

BAB II

LANDASAN TEORI

2.1 Android

Android adalah sistem operasi yang berbasis Linux untuk telepon seluler seperti telepon pintar dan komputer tablet. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti 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.

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).

2.1.1 Keunggulan Ponsel/Tablet yang berbasis Android

- Multitasking

Dengan adanya fitur multitasking ini, kita dapat menjalankan beberapa aplikasi sekaligus seperti di komputer.

- Open Source

Android bersifat Open Source yang berarti aplikasi android bisa dibuat oleh siapapun yang mengerti dengan kode-kode sumber sehingga bisa membuat sebuah aplikasi versi pengembang aplikasi tersebut.

- Fitur Widget

Android memanjakan penggunanya dengan kebebasan membuat widget di homescreen untuk penggunanya. Sehingga bisa dengan mudah mengakses berbagai folder dan aplikasi dengan cepat dan mudah.

- Integrasi mudah dan cepat ke produk-produk Google

Karena Android adalah milik Mbah Google otomatis akses ke berbagai produk Google lainnya seperti Google Docs, Gmail, Google Reader, Blogger, dan banyak lainnya. akan lebih mudah dan cepat pastinya.

- Harga lebih variatif

Terakhir yang membuat Android unggul dan menjadi nomor 1 di dunia saat ini yaitu Android bisa dibeli mulai dari harga murah sampai yang mahal. Tidak seperti Apple dan Blackberry yang menjual produknya dengan harga mahal, Android bisa saya katakan lebih merakyat.

Mungkin dari kelebihan-kelebihan diatas yang sudah saya sebutkan masih ada lagi kelebihan lainnya. Apalagi sejak diumumkannya Android terbaru, yaitu Android 4.0 (Android Ice Cream Sandwich) banyak fitur baru yang ditawarkan, salah satunya Face Unlock.

2.1.2 Sejarah Android

Android Inc, adalah sebuah perusahaan software kecil yang didirikan pada bulan Oktober 2003 di Palo Alto, California, USA. Didirikan oleh beberapa senior di beberapa perusahaan yang berbasis IT & Communication; Andy Rubin, Rich Miner, Nick Sears dan Chris White. Menurut Rubin, Android Inc didirikan untuk mewujudkan mobile device yang lebih peka terhadap lokasi dan preferensi pemilik. Dengan kata lain, Android Inc, ingin mewujudkan mobile device yang lebih mengerti pemiliknya.

2.1.3 Google dan Sejarah Android

Konsep yang dimiliki *Android Inc*, ternyata menggugah minat raksasa Google untuk memilikinya. Pada bulan Agustus 2005, akhirnya Android Inc diakuisisi oleh Google Inc. Seluruh sahamnya dibeli oleh Google. Nilai pembelian *Android Inc* ini oleh google tidak ada release pastinya. Tetapi banyak yang memperkirakan nilai pembelian *Android Inc* oleh Google adalah sebesar USD 50 juta. Saat itu banyak yang berspekulasi, bahwa akuisisi ini adalah langkah awal yang dilakukan Google untuk masuk ke pasar *mobile phone*.

Andy Rubin, Rich Miner, Nick Sears dan Chris White tetap di *Android Inc* yang dibeli Google, sehingga akhirnya mereka semua menjadi bagian dari raksasa Google dan sejarah *android*. Saat itulah mereka mulai menggunakan *platfor linux* untuk membuat sistem operasi bagi *mobile phone*.

2.1.4 Sejarah Android dan Open Handset Alliance

Pada bulan nopember 2007, terbentuklan *Open Handset Alliance* yang merupakan konsorsium dari beberapa perusahaan : Broadcom Corporation, Google, HTC, Intel, LG, Marvell Technology Group, Motorola, Nvidia, Qualcomm, Samsung Electronics, Sprint Nextel, T-Mobile dan Texas Instruments. Mereka sepakat untuk membuat *open standart* bagi *mobile phone*. Pada hari yang sama, mereka mengumumkan produk pertama mereka, yaitu Android yang berbasis Linux kernel versi 2.6.

Bulan Desember 2008, bergabunglah 14 perusahaan lainnya yaitu : ARM Holdings, Atheros Communications, Asustek Computer Inc, Garmin Ltd, PacketVideo, Softbank, Sony Ericsson, Toshiba Corp dan Vodafone Group Plc. Hal ini merupakan langkah besar dalam sejarah Android untuk menjadi pemimpin dalam sistem operasi untuk *mobile phone*. Beberapa Versi release resmi Android :

Android	Cupcake	1.5
Android	Donut	1.6
Android	Eclair	2.0.x/2.1.x
Android	Froyo	2.2.x
Android	Gingerbread	2.3
Android	Gingerbread	2.3.3
Android	Honeycomb	3.0

2.2 Java sebagai *Application Development Interface*

Java adalah bahasa berorientasi objek yang dapat digunakan untuk pengembangan aplikasi mandiri, aplikasi berbasis internet, serta aplikasi untuk perangkat-perangkat cerdas yang dapat berkomunikasi lewat internet atau jaringan

komunikasi, Java bukan turunan langsung dari bahasa pemrograman manapun. Dalam Java ada 2 (dua) jenis program berbeda, yaitu aplikasi dan *applet*. Aplikasi adalah program yang biasanya disimpan dan dieksekusi dari komputer lokal sedangkan applet adalah program yang biasanya disimpan pada komputer yang jauh, yang dikoneksikan pemakai lewat *web browser*.

OOP (object oriented programming) adalah cara yang ampuh dalam pengorganisasian dan pengembangan perangkat lunak.

2.2.1 Syntax Java

2.2.1.1 Komentar

Ada beberapa bentuk komentar pada java, yaitu :

- Komentar baris tunggal dengan tanda //
- Komentar baris banyak diawali dengan tanda /* dan ditutup dengan */
- Komentar terdokumentasi digunakan komentar bertanda /** untuk menyatakan komentar dokumentasi, diakhiri dengan tanda */

2.2.1.2 Kata Kunci Simpanan (*Keywords*)

Kata kunci simpanan adalah *identifier* khusus yang disimpan oleh bahasa Java untuk mengendalikan bagaimana program didefinisikan. Kata tidak dapat digunakan sebagai *identifier* nama suatu variabel, class dan method.

Tabel 2.1 Daftar *Keywords* di Java

abstract	boolean	Break	byte	byvalue	Case
cast	catch	char	class	const	continue
default	do	double	Else	extends	False
final	finally	float	for	future	generic
goto	if	implements	import	inner	instanceof
Int	interface	long	native	new	Null

operator	outer	package	private	protected	Public
Rest	return	short	static	super	switch
synchronized	this	throw	throws	transient	True
Try	var	void	volatile	while	

2.2.1.3 Identifier

Digunakan untuk nama *class*, *method*, dan variabel. Dapat berupa urutan huruf (besar atau kecil), angka, garis bawah, dan tanda dolar. Tidak boleh diawali oleh angka dan bersifat *case sensitive*.

Dengan huruf awal kecil dan menandai bagian kata selanjutnya dengan huruf besar, misalnya `nextItem`, `currentValue`, `getTimeOfDay`.

2.2.1.4 Separator (Pemisah)

Tabel 2.2 Daftar *Separator*

Simbol	Nama	Fungsi
()	Kurung	Digunakan untuk menghimpun parameter dalam definisi dan pemanggilan <i>method</i> , juga di gunakan untuk menyatakan tingkatan pernyataan, menghimpun pernyataan untuk pengaturan alur program dan menyatakan tipe cast.
{ }	Kurung kurawal	Digunakan untuk menghimpun nilai yang otomatis dimasukkan kedalam array, juga digunakan untuk mendefinisikan blok program, untuk cakupan <i>class</i> , <i>method</i> , dan lokal.
[]	kurungsi ku	Digunakan untuk menyatakan tipe array, juga digunakan untuk membedakan nilai array.
;	titik-koma	Pemisah pernyataan.

,	Koma	Pemisah urutan <i>identifier</i> dalam deklarasi variabel, juga digunakan untuk mengaitkan pernyataan di dalam pernyataan <i>for</i> .
.	Titik	Digunakan untuk memisahkan nama paket dari sub-paket dan <i>class</i> , juga digunakan untuk memisahkan variabel atau <i>method</i> dari variabel referensi.

2.2.1.5 Variabel

Variabel adalah satuan dasar penyimpanan dalam program Java. Suatu variabel didefinisikan dengan kombinasi identifier, tipe, dan cakupan. Bergantung pada tempat kita mendeklarasikannya, variabel dapat bersifat lokal atau sementara.

2.2.1.6 Tipe Data

Java merupakan contoh bahasa yang *strongly typed language*. Hal ini berarti bahwa setiap variabel harus memiliki tipe yang sudah dideklarasikan. Tipe data pada Java : Integer, Floting point, Char, Boolean

2.2.1.7 Struktur Percabangan

Berikut akan dibahas mengenai bentuk umum dari *statement* kendali percabangan yang terdiri dari *switch case* dan *if-else if*, serta statement perulangan yaitu *for*, *while* dan *do-while*.

- Bentuk umum *switch case*:

```
switch (Variabel_Pembantu.getId()) {
case Kondisi_pertama:
Statement. . . . . ; break;
case Kondisi ke-n:
Statement. . . . . ; break; }
```
- Bentuk umum *if – else if*:

```
If (kondisi dengan nilai ekspresi boolean){ statement .....;}
Else if { statement ..... ;}
```

2.3 Android SDK(*software Development kit*)

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 *release* oleh Google. Saat ini disediakan Android SDK (*Software Development Kit*) sebagai alat bantu API untuk mulai mengembangkan Aplikasi pada *platform* Android menggunakan bahasa pemrograman java. Sebagai *platform* aplikasi netral, Android memberi kesempatan pada anda untuk membuat aplikasi yang kita butuhkan yang bukan merupakan aplikasi bawaan *Handphone/Smartphone*. Beberapa fitur-fitur Android yang paling penting adalah

- *Framework* yang mendukung penggantian komponen dan *reusable*.
- Mesin *Virtual Dalvik* dioptimalkan untuk perangkat *mobile*
- *Integerated browser* berdasarkan *engine open source Webkit*
- Grafis yang dioptimalkan dan didukung oleh *libraries* grafis 2D, garfis 3D berdasarkan *spesifikasi* opengl ES 1,0(Opsional akselerasi *hardware*)
- SQLite untuk penyimpanan data (database)
- Media support yang mendukung audio, video, gambar
- Bluetooth, EDGE, 3G,dan WiFi (tergantung *hardware*)
- Kamera, GPS, Kompas, dan accelerometer (tergantung *hardware*)
- Lingkungan *Development* yang lengkap dan kaya termasuk perangkat emulator, *tools* untuk *debugging*, profil dan kinerja memori, dan *plugin* untuk IDE Eclipse.

2.4 Eclipse sebagai IDE untuk pengembangan java

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

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

- *Multilanguage*: 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.

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*.

2.4.1 Sejarah Eclipse

Eclipse awalnya dikembangkan oleh IBM untuk menggantikan perangkat lunak IBM Visual Age for Java 4.0. Produk ini diluncurkan oleh IBM pada tanggal 5 November 2001, yang menginvestasikan sebanyak US\$ 40 juta^[1] untuk pengembangannya. Semenjak itu konsorsium *Eclipse Foundation* mengambil alih untuk pengembangan Eclipse lebih lanjut dan pengaturan organisasinya.

2.4.2 Arsitektur

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 meng-*instal plug-in* yang dibutuhkan. Apabila ingin mengembangkan program C/C++ terdapat plug-inCDT (C/C++ *Development Tools*). Selain itu, pengembangan secara visual bukan hal yang tidak mungkin oleh Eclipse, plug-inUML2 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 seperti *Eclipse downloads by project*.

2.5 ADT(*Android Development Tolls*)

ADT adalah singkatan dari **Android Development Tools** yaitu sebuah plugin untuk program eclipse, yang menjadi sebuah penghubung diantara eclipse dengan **Android SDK** agar dapat membuat sebuah aplikasi berbasis android.

AndroidDevelopment Tools (ADT) adalah plugin untuk IDE eclipse yang didesign powerfull untuk pengembangan aplikasi Android.

Developing Android di eclipse sangat direkomendasikan karena banyak kemudahan2 sebagai *tools* terintegrasi seperti custom XML editor, *debuging* dan banyak hal2 lain yang mempercepat pembuatan aplikasi.

2.6 Cloud Computing

Cloud Computing atau komputasi awan adalah gabungan pemanfaatan teknologi komputer ('komputasi') dan pengembangan berbasis Internet ('awan'). Awan (cloud) adalah metafora dari internet, sebagaimana awan yang sering digambarkan di diagram jaringan komputer. Sebagaimana awan dalam diagram jaringan komputer tersebut, awan (cloud) dalam Cloud Computing juga merupakan abstraksi dari infrastruktur kompleks yang disembunyikannya. Ia adalah suatu metoda komputasi di mana kapabilitas terkait teknologi informasi disajikan sebagai suatu layanan (as a service), sehingga pengguna dapat mengaksesnya lewat Internet ("di dalam awan") tanpa mengetahui apa yang ada didalamnya, ahli dengannya, atau memiliki kendali terhadap infrastruktur teknologi yang membantunya. Menurut sebuah makalah tahun 2008 yang dipublikasi IEEE Internet Computing "Cloud Computing adalah suatu paradigma di mana informasi secara permanen tersimpan di server di internet dan tersimpan secara sementara di komputer pengguna (client) termasuk di dalamnya adalah desktop, komputer tablet, notebook, komputer tembok, handheld, sensor-sensor, monitor dan lain-lain."

Komputasi awan adalah suatu konsep umum yang mencakup SaaS, Web 2.0, dan tren teknologi terbaru lain yang dikenal luas, dengan tema umum berupa ketergantungan terhadap Internet untuk memberikan kebutuhan komputasi pengguna. Sebagai contoh, Google Apps menyediakan aplikasi bisnis umum secara daring yang diakses melalui suatu penjelajah web dengan perangkat lunak dan data yang tersimpan di server. Komputasi awan saat ini merupakan trend teknologi terbaru, dan contoh bentuk pengembangan dari teknologi Cloud Computing ini adalah iCloud

2.6.1 Platform as a Service (PaaS)

Platform as a Service adalah layanan yang menyediakan computing platform. Biasanya sudah terdapat sistem operasi, database, web server dan framework aplikasi agar dapat menjalankan aplikasi yang telah dibuat. Perusahaan yang menyediakan layanan tersebutlah yang bertanggung jawab dalam pemeliharaan computing platform ini. Keuntungan layanan PaaS ini bagi pengembang adalah mereka bisa fokus pada aplikasi yang mereka buat tanpa memikirkan tentang pemeliharaan dari computing platform. Contoh penyedia layanan PaaS adalah Amazon Web Service dan Windows Azure.

2.7 Struktur Navigasi

Struktur navigasi adalah struktur atau alur dari suatu program. Menentukan struktur navigasi merupakan hal yang sebaiknya dilakukan sebelum membuat suatu aplikasi. Ada empat macam bentuk dasar dari struktur navigasi yang biasa digunakan dalam proses pembuatan aplikasi, yaitu :

- Struktur Navigasi linier

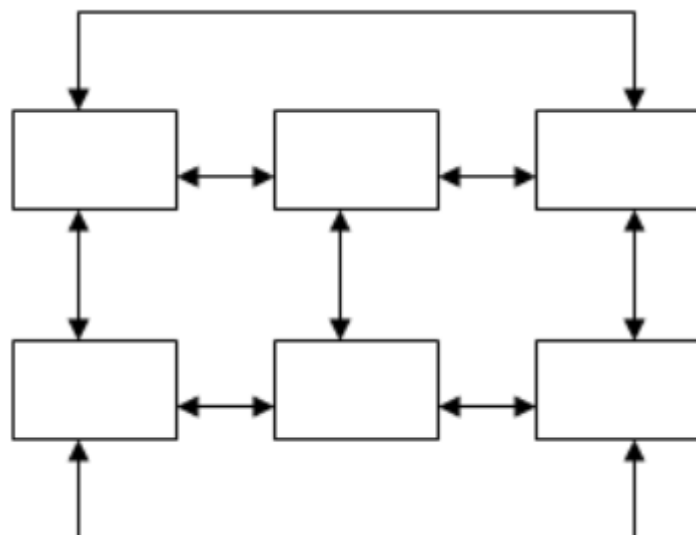
Struktur navigasi linier hanya mempunyai satu rangkaian cerita yang berurut, yang menampilkan satu demi satu tampilan layar secara berurut menurut urutannya. Tampilan yang dapat ditampilkan pada struktur jenis ini adalah satu halaman sebelumnya atau satu halaman sesudahnya, tidak dapat dua halaman sebelumnya atau dua halaman sesudahnya. Jadi, pemakai menelusuri program secara berurutan.



Gambar 2.1 Struktur Navigasi Linier

- Struktur Navigasi Nonlinier

Struktur navigasi nonlinier atau struktur tidak berurut merupakan pengembangan dari struktur navigasi linier. Pada struktur ini diperkenankan membuat navigasi bercabang. Percabangan yang dibuat pada struktur nonlinier ini berbeda dengan percabangan pada struktur hirarki karena pada percabangan nonlinier ini walaupun terdapat percabangan, tetapi tiap-tiap tampilan mempunyai kedudukan yang sama yaitu tidak ada Master Page dan Slave Page. Jadi, pemakai bebas menelusuri program tanpa dibatasi oleh suatu rute.

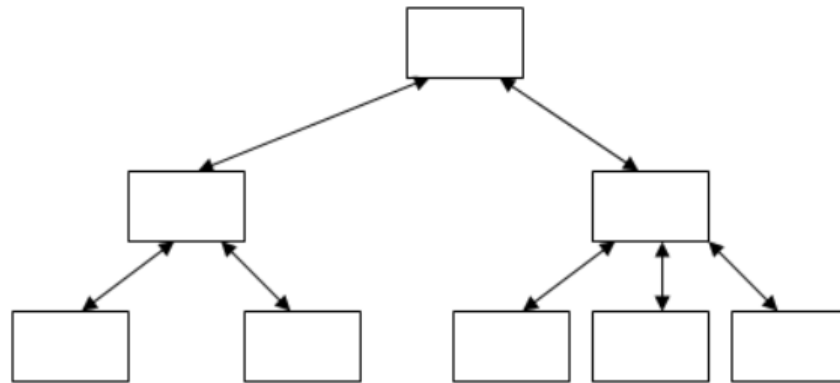


Gambar 2.2 Struktur Navigasi Non Linier

- Struktur Navigasi Hirarki

Struktur navigasi hirarki biasa disebut struktur bercabang. Struktur navigasi hirarki merupakan suatu struktur yang mengandalkan percabangan untuk menampilkan data berdasarkan kriteria tertentu. Tampilan pada menu pertama akan disebut sebagai Master Page (halaman utama pertama). Halaman utama ini mempunyai halaman percabangan yang disebut Slave Page (halaman pendukung). Jika salah satu halaman pendukung dipilih atau diaktifkan, maka tampilan tersebut akan bernama Master Page (halaman utama kedua), dan seterusnya.

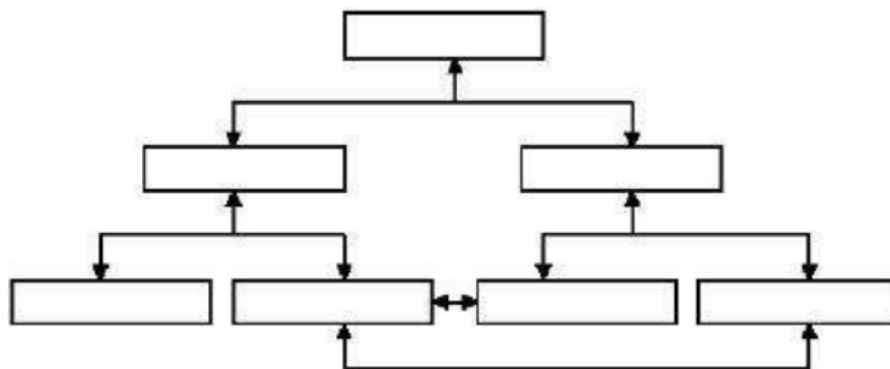
Pada struktur navigasi ini tidak diperkenankan adanya tampilan secara linier. Jadi, pemakai menelusuri program melalui titik-titik percabangan dari suatu struktur pohon.



Gambar 2.3 Struktur Navigasi Hirarki

- Struktur Navigasi Campuran

Struktur navigasi campuran merupakan gabungan dari ketiga struktur sebelumnya yaitu linier, non-linier dan hirarki. Struktur navigasi ini juga biasa disebut dengan struktur navigasi bebas. Struktur navigasi ini banyak digunakan dalam pembuatan website karena dapat memberikan keinteraksian yang lebih tinggi.



Gambar 2.4 Struktur Navigasi Campuran