BAB II

LANDASAN TEORI

2.1 Aplikasi

Menurut ALI ZAKI dan SMITDEV COMMUNITY Aplikasi adalah komponen yang berguna melakukan pengolahan data meupun kegiatan-kegiatan seperti pembuatan dokumen atau pengolahan data.

Aplikasi adalah bagian PC yang berinteraksi langsung dengan user. Aplikasi berjalan di atas sistem operasi, agar aplikasi bisa diaktifkan, kita perlu melakukan instalasi sistem operasi terlebih dahulu.

(sumber : http://carapedia.com/pengertian_definisi_aplikasi_info2062.html/27-juni-2012)

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 *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 s*ofware* 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 Budaya

Kebudayaan sangat erat hubungannya dengan masyarakat. *Melville J. Herskovits dan Bronislaw Malinowski* mengemukakan bahwa segala sesuatu yang terdapat dalam masyarakat ditentukan oleh kebudayaan yang dimiliki oleh masyarakat itu sendiri. Istilah untuk pendapat itu adalah *Cultural-Determinism.*

Koentjaraningrat (2000:180) menyatakan "*kebudayaan adalah keseluruhan sistem gagasan, tindakan dan hasil karya manusias dalam rangka kehidupan masyarakat yang dijadikan milik diri manusia dengan belajar*".

Herskovits memandang kebudayaan sebagai sesuatu yang turun temurun dari satu generasi ke generasi yang lain, yang kemudian disebut sebagai *superorganic*.

Menurut *Andreas Eppink*, kebudayaan mengandung keseluruhan pengertian nilai sosial,norma sosial, ilmu pengetahuan serta keseluruhan struktur-struktur sosial, religius, dan lain-lain, tambahan lagi segala pernyataan intelektual dan artistik yang menjadi ciri khas suatu masyarakat.

Menurut *Edward Burnett Tylor* (*2003: 136)*, *"kebudayaan merupakan keseluruhan yang kompleks, yang di dalamnya terkandung pengetahuan, kepercayaan, kesenian, moral, hukum, adat istiadat, dan kemampuan-kemampuan lain yang didapat seseorang sebagai anggota masyarakat"*.

Menurut Selo Soemardjan dan Soelaiman Soemardi, kebudayaan adalah sarana hasil karya, rasa, dan cipta masyarakat.

Dari berbagai definisi tersebut, dapat diperoleh pengertian mengenai kebudayaan adalah sesuatu yang akan memengaruhi tingkat pengetahuan dan meliputi sistem ide atau gagasan yang terdapat dalam pikiran manusia, sehingga dalam kehidupan sehari-hari, kebudayaan itu bersifat abstrak.

2.3 Android



Gambar 2.1 Logo Android

Android merupakan subset perangkat lunak untuk perangkat *mobile* yang meliputi sistem operasi, *middleware* dan aplikasi inti yang di-*release* oleh Google. Sedangkan Android SDK (*Software Development Kit*) menyediakan *Tools* dan API yang diperlukan untuk mengembangkan aplikasi pada *platform* Android dengan menggunakan bahasa pemrograman Java. Dikembangkan bersama antara Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, NVIDIA yang tergabung dalam OHA (*Open Handset Alliance*) dengan tujuan membuat sebuah standar terbuka untuk perangkat bergerak (*mobile device*).

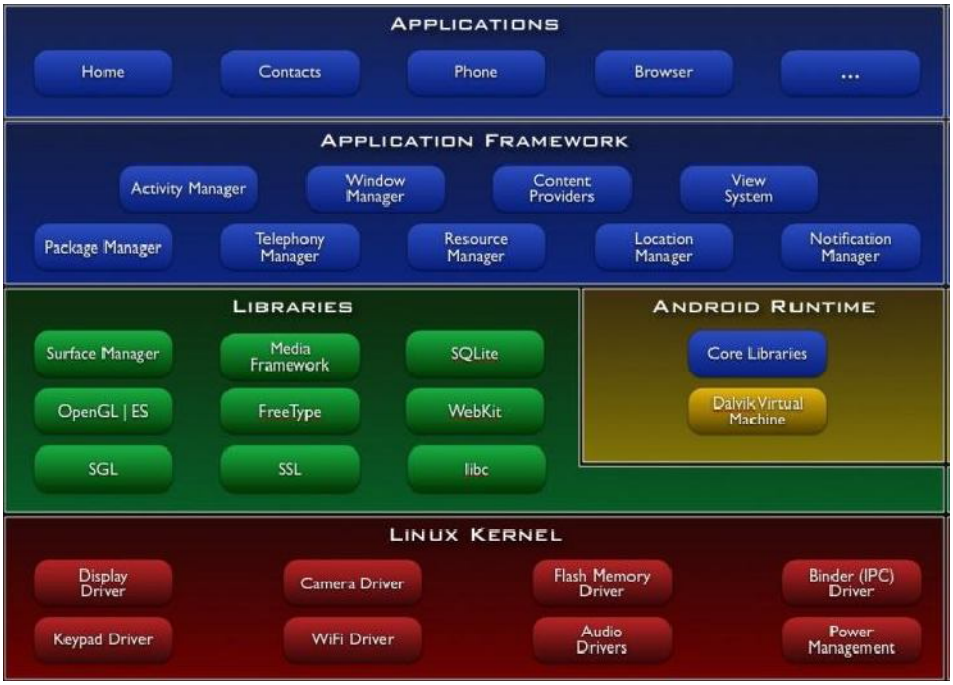
2.4 Sejarah Android

Pada Juli 2000, Google bekerjasama dengan Android Inc., perusahaan yang berada di Palo Alto, California Amerika Serikat. Para pendiri Android Inc. bekerja pada Google, diantaranya 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, Samsung dan lainnya.

2.5 Arsitektur Android

Arsitektur Android menunjukkan komponen-komponen utama yang terdapat pada sistem operasi Android. Berikut ini arsitektur Android yang digambarkan pada Gambar 2.1.



Gambar 2.2 Arsitektur Platform Android

(Sumber : http://developer.android.com/quide/basics/what-is-android.html)

1. Linux Kernel

Android bukan Linux, akan tetapi Android dibangun diatas Linux *Kernel*, yaitu versi 2.6 sehingga kehandalannya bisa dipercaya. Untuk inti sistem *service* Linux yang digunakan seperti keamanan, manajemen memori, proses manajemen, *network*, dan *driver* model. Seperti yang terlihat digambar (Gambar II.3), Linux *Kernel* menyediakan *Driver* Layar, Kamera, *Keypad*, WiFi, *Flash Memory*, *Audio*, dan IPC (*Interprocess Communication*) untuk mengatur aplikasi dan keamanan.

2. Libraries

Android menyertakan *libraries* C/C++ yang digunakan oleh berbagai komponen dari sistem Android. Kemampuan ini disediakan kepada *Developer* aplikasi melalui *Framework* Aplikasi Android. Beberapa inti *libraries* tercantum dibawah ini:

1. *System C Library* – Variasi dari implementasi BSD-berasal pelaksana sistem standar C *library* (*libc*), sesuai untuk perangkat *embedded* berbasis Linux.
2. *Media Libraries* – *PacketVideo* berdasarkan OpenCORE; *library* medukung pemutaran rekaman dan populer banyak format audio dan

video, serta *file* gambar, termasuk MPEG4, H.264, MP3, AAC, AMR, JPG, dan PNG.

3. *Surface Manager* – mengelola akses ke subsistem layar, lapisan komposit 2D dan grafis 3D dari beberapa aplikasi.
4. LibWebCore – mesin *web modern* yang *powerfull* yang baik *browser* Android *embedded web*.
5. 3D *Libraries* – sebuah pelaksana berdasarkan openGL ES 1,0 API; perpustakaan baik menggunakan perangkat keras akselerasi 3D (apabila tersedia) atau yang disertakan, sangat optimal 3D *software rasterizer*.
6. FreeType – bitmap dan vektor *font rendering*.
7. SQLite – mesin *database* yang kuat dan ringan, dan penghubung tersedia untuk semua aplikasi.

3. Android Runtime

Android terdiri dari satu set *core libraries* yang menyediakan sebagian besar fungsi yang sama dengan yang terdapat pada *core libraries* bahasa pemrograman Java. Setiap aplikasi menjalankan prosesnya sendiri dalam Android, dengan masing-masing instan dari mesin virtual Dalvik (*Dalvik VM*). Dalvik dirancang agar perangkat dapat menjalankan multiple VMs secara efisien. Mesin Virtual Dalvik mengeksekusi *file* dalam Dalvik *executable* (.dex), sebuah format yang dioptimalkan untuk memori yang kecil.

Dalvik VM berbasis, berjalan, dan dikompilasi oleh *compiler* bahasa Java yang telah ditransformasikan ke dalam .dex format oleh tool “dx” yang telah disertakan. Dalvik VM bergantung pada *kernel* Linux untuk berfungsi seperti *threading* dan manajemen memori tingkat rendahnya.

4. FrameWork Aplikasi

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