BAB 2
LANDASAN TEORI

2.1 Perangkat Lunak

Perangkat lunak adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan. Sebuah program komputer tanpa terasosiasi dengan dokumentasinya maka belum dapat disebut perangkat lunak (*software*). Sebuah perangkat lunak juga sering disebut dengan sistem perangkat lunak. Sistem berarti kumpulan komponen yang saling terkait dan mempunyai satu tujuan yang ingin dicapai.

Sistem perangkat lunak berarti sebuah sistem yang memiliki komponen berupa perangkat lunak yang memiliki hubungan satu sama lain untuk memenuhi kebutuhan pelanggan (*customer*). Pelanggan (*customer*) adalah orang atau organisasi yang memesan atau membeli perangkat lunak (*software*) dari pengembang perangkat lunak atau bisa dianggap bahwa pelanggan (*customer*) adalah orang atau organisasi yang dengan sukarela mengeluarkan untuk memesan atau membeli perangkat lunak. *User* atau pemakai perangkat lunak adalah orang yang memiliki kepentingan untuk memakai atau menggunakan perangkat lunak untuk memudahkan pekerjaannya[7].

Secara umum perangkat lunak ini dapat diklasifikasikan menjadi 2 [8], yaitu:

1. Perangkat Lunak Sistem Operasi.

   Adalah perangkat lunak *(software)* yang berfungsi melakukan operasi yang mengkonfigurasi segala aktifitas komputer yang mendukung operasi sistem aplikasi dan mengendalikan semua perangkat komputer agar dapat berjalan sesuai dengan fungsinya.

   Contoh: *WINDOWS , MACINTOSH, LINUX*, dan lain sebagainya.

2. Perangkat Lunak Aplikasi.

   Adalah program *(software)* yang digunakan untuk aplikasi dibidang tertentu secara terfokus. Perangkat lunak aplikasi yang membantu user sehingga dapat bekerja lebih efektif dan efisien. Aplikasi ini dibagi atas beberapa bagian, yaitu:

   a) *Business & Office Application*

      a. Aplikasi perkantoran dipergunakan untuk menyelesaikan pekerjaan kantor seperti mengetik (*Word processing, document management*) membuat tabel kerja (*Worksheet*), membuat Database sederhana (*MS. Access*) ataupun mengolah *image*/citra sederhana. Contoh aplikasi perkantoran yang terpaket komplit: *Mocrosoft Office, Koffice, Star Office, Abiword, Open Office, MS_Office for MAC,* dan lain sebagainya.

      b. Aplikasi bisnis lainnya juga meliputi aplikasi pengolahan data keuangan, misal (*Personal Finance Application & Business Acounting Application*). Contoh: Aplikasi perhitungan neraca, rugi laba perusahaan, Aplikasi perhitungan pajak personal & perusahaan, Aplikasi manajemen proyek, Aplikasi persediaan stok barang dan penjualan, dan lain sebagainya.

   b) *Database Application*

      Aplikasi yang digunakan dalam pengolahan data baik yang berukuran kecil maupun besar bisa digunakan secara *stand-alone* (tunggal) maupun dalam sistem berbasis jaringan lokal *Client server* maupun *webbase* (intranet maupun internet). Contoh: *MS SQL, FoxBase, Oracle, MySQL, PostgresSQL, SQL Server*, dan lain sebagainya.

   c) *Graphics Design Application*

      a. Aplikasi yang digunakan untuk membuat sketsa dua dimensi untuk desain ruangan, perabotan, mesin-mesin model industri, juga model tubuh manusia, hewan, dan lain sebagainya. Contoh: *AutoCad, Pro Design, CorelDraw, Adobe Ilistrator,* dan lain sebagainya.

b. Ada juga yang memasukan dalam golongan ini aplikasi *image, audio, video manipulator*, yaitu aplikasi yang digunakan untuk mengolah citra, suara hingga gambar 3D untuk pembuatan animasi, pengolahan *image, photo, audio, video editing*, dan lain sebagainya. Contoh: *Adobe (Photoshop, After Effect, Premire, & Audition), Corel (Draw, RAVE, Photo Paint), Ulead Media studio, GIMP (In Linux), Cool Edit Pro, Sound Force, Macromedia (Flash, Director), 3D Studio Max,* dan lain sebagainya.

d) *Antivirus & Utility Tools*

e) *Software* ini digunakan untuk operasi minor (*Background*) yang bekerja di belakang sistem, berguna untuk meningkatkan kinerja sistem operasi atau aplikasi dan juga performa *hardware*. Terdapat 2 jenis utilitas di dalam sistem operasi:

f) *Utilitas Internal,* sudah termasuk dalam paket yang di instalasi dalam sistem operasi seperti *scandisk* di *Windows* atau perintah *fsck* (*file system check*) di OS Linux.

g) *Utilitas eksternal,* yang harus diinstalasi kemudian pada OS, seperti *System Optimizer, PC Tools, Data/File Recovery, PC Maintenance, Internet Utilities, Norton System Work, Registry Clean & Repair, Hardware Repair*.

    a. Sementara untuk proteksi virus dan antivirus, contohnya adalah antivirus produk *Mc Affee, Grisoft, Norton, PandaSoft*, dan lain sebagainya

h) *Development Tools*

    a. Merupakan sistem bahasa pemrograman komputer yang dilengkapi dengan *compiler* tertentu sebagai sumber media penyusunan program aplikasi. Contoh: *QuikBasic, Clipper, Visual Basic, Turbo Pascal, Visual C++, GCC, Turbo Assembler, Phyton, KDEvelopt, Gdekstop,* dan lain sebagainya.

    b. Termasuk dalam jenis aplikasi ini adalah *web development tools*, yaitu aplikasi yang digunakan untuk membuat/mendesain tampilan

website (*homepage*) di internet, seperti *FrontPage, Macromedi Dreamweaver, PHPedit, WebPage Editor*, dan lain sebagainya

i) *Communication Application*

- Aplikasi yang digunakan untuk membangun komunikasi antara satu komputer dengan komputer lain dalam jaringan komputer local maupun global. Contoh: *E-mail & Fax Application, Remote Access, Facebook chat, YM, GTalk, MSN, Terminal Emulation, Instant Messaging, Voice Recognition, Web Browser, Internet Telephony,* dan lain sebagainya.

j) Jenis Aplikasi Lain

1. *Multimedia Player,* seperti *WinAmp, XMMS, Power DVD, JetAudio, Windows Media Player,* dan lain sebagainya.
2. *Hacker Tools,* seperti *Find Password, Keygen/Crack Software License, Scanning Port,* dan lain sebagainya.
3. *Network & Security Application,* seperti *Firewall, Intruder Detections, Network Monitoring,* dan lain sebagainya.
4. *PC Games Apllication,* dan masih banyak lagi.

2.2. Pemrograman Berorientasi Objek

Pendekatan berorientasi objek merupakan suatu teknik atau cara pendekatan dalam melihat permasalahan dan sistem (sistem perangkat lunak, sistem informasi, atau sistem lainnya). Pendekatan berorientasi objek akan memandang sistem yang akan dikembangkan sebagai suatu kumpulan objek yang berkorespondensi dengan objek-objek dunia nyata.

Ada banyak cara untuk mengabstraksikan dan memodelkan objek-objek tersebut, mulai dari abstraksi objek, kelas, hubungan antar kelas sampai abstraksi sistem. Saat mengabstraksikan dan memodelkan objek, data dan proses-proses yang dimiliki oleh objek akan dienkapsulasi (dibungkus) menjadi satu kesatuan.

Sistem berorientasi objek merupakan sebuah sistem yang kompunennya dibungkus (dienkapsulasi) menjadi kelompok data dan fungsi. Setiap

komponen dalam sistem tersebut dapat mewarisi atribut, sifat, dan kompunen lainnya yang dapat berinteraksi satu sama lain.

Terdapat beberapa konsep utama pada metodologi berorientasi objek, diantaranya [7]:

a) Kelas (*class*), kumpulan objek-objek dengan karakteristik yang sama. Kelas merupakan definisi statik dari himpunan objek yang sama yang mungkin lahir atau diciptakan dari kelas tersebut. Sebuah kelas akan mempunyai sifat (atribut), kelakuan (operasi/metode), hubungan (*relationship*), dan arti. Suatu kelas dapat diturunkan dari kelas yang lain, dimana atribut dari kelas semula dapat diwariskan ke kelas yang baru.
b) Objek (*object*), abstraksi dari sesuatu yang mewakili dunia nyata seperti benda, manusia, satuan organisasi, tempat, kejadian, struktur, status, atau hal-hal lain yang bersifat abstrak. Objek merupakan suatu entitas yang mampu menyimpan informasi (status) dan mempunyai operasi (kelakuan) yang dapat diterapkan atau dapat berpengaruh pada status objeknya. Objek mempunyai siklus hidup yaitu diciptakan, dimanipulasi, dan dihancurkan.
c) Abstraksi (*abstraction*), prinsip untuk merepresentasikan dunia nyata yang kompleks menjadi suatu bentuk model yang sederhana dengan mengabaikan aspek-aspek lain yang tidak sesuai dengan permasalahan.
d) Enkapsulasi (*encapsulation*), pembungkusan atribut data dan layanan (operasi-operasi) yang dimiliki objek untuk menyembunyikan implementasi dari objek sehingga objek lain tidak mengetahui cara kerjanya.
e) Pewarisan (*inheritance*), mekanisme yang memungkinkan suatu objek mewarisi sebagian atau seluruh definisi dan objek lain sebagai bagian dari dirinya.

Polimorfisme (*polymorphism*), kemampuan seuatu objek untuk digunakan dibanyak tujuan yang berbeda dengan nama yang sama sehingga menghemat baris program.

2.3 Basis Data

Sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan. Pada intinya basis data adalah media untuk penyimpanan data agar dapat diakses dengan mudah dan cepat.

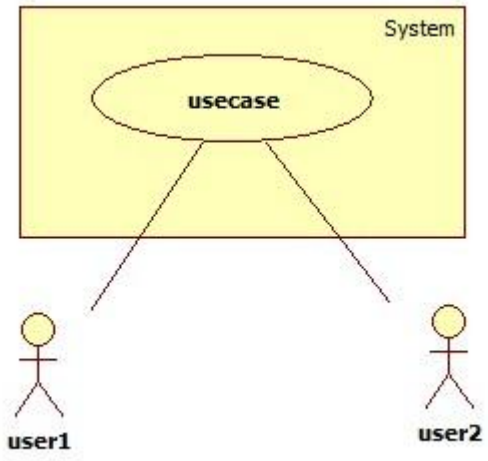
Sistem informasi tidak dapat dipisahkan dengan kebutuhan akan basis data apapun bentuknya, entah berupa *file* teks ataupun *Database Management System* (DBMS) [7]. Kebutuhan basis data dalam sistem informasi meliputi memasukan, menyimpan, dan mengambil data serta membuat laporan berdasarkan data yang telah disimpan.

2.4 UML (*Unified Modeling Language)*

UML (*Unified Modeling Language*) adalah sebuah bahasa yang berdasarkan grafik/gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan software berbasis OO (*Object-Oriented*)[16]. UML sendiri juga memberikan standar penulisan sebuah sistem blue print, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema database, dan komponen-komponen yang diperlukan dalam sistem software. *Unified Modelling Language* (UML) juga merupakan sebuah bahasa yg telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem.

Terdapat beberapa diagram yang umum digunakan dalam pemodelan UML diantaranya [7]:

1. *Use Case Diagram,* merupakan pemodelan untuk kelakuan (*behavior*) sistem yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antar satu atau lebih aktor dengan sistem yang akan dibuat. Secara kasar *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi itu.



Gambar 2.1 Gambar usecase

2. *Class Diagram,* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki atribut dan metode (operasi). Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas. Metode (operasi) fungsi-fungsi yang dimiliki oleh suatu kelas. Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki kelas *main*, kelas *view*, kelas *controller*, serta kelas *model*.



Gambar 2.2 Gambar class diagram

3. *Sequence Diagram,* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk

menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use cse* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case*.

4. *Activity Diagram*, menggambarkan aliran kerja (*workflow*) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, melainkan aktivitas yang dapat dilakukan oleh sistem.
5. *Component Diagram*, dibuat untuk menunjukan organisasi dan ketergantungan diantara kumpulan komponen dalam sebuah sistem. Diagram komonen fokus pada komponen sistem yang dibutuhkan dan ada di dalam sistem. Komponen dasar yang biasanya ada dalam suatu sistem adalah komponen *user interface* yang menagani tampilan, komponen *bussiness processing* yang menangani fungsi-fungsi proses bisnis, komponen *data* yang menangani manipulasi data, dan komponen *security* yang menangani keamanan sistem. Komponen lebih terfokus pada penggolongan secara umum fungsi-fungsi yang diperlukan.

2.5 Aplikasi Mobile

Aplikasi *mobile* adalah sebuah aplikasi yang memungkinkan untuk melakukan mobilitas dengan menggunakan perlengkapan seperti telepon genggam (*hanphone*), PDA, atau *smartphone*. Aplikasi *mobile* dapat mengakses dan menggunakan sebuah aplikasi secara *nirkabel* dengan menggunakan perangkat *mobile*, Penggunakan aplikasi *mobile* hanya memerlukan *handphone* yang sudah dilengkapi dengan fasilitas GPRS dan koneksinya[17].

Terdapat beberapa aspek yang harus diperhatikan khususnya perangkat keras dalam membangun sebuah aplikasi *mobile*. Selain itu pertimbangan terhadap keterbatasan pada piranti *mobile* pun harus di perhatikan, antara lain:

1. Keterbatasan dari kecepatan *processor* dalam mengeksekusi program
2. Keterbatasan memori penyimpanan dan RAM
3. Perbedaan ukuran layar secara fisik dan resolusi pada masing-masing piranti
4. Keterbatasan input pada masing-masing piranti mobile
5. Ketahanan baterai yang berbeda pada setiap piranti mobile

Selain itu dari segi perangkat lunak (*software*), kompatibilitas dan *mobile* pendukung cukup berpengaruh dalam menjalankan sebuah aplikasi *mobile*.

2.6 Internet

Internet merupakan jaringan global komputer dunia, besar dan sangat luas dimana setiap komputer saling terhubung satu sama lainnya dari negara ke negara lainnya di seluruh dunia dan berisi berbagai macam informasi, mulai dari teks, gambar, audio, video, dan lain sebagainya yang terhubung melalui protokol TCP/IP. Dalam mengatur integrasi dan komunikasi jaringan komputer ini menggunakan protokol yaitu TCP/IP. TCP (*Transmission Control Protocol*) bertugas untuk memastikan bahwa semua hubungan bekerja dengan benar, sedangkan IP (*Internet Protocol*) yang mentransmisikan data dari satu komputer ke komputer lain. TPC/IP secara umum berfungsi memilih rute terbaik transmisi data, memilih rute alternatif jika suatu rute tidak dapat di gunakan, mengatur dan mengirimkan paket-paket pengiriman data[18].

2.7 Pengujian

Pengujian sistem adalah pengujian yang dilakukan pada sistem komputer secara keseluruhan.

Pengujian perangkat lunak disebut dengan *alpha testing* dan *beta testing*.

yaitu : Pengujian Alpha dan Beta

Kebanyakan pembuat produk software menggunakan sebuah proses yang disebut pengujian alpha dan beta, untuk mengungkap kesalahan yang hanya dapat ditemukan oleh end user (pemakai akhir). Pengujian alpha dilakukan pada sisi pengembang oleh seorang pelanggan. Pengujian alpha dilakukan pada sebuah lingkungan yang terkontrol dan pengoperasiannya diawasi oleh

pengembang atau programmer. Pengujian alpha biasanya dilakukan saat pelatihan atau training software.

Pengujian beta dilakukan pada satu atau lebih end user (biasa disebut multi user). Pada pengujian beta biasanya programmer tidak melakukan pengawasan, sehingga pengujian beta merupakan sebuah aplikasi ‘live’ dari perangkat lunak. End user merekam/mencatat semua ‘bug’ yang ditemui dan dilaporkan ke programmer dalam interval yang sudah terjadual [24].

2.8 Jenis Pembelajaran Bantuan Komputer

Media pembelajaran berbasis komputer atau CAI (Computer-Assisted Instruction) merupakan media pembelajaran yang memanfaatkan komputer sebagai pembantu atau tambahan dalam belajar, pemanfaatannya meliputi penyajian informasi isi materi pelajaran, latihan, atau kedua-duanya. Media pembelajaran CAI memungkinkan dalam penyajian tutorial untuk pemahaman, serta praktik atau simulasi untuk penerapan. [15].

2.8 Client/server

Teknologi client-server saat ini sudah sangat populer dikalangan masyarakat terutama didunia jaringan komputer baik internet maupun intranet, secara harifah client bisa berarti pelanggan , nasabah atau sesuatu yang harus dilayani yang jumplahnya biasanya lebih dari satu, sementara server berartipelayan atau yang melayani.

Menurut:*http://compnetworking.about.com/od/basicnetworkingfaqs/a/client-server.htm*

Definisi yang paling dasar client / server berasal dari usenet yang sesuai FAQ [13] client / server adalah arsitektur komputasi yang melibatkan proses klien meminta layanan dari server proses.

2.9 Multimedia

Konsep multimedia telah banyak diterapkan dalam dunia pendidikan.Pembelajaran menggunakan multimedia interaktif berkembang atas dasar pembelajaran konvensional yang tidak bisa memenuhi kebutuhan peserta didik dalam pembelajaran. Multimedia dapat menjadikan suatu aplikasi menjadi sangat interaktif dan menyajikan interface yang menarik.

2.9.1 Definisi Multimedia

Menurut Vaughan(2004), multimedia merupakan mombinasiteks, seni, suara, gambar, animasi dan video yang disampaikan dengan komputer atau dimanipulasi secara digital dan dapat disampaikan atau di kontrol secara interaktif. ada tiga jenis multimedia yaitu[21] :

1. Multimedia Interaktif

   pengguna dapat mengontrol apa dan kapan elemen-elemen multimedia akan dikirimkan atau ditampilkan
2. Multimedia Hiperaktif
3. Multimedia jenis in mempunyai struktur dari elemen-elemen terkait dengan pengguna yang dapat mengarahkanya. dapat dikatakan bahwa multimedia jenis ini mempunyai banyak tautan (link) yang menghubungkan elemen multimedia yang ada.
4. Multimedia linear

   Pengguna hanya menjadi penonton dan menikmati produk multimedia yang disajikan dari awal hingga akhir.

2.9.2 Android

Android adalah sebuah sistem operasi untuk perangkat *mobile* berbasis Linux yang mencakup sistem operasi, *middleware* dan aplikasi. Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Awalnya, Google Inc. membeli Android Inc., yang merupakan pendatang baru yang membuat peranti lunak untuk ponsel/*smartphone*. Kemudian untuk mengembangkan Android, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia.

Pada saat perilisan perdana Android, 5 November 2007, Android bersama *Open Handset Alliance* menyatakan mendukung pengembangan *open source* pada perangkat *mobile*. Di lain pihak, Google merilis kode–kode Android di bawah lisensi Apache, sebuah lisensi perangkat lunak dan *open platform* perangkat seluler.

Di dunia ini terdapat dua jenis distributor sistem operasi Android. Pertama yang mendapat dukungan penuh dari Google atau *Google Mail Services* (GMS) dan kedua adalah yang benar–benar bebas distribusinya tanpa dukungan langsung Google atau dikenal sebagai *Open Handset Distribution* (OHD). Tidak hanya menjadi sistem operasi di *smartphone*, saat ini Android menjadi pesaing utama dari Apple pada sistem operasi Tablet PC.

Pengembang memiliki beberapa pilihan ketika membuat apliaksi yang berbasis android. Sebagian besar pengembang menggunakan *Eclipse* yang tersedia secara bebas untuk merancang dan mengembangkan Aplikasi Android. *Eclipse* adalah IDE yang paling populer untuk pengembangan Android, karena memiliki Android *plug-in* yang tersedia untuk memfasilitasi pengembangan Android. Selain itu, *Eclipse* juga mendapat dukungan langsung dari Google untuk menjadi IDE pengembangan aplikasi Android, ini terbukti dengan adanya penambahan *plugin* untuk *eclipse* untuk membuat project android dimana *source software* langsung dari situs resmi Google. Akan tetapi, hal tersebut tidak menutup kemungkinan untuk menggunakan IDE yang lain seperti Netbeans untuk melakukan pengembangan android [19].

Beberapa versi Android yang telah dirilis diantaranya :

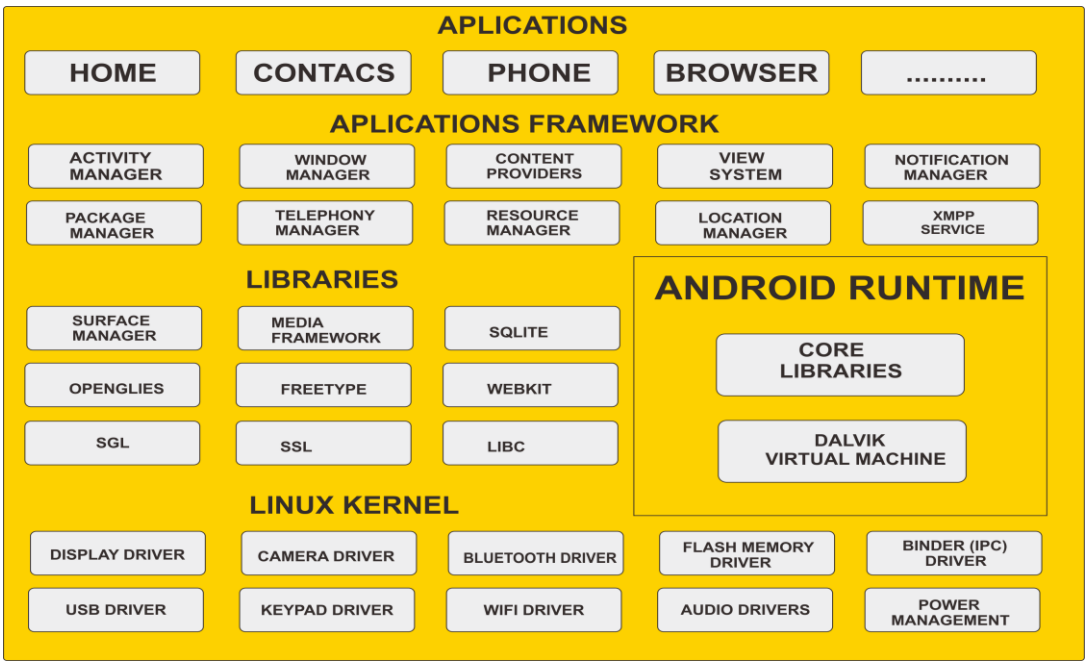
1. Android 1.0 / 1.1 (API level 1 / 2)
2. Android 1.5 *Cupcake* (API level 3)
3. Android 1.6 *Donut* (API level 4)
4. Android 2.0 / 2.0.1 / 2.1 *Eclair* (API level 5 / 6 / 7)
5. Android 2.2-2.2.3 *Frozen Yoghurt* (API level 8)
6. Android 2.3-2.3.2 / 2.3.3-2.3.7 *Gingerbread* (API level 9 / 10)
7. Android 3.0 / 3.1 / 3.2 *Honeycomb* (API level 11 / 12 / 13)
8. Android 4.0-4.0.2 / 4.0.3-4.0.4 *Ice Cream Sandwich* (API level 14 / 15)
9. Android 4.1 / 4.2 / 4.3 *Jelly Bean* (API level 16 / 17 / 18)
10. Android 4.4 *KitKat* (API level 19)



Gambar 2.3 logo android

2.10 Arsitektur Android

Berikut ini adalah garis besar Arsitektur Android yang dapat dijelaskan dan digambarkan sebagai berikut :



Sumber: http://candra.web.id/2011/01/30/arsitektur-android-os/

Gambar 2.4 Gambar Arsitektur android

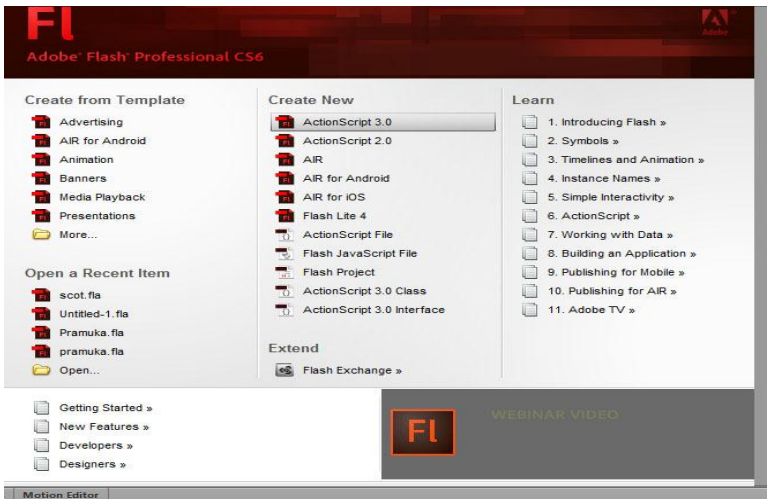
2.10.1 Fitur Android

Beberapa fitur menurut Safaat 7 yang tersedia pada Android adalah sebagai berikut ini :

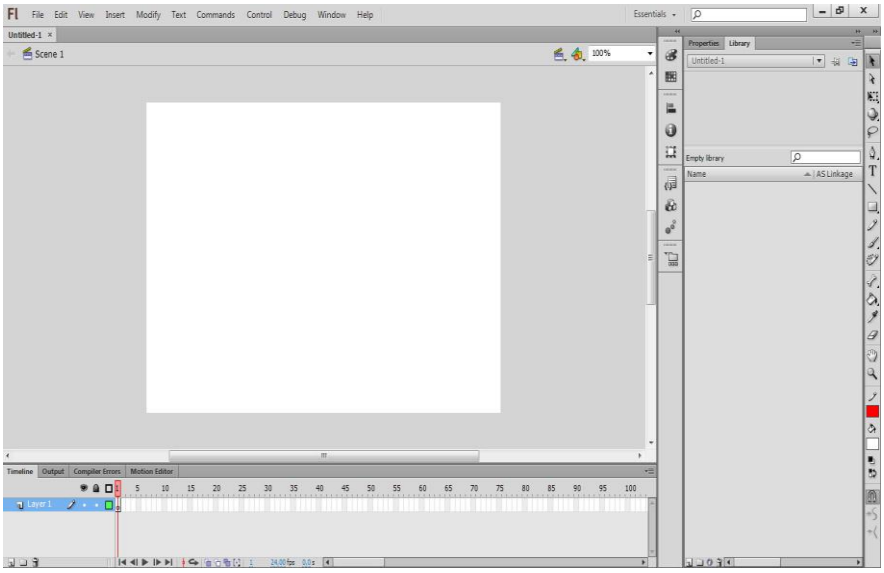
1. Kerangka aplikasi: memungkinkan penggunaan dan penghapusan komponen yang tersedia.
2. Dalvik mesin virtual: mesin virtual difungsikan untuk perangkat telepon seluler.
3. Grafik: gambar 2D dan gambar 3D berdasarkan pustaka OpenGL.
4. SQLite: sebagai penyimpanan data.
5. Mendukung media: audio, video, dan berbagai format gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF).
6. GSM, *Bluetooth*, EDGE, 3G, 4G dan WiFi (tergantung perangkat keras yg dimiliki) .
7. Kamera, *Global Positioning System*(GPS), kompas, NFC dan *accelerometer* (tergantung perangkat keras yg dimiliki).

2.10.2 Adobe Flash CS6

AdobeAdobe Flash CS6 adalah sebuah pogram animasi yang telah banyak digunakan oleh para Animator untuk menghasilkan animasi yang professional. (Tim Madcoms,2006)[24].



Gambar 2.5 Gambar utama adobe flash cs6



Gambar 2.6 Menu utama adobe flash cs6

2.10.3 Action Script 3.0

ActionScript 3.0 adalah bahasa pemrograman berorientasi objek yang menandakan sebuah langkah penting dalam evolusi kemampuan Flash Player runtime. Motivasi pembuatan ActionScript 3.0 adalah untuk menciptakan bahasa ideal untuk cepat membangun pengayaan aplikasi Internet, yang telah menjadi bagian penting dari pengalaman dalam pembuatan web [6].

2.10.4 XML (eXtensible Markup Language)

XML (eXtensible Markup Language) XML kependekan dari eXtensible Markup Language, dikembangkan mulai tahun 1996 dan mendapatkan pengakuan dari W3C pada bulan Februari 1998. Teknologi yang digunakan pada XML sebenarnya bukan teknologi baru, tapi merupakan turunan dari SGML yang telah dikembangkan pada awal 80-an dan telah banyak digunakan pada dokumentasi teknis proyek-proyek berskala besar. Ketika HTML dikembangkan pada tahun 1990, para penggagas XML mengadopsi bagian paling penting pada SGML dan dengan berpedoman pada pengembangan HTML menghasilkan markup language yang tidak kalah hebatnya dengan SGML.[22]

XML dengan cepat diadopsi sebagai standar untuk pertukaran data, khususnya untuk penggunaan lintas aplikasi dan platform. Kelebihan dari XML adalah karakteristiknya yang extensible dan platform independent.

2.10.5 SQLite



Gambar 2.7 logo sqlite

SQLite adalah suatu library yang menerapakan mesin database self-contained, serverless, zero-configuration, dantransactional. SQLite membutuhkan sedikit dukungan dari library eksternal atau dari system operasi. SQLite dalam memproses akses database baik itu read atau write secara langsung dari file database tanpa ada intermediary (penengah) proses server. Kebanyakan mesin SQL database mengimplementasikannya sebagai proses server yang terpisah. Ada beberapa API yang mendukung SQLite database,salah satunya adalah PySQLite sebagai penerapan Python DatabaseAPI interface 2.0 (DBAPI2). DBAPI2 didefinisikan untuk memberikan pendekatan antara Python modul yang digunakan untuk mengakses database. Tujuan pendefinisian ini yaitu memberikan ketetapan yang mengarahkan pada kemudahan pemahaman pada modul-modul Python. Sehingga modul tersebut memberikan portabilitas antar database, dan pencapaian koneksi dari bahasa pemrograman Python. Modul PySQLite adalah contoh penerapan dari DBAPI2 untuk database SQLite dan telah ter-include pada modul Python yang dikenal sebagai paket sqlite3. Fitur-fitur SQLite[20] :

- *no-depedency*, SQLite *library* didesain untuk mandiri dan memerlukan bantuan yang minimal dari library lain sehingga bisa mudah diaplikasikan pada aneka *platform*
- *serverless* dan *zero-configuration*, SQLite tidak memerlukan proses server yang tersendiri untuk mengakses data. Keseluruhan operasi di SQLite adalah berdasarkan konsep baca dan tulis langsung ke *file* di *storage* atau *memory*.
- SQL *compliance*, SQLite mendukung 90% *standard* SQL *syntax*. Hal ini akan mempermudah kurva belajar para pengembang aplikasi yang sudah terbiasa dengan SQL Programming
- dukungan banyak *platform*, SQLite memiliki kompatibilitas yang tinggi untuk diaplikasikan pada banyak *platform* dan beraneka jenis bahasa pemrograman.