BAB II

LANDASAN TEORI

2.1 Sistem Operasi Android

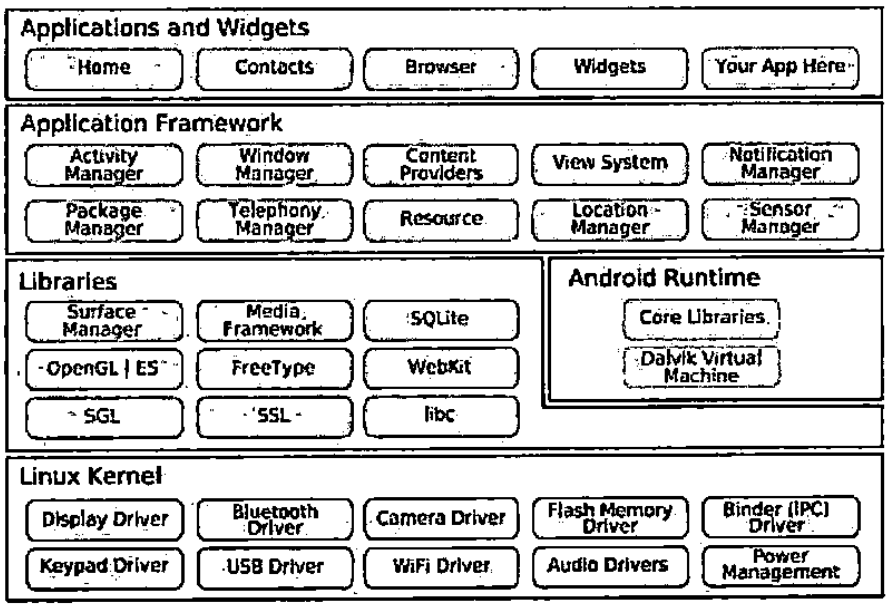
Android adalah sistem operasi yang ditargetkan untuk piranti mobile, yang arsitekturnya didasarkan pada sistem operasi Linux. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi yang dapat berjalan pada berbagai piranti mobile. Untuk mengembangkan Android dibentuklah Open Handset Alliance, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk di antaranya Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia.

Pada saat perilisan perdana Android, 5 November 2007, Android bersama Open Handset Alliance menyatakan mendukung pengembangan standar terbuka pada perangkat seluler. Di lain pihak, Google merilis kode-kode Android di

7. Android versi 3.0/3.1 (Honeycomb)
8. Android versi 4.0 (ICS :Ice Cream Sandwich)

2.1.1 Arsitektur Sistem Operasi Android

Sistem operasi Android terdiri atas 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 Arsitektur Android

2.1.1.1 Kernel Linux

Bagian paling bawah pada arsitektur Android ini adalah kernel. Google menggunakan kernel Linux versi 2.6, yang mencakup memory management, security setting, power management, dan beberapa driver hardware. Kernel berperan sebagai abstraction layer antara hardware dan keseluruhan software. Sebagai contoh, HTC GI dilengkapi dengan kamera. Kernel Android mempunyai driver kamera yang memungkinkan pengguna mengirimkan perintah kepada hardware kamera tersebut.

2.1.1.2 Libraries

Android menyertakan libraries C/C++ yang digunakan oleh berbagai komponen dari sistem Android. Library ini nantinya akan digunakan oleh Developer aplikasi melalui Framework Aplikasi Android. Beberapa libraries inti yang tersedia oleh Android adalah sebagai berikut:

- System C Library – Impelemntasi turunan BSD pada standar sistem library bahasa C (libc) yang diset untuk tertanam pada perangkat berbasis Linux.
- Media Libraries – menggunakan PacketVideo OpenCORE; library ini mendukung pemutaran dan perekaman dengan berbagai format multimedia yang populer digunakan seperti PNG, JPEG, AAC, MPEG4, MP3, H.264, ACC, dan AMR.
- Surface Manager – mengatur pembuatan grafik 2D dan 3 D dan

- LibWebCore – mesin web browser untuk Android browser.
- SGL – bertanggung jawab terhadap tampilan grafis 2D.
- 3D libraries – adalah implementasi dari OpenGL ES 1.0 API. Library ini menggunakan hardware pendukung 3D dan software rasterisasi 3D.
- FreeType – bertugas untuk me-render bitmap font dan vector font.
- SQLite – sistem database yang bisa digunakan dalam pengembangan aplikasi.

2.1.1.3 Android Runtime

Android terdiri atas sekumpulan core library yang menyediakan hampir semua fungsi yang tersedia pada core library bahasa pemrograman Java. Setiap aplikasi Android berjalan secara terpisah pada sebuah instance di Dalvik Virtual Machine. Dalvik telah dirancang dengan sedemikian rupa sehingga hardware (smartphone) dapat berjalan pada beberapa Virtual Mechine (VM) dengan efisien. Dalvik VM akan mengeksekusi file yang berekstensi .dex (Dalvik Excutable) yang dioptimalkan untuk penggunaan memori. File .dex adalah format hasil transformasi dari file .class dari Java Compiler oleh dx tool.

2.1.1.4 Application Framework

Dengan tersedianya platform pengembangan terbuka, Android memberikan kemampuan kepada pengembang untuk membangun aplikasi yang canggih dan inovatif. Pengembang bebas untuk mengakses kelebihan hardware, mengakses

informasi lokasi, menjalankan background service, mengatur alarm, menambah notifikasi pada status bar dan masih banyak lagi.

Pengembang memiliki hak akses penuh untuk menggunakan framework API yang sudah digunakan pada aplikasi inti. Arsitektur framework dirancang sedemikian sehingga memudahkan pengembang untuk menggunakan kembali komponen yang sudah dibuat. Setiap aplikasi yang dibuat dapat menggunakan kemampuan suatu framework secara bersamaan dengan framework yang lain (disesuaikan dengan tingkat keamanan yang diatur oleh framework). Jika ada komponen yang sama, maka komponen tersebut bisa ditimpa dari sisi user.

Setiap aplikasi yang nanti akan dibuat akan terdiri atas service dan sistem. Berikut adalah service dan sistem yang terdapat pada suatu aplikasi:

1. **View** -> mekanisme seorang developer untuk merancang tampilan dari suatu aplikasi.
2. **Content Provider** -> suatu mekanisme untuk mengakses database dalam pembangunan suatu aplikasi.
3. **Resource Manager** -> yang menyediakan akses ke kode non sumber lokal seperti string, gambar, dan tata letak file.
4. **Notification Manager** -> yang memungkinkan semua aplikasi menampilkan alert atau pemberitahuan yang bisa diatur pada status bar.

2.1.1.5 Aplikasi

Android telah menyediakan aplikasi inti seperti email client, sms, kalender, peta, browser, kontak, dan sebagainya. Semua aplikasi ditulis dengan bahasa pemrograman Java. Pada layer ini developer atau kita menempatkan aplikasi yang dibuat. Yang istimewa adalah pada Android semua aplikasi, baik aplikasi inti maupun aplikasi pihak ketiga, berjalan pada layer aplikasi dengan menggunakan library API yang sama. Ini berarti bahwa semua aplikasi yang dibuat untuk Android akan memiliki akses yang setara dalam mengakses seluruh kemampuan handset, tanpa membedakan apakah itu merupakan aplikasi inti atau aplikasi pihak ketiga. Dengan kata lain, dengan platform Android ini, developer secara penuh akan bisa mengatur perangkat Androidnya.

2.2 Eclipse

Eclipse adalah sebuah IDE (Integrated Development Environment) untuk mengembangkan perangkat lunak dan dapat dijalankan di berbagai platform (platform-independent). Berikut ini adalah sifat-sifat dari Eclipse:

- **Multi-platform**: Target sistem operasi Eclipse adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X.
- **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.

- **Multi-role**: Selain sebagai IDE untuk pengembangan aplikasi, Eclipse pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, test perangkat lunak, pengembangan web, dan lain sebagainya.

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:

- Core platform
- OSGi
- SWT (Standard Widget Toolkit)
- JFace
- Eclipse Workbench

Secara standar Eclipse selalu dilengkapi dengan JDT (Java Development Tools), plug-in yang membuat Eclipse kompatibel untuk mengembangkan program Java, dan PDE (Plug-in Development Environment) untuk mengembangkan plug-in baru. Eclipse beserta plug-in-nya diimplementasikan dalam bahasa pemrograman Java.

Konsep Eclipse adalah IDE yang terbuka (open), mudah diperluas (extensible) untuk apa saja, dan tidak untuk sesuatu yang spesifik. Jadi, Eclipse tidak saja untuk mengembangkan program Java, akan tetapi dapat digunakan untuk berbagai macam keperluan, cukup dengan menginstal plug-in yang

dibutuhkan. Apabila ingin mengembangkan program C/C++ terdapat plug-in CDT (C/C++ Development Tools). Selain itu, pengembangan secara visual bukan hal yang tidak mungkin oleh Eclipse, plug-in UML2 tersedia untuk membuat diagram UML. Dengan menggunakan PDE setiap orang bisa membuat plug-in sesuai dengan keinginannya. Salah satu situs yang menawarkan plug-in secara gratis adalah Eclipse downloads by project.

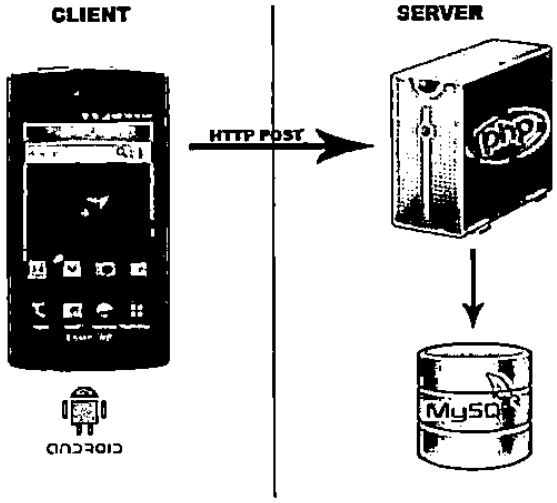
2.3 Web Service

Web Service adalah sekumpulan logika aplikasi beserta objek-objek dan metode-metode yang dimilikinya yang terletak di suatu server yang terhubung ke internet sehingga dapat diakses menggunakan protocol HTTP dan SOAP (Simple Object Access Protocol). Dalam penggunaannya, web service dapat digunakan dari hanya untuk memeriksa data user yang login ke sebuah website ataupun untuk digunakan pada transaksi perbankan on-line yang rumit.

Tujuan dari teknologi ini adalah untuk memudahkan beberapa aplikasi atau komponennya untuk saling berhubungan dengan aplikasi lain dalam sebuah organisasi maupun di luar organisasi menggunakan standar yang tidak terikat platform (platform-neutral) dan tidak terikat akan bahasa pemrograman yang digunakan (language-neutral). Hal tersebut dapat terjadi karena penggunaan XML standar yang didukung oleh banyak perusahaan besar di dunia, yang digunakan untuk bertukar data. Selain daripada itu, penggunaan SOAP menjadikan metode-metode dari objek-objek yang ada dalam sebuah web service

dapat diakses dari aplikasi lain seperti halnya aplikasi tersebut mengakses metode lokal.

Pada umumnya sebuah aplikasi memerlukan driver database agar bisa melakukan koneksi ke sebuah database. Web Server dapat dijadikan sebagai jembatan penghubung antara aplikasi dengan database. Jadi, dengan memanfaatkan Web Server sebuah aplikasi tidak lagi memerlukan driver database dan tidak perlu mengetahui database apa yang digunakan oleh server serta bagaimana struktur database tersebut. Aplikasi yang ingin mengakses cukup mengetahui metode atau fungsi apa saja yang disediakan oleh Web Server. Dengan memanfaatkan Web Server yang berbasis PHP, sebuah aplikasi android pun dapat mengakses database MySQL yang secara default tidak didukung oleh Android.



Gambar 2.2 Komunikasi Android dengan Web Server

2.4 Database

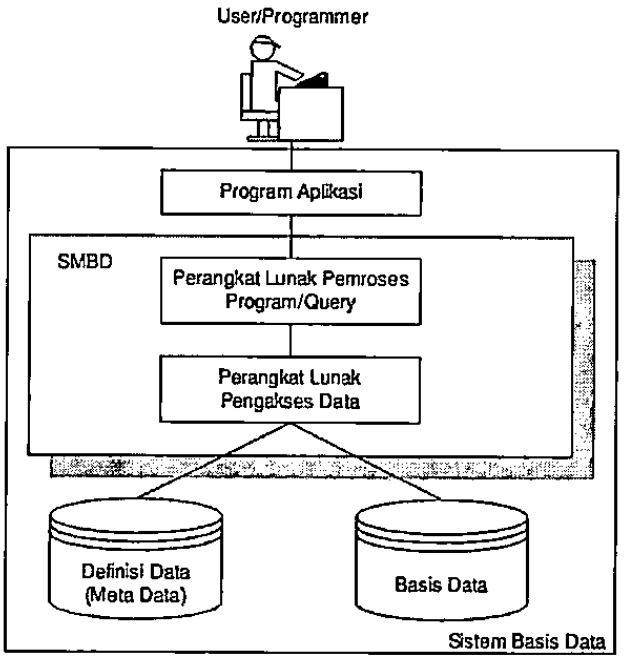
Database adalah kumpulan dari berbagai data yang saling berhubungan satu dengan yang lainnya. Database tersimpan di perangkat keras, serta dimanipulasi dengan menggunakan perangkat lunak. Pendefinisian database meliputi spesifikasi dari tipe data, struktur dan batasan dari data atau informasi yang akan disimpan. Database merupakan salah satu komponen yang penting dalam sistem informasi, karena merupakan basis dalam menyediakan informasi pada para pengguna atau user.

Penyusunan database meliputi proses memasukkan data ke dalam media penyimpanan data dan diatur dengan menggunakan perangkat Sistem Manajemen Database (Database Management System, DBMS). Manipulasi database meliputi pembuatan pernyataan (query) untuk mendapatkan informasi tertentu, melakukan pembaharuan atau penggantian (update) data, serta pembuatan report data.

Tujuan utama DBMS adalah untuk menyediakan tinjauan abstrak dari data bagi user. Jadi sistem menyembunyikan informasi mengenai bagaimana data disimpan dan dirawat, tetapi data tetap dapat diambil dengan efisien. Pertimbangan efisien yang digunakan adalah bagaimana merancang struktur data yang kompleks, tetapi tetap dapat digunakan oleh pengguna yang masih awam, tanpa mengetahui kompleksitas struktur data. Database menjadi penting karena munculnya beberapa masalah bila tidak menggunakan data yang terpusat, seperti adanya duplikasi data, hubungan antar data tidak jelas, organisasi data dan update

2.4.1 Sistem Database

Gabungan antara database dan perangkat lunak DBMS, termasuk di dalamnya program aplikasi yang dibuat dan bekerja dalam satu sistem, disebut dengan Sistem Database.



Gambar 2.3 Skema sistem database

2.4.2 Model Konseptual Database

Perancangan model konseptual database dalam sebuah organisasi menjadi tugas dari Administrator database. Model konseptual merupakan kombinasi beberapa cara untuk memproses data untuk beberapa aplikasi. Model konseptual tidak tergantung pada aplikasi individual, DBMS digunakan, Hardware komputer dan model fisiknya. Pada perancangan model konseptual database ini penekanan

dilakukan pada struktur data dan relasi antara file. Pada perancangan model konseptual ini dapat dilakukan dengan menggunakan model data relasional.

2.4.3 Tahapan Normalisasi

Proses normalisasi adalah proses pengelompokan data elemen menjadi tabel-tabel yang menunjukkan entity dan relasinya. Pada proses normalisasi dilakukan pengujian pada beberapa kondisi apakah ada kesulitan pada saat menambah/menyisipkan, menghapus, mengubah dan mengakses pada suatu database. Bila terdapat kesulitan pada pengujian tersebut maka perlu dipecahkan relasi pada beberapa tabel lagi atau dengan kata lain perancangan database belum optimal.

Tujuan dari normalisasi itu sendiri adalah untuk menghilangkan kerangkapan data, mengurangi kompleksitas, dan untuk mempermudah

Bentuk Tidak Normal

↓ Menghilangkan perulangan group

Bentuk Normal Pertama (1NF)

↓ Menghilangkan ketergantungan sebagian

Bentuk Normal Kedua (2NF)

↓ Menghilangkan ketergantungan transitif

Bentuk Normal Ketiga (3NF)

↓ Menghilangkan anomali-anomali hasil dari ketergantungan fungsional

Bentuk Normal Boyce-Codd (BCNF)

↓ Menghilangkan Ketergantungan Multivalue

Bentuk Normal Keempat (4NF)

↓ Menghilangkan anomali-anomali yang tersisa

Bentuk Normal Kelima

Gambar 2.4 Tahapan normalisasi database

2.4.3.1 Bentuk Normal Kesatu (1 NF/First Normal Form)

Bentuk Bentuk Normal Kesatu mempunyai ciri yaitu setiap data dibentuk dalam file flat, data dibentuk dalam satu record demi satu record dan nilai dari field berupa "atomic value". Tidak ada set atribut yang berulang ulang atau atribut bernilai ganda (multi value). Tiap field hanya satu pengertian, bukan merupakan

kumpulan data yang mempunyai arti mendua. Hanya satu arti saja dan juga bukanlah pecahan kata kata sehingga artinya lain.

2.4.3.2 Bentuk Normal Kedua (2NF)

Bentuk Normal kedua mempunyai syarat yaitu bentuk data telah memenuhi kriteria bentuk Normal Kesatu. Atribut bukan kunci haruslah bergantung secara fungsi pada kunci utama, sehingga untuk membentuk Normal Kedua haruslah sudah ditentukan kunci-kunci field. Kunci field harus unik dan dapat mewakili atribut lain yang menjadi anggotanya.

2.4.3.3 Bentuk Normal Ketiga (3NF)

Untuk menjadi bentuk Normal Ketiga maka relasi haruslah dalam bentuk Normal Kedua dan semua atribut bukan primer tidak punya hubungan yang transitif. Artinya setiap atribut bukan kunci harus bergantung hanya pada kunci primer