatan aplikasi android dengan Eclipse beserta ADT sangat dianjurkan karena merupakan cara tercepat untuk memulai membuat projek Android. Dengan disediakannya *project setup,* serta *tools* yang sudah terintegrasi di dalamnya, *custom XML editor,* dan *debugging* beserta *output* dalam *emulator* yang sudah

disediakan Android SDK, mempermudah para pengembang aplikasi Android dalam pembuatan aplikasinya.

2.10 UML

Unified Modeling Language (UML) adalah bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan, dan membangun sistem perangkat lunak. UML adalah himpunan struktur dan teknik untuk pemodelan desain program berorientasi objek (OOP) serta aplikasinya. UML adalah metodologi untuk mengembangkan sistem OOP dan sekelompok perangkat tool untuk mendukung pengembangan sistem tersebut. UML mulai diperkenalkan oleh Object 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.

2.10. 1. Sejarah UML

Sampai era tahun 1990 puluhan metodologi pemodelan berorientasi objek telah bermunculan di dunia. Diantaranya adalah: metodologi booch, metodologi coad, metodologi OOSE, metodologi OMT, metodologi shlaer-mellor, metodologi wirfs-brock, dan sebagainya. Masa itu terkenal dengan masa perang metodologi (method war) dalam pendesainan berorientasi objek. Masing-masing metodologi membawa notasi sendiri-sendiri, yang mengakibatkan timbul masalah baru apabila kita bekerjasama dengan kelompok/perusahaan lain yang menggunakan metodologi yang berlainan.

Dimulai pada bulan Oktober 1994 Booch, Rumbaugh dan Jacobson, yang merupakan tiga tokoh yang boleh dikata metodologinya banyak digunakan mempelopori usaha untuk penyatuan metodologi pendesainan berorientasi objek.

Pada tahun 1995 direlease draft pertama dari UML (versi 0.8). Sejak tahun 1996 pengembangan tersebut dikoordinasikan oleh Object Management Group.

2.10. 2. 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 objects
4. Collaboration Diagram untuk memodelkan interaksi antar objects.

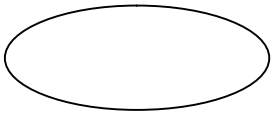
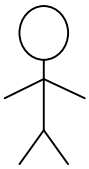
5. State Diagram untuk memodelkan perilaku objects di dalam sistem.
6. Activity Diagram untuk memodelkan perilaku Use Cases dan objects di dalam system.
7. Class Diagram untuk memodelkan struktur kelas.
8. Object Diagram untuk memodelkan struktur object.
9. Component Diagram untuk memodelkan komponen object.
10. Deployment Diagram untuk memodelkan distribusi aplikasi.

Berikut akan dijelaskan empat macam diagram yang paling sering digunakan dalam pembangunan aplikasi berorientasi object, yaitu use case diagram, sequence diagram, Activity diagram, dan class diagram.

1. 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. Actor merepresentasikan orang yang akan mengoperasikan atau orang yang berinteraksi dengan sistem aplikasi. Use case merepresentasikan operasi-operasi yang dilakukan oleh aktor. Use case digambarkan berbentuk elips dengan nama operasi dituliskan di dalamnnya. Actor yang melakukan operasi dihubungkan dengan garis lurus ke usecase.

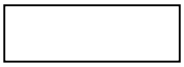
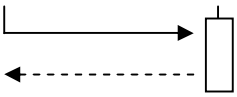
Tabel 2.1 Keterangan Komponen *Use Case Diagram*

Notasi	Keterangan
	Use Case : mengggambarkan prilaku aplikasi termasuk interaksi antar actor dengannya
	Pembatas system : tanda yang menggelompokan beberapa use case menjadi suatu system dan dan pembatas antara internal dan external
	Actor : menggambarkan pengguna aplikasi yang juga menjelaskan objek yang harus
	Asosiasi : memodelkan koneksi antar objek ke kelompok yang berbeda
	Hubungan : menggambarkan hubungan yang ada atau use-case
<<include>>	Label<<include>> : tipe hubungan yang menerangkan bahwa suatu use-case

2. 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.

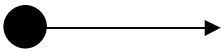
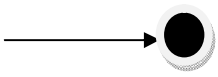
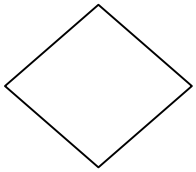
Tabel 2.2 Keterangan Komponen *Sequence Diagram*

Notasi	Keterangan
	Actor : pengguna aplikasi yang juga menjelaskan objek yang harus dikerjakan aplikasi.
	Objek : Elemen yang dituju oleh actor
	Transisi : arah aliran proses menuju objek
	Transisi bolak-balik : arah aliran proses menuju objek yang timbal balik

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)*

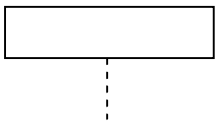
Tabel 2.3 Keterangan komponen *Activity Diagram*

Notasi	Keterangan
	Aktifitas : penggambaran sebuah aksi yang akan dijalankan hingga selesai
	Titik awal : titik yang menandakan akan dimulainya suatu aktifitas
	Titik akhir : titik yang menandakan proses keseluruhan telah selesai
	Transisi : arah aliran proses menuju aktifitas tersebut sesudah selesai
	Percabangan atau penggabungan : titik dimana suatu aliran dipisah berdasarkan kondisi tertentu atau beberapa aliran digabungkan kembali

4. 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.

Tabel 2.4 Keterangan Komponen *Class Diagram*

Notasi	Keterangan
	Objek : sumber atau tujuan pesan
	Pesan sederhana : memodelkan arah tujuan pesan yang diterima.
	Rentan hidup : jangka waktu suatu objek antara pembuatan dan penghapusan
	Aktivasi : jangka waktu pemrosesan yang dibutuhkan suatu objek