

Pemrograman Mobile (Android)

3 SKS | Semester 7 | S1 Sistem Informasi | UNIKOM | 2015

Nizar Rabbi Radliya | nizar.radliya@yahoo.com

Nama Mahasiswa	
NIM	
Kelas	
<p style="text-align: right;">Kompetensi Dasar</p> <ol style="list-style-type: none"> Memahami tentang lingkungan pembangunan aplikasi android (perangkat bangun). Menerapkan pembangunan aplikasi android yang sederhana (<i>hello world</i>). 	
<p style="text-align: right;">Pokok Bahasan</p> <p>Pengenalan Pemrograman Android</p> <ol style="list-style-type: none"> Pengenalan android Instalasi perangkat pembangunan aplikasi android Membuat aplikasi sederhana 	

I. Pengenalan Android

Android adalah sistem operasi berbasis linux yang dirancang untuk perangkat mobile. Android awalnya dikembangkan oleh Android, Inc., dengan dukungan finansial dari Google, yang kemudian membelinya pada tahun 2005. Sistem operasi ini dirilis secara resmi pada tahun 2007, bersamaan dengan didirikannya *Open Handset Alliance*, konsorsium dari perusahaan-perusahaan perangkat keras, perangkat lunak, dan telekomunikasi yang bertujuan untuk memajukan standar terbuka perangkat seluler. Di dunia ini terdapat dua jenis distributor sistem operasi Android. Pertama yang mendapat dukungan penuh dari Google atau *Google Mail Service* (GMS) dan kedua adalah yang benar-benar bebas distribusinya tanpa dukungan langsung dari Google atau dikenal sebagai *Open Handset Distribution* (OHD).

Pada masa saat ini kebanyakan vendor-vendor *smartphone* sudah memproduksi *smartphone* berbasis android. Hal ini karena android itu adalah sistem operasi yang *open source* sehingga bebas didistribusikan dan dipakai oleh vendor manapun. Tidak hanya menjadi sistem operasi di *smartphone*, saat ini android menjadi pesaing Apple pada sistem operasi Tablet PC. Pesatnya pertumbuhan android adalah karena android itu sendiri merupakan platform yang lengkap, tersedianya tools pengembangan, adanya fasilitas market dan didukung oleh komunitas *open source*.

1.1. Arsitektur Android

Secara garis besar arsitektur android dapat digambarkan sebagai berikut:



Gambar 1. Arsitektur Android

1. Applications dan Widgets

Applications dan Widgets ini adalah layer dimana kita berhubungan dengan aplikasi saja. Aplikasi tersebut biasanya kita download, lalu melakukan instalasi dan menggunakan aplikasi yang sudah terinstal. Di layer terdapat aplikasi inti seperti klien email (Gmail), program SMS, kalender, peta (Google Map), browser (Google Chrome), kontak, android market (Google Play), pemutar video (Youtube), dan lain-lain.

Semua aplikasi dibangun di atas application layer atau pada library API yang sama. Application layer ini berjalan di dalam android runtime, menggunakan clas dan layanan yang dibuat dari application framework.

2. Application Framework

Android adalah *Open Development Platform* yaitu android menawarkan kepada pengembang untuk membangun aplikasi yang bagus dan inovatif. Pengembang bebas untuk mengakses perangkat keras, akses informasi *resources*, menjalankan *service background* dan sebagainya. Sehingga bisa kita simpulkan Application Framework ini

adalah layer yang dapat memungkinkan kita melakukan pengembangan/pembangunan aplikasi yang dapat dijalankan pada sistem operasi android.

Application framework menyediakan berbagai class yang digunakan untuk membuat aplikasi android. Termasuk juga berhubungan dengan akses ke hardware dan manajemen *user-interface* dan manajemen memori.

3. Libraries

Libraries ini adalah layer dimana fitur-fitur android berada. Para pembuat aplikasi dapat mengakses libraries untuk menjalankan aplikasinya. Berjalan di atas kernel yang didalamnya terdapat kumpulan library C dan C++ seperti Libc dan SSL, dan juga:

- a. Library media untuk pemutaran media audio dan video,
- b. Library untuk manajemen tampilan
- c. Library Graphics mencakup SGL dan OpenGL untuk grafis 2D dan 3D,
- d. Library SQLite untuk dukungan database,
- e. Library LiveWebcore mencakup modern web browser dengan engine embedded web view,
- f. Library SSL dan WebKit untuk integrasi web browser dan keamanan internet.

4. Android Runtime

Terdiri dari *Core Libraries* dan *Dalvik Virtual Machine (DVM)*. *Core Libraries* ini meliputi serangkaian library Java yang terdiri dari berbagai macam fungsi dasar pada bahasa pemrograman Java, sedangkan DVM merupakan *Java Virtual Machine* (semacam JVM pada Java ME) yang secara khusus dijalankan pada android. DVM dapat mengeksekusi file dengan format Dalvik Executable (*.dex) yang telah dioptimasi dengan menggunakan memori yang minimal. DVM telah di-register dan menjalankan berbagai class yang di-compile menggunakan compiler Java yang kemudian ditransformasi menjadi format *.dex dengan menggunakan 'dx' tool. DVM menggunakan kernel linux untuk menjalankan berbagai fungsi seperti threading dan manajemen memori.

5. Linux Kernel

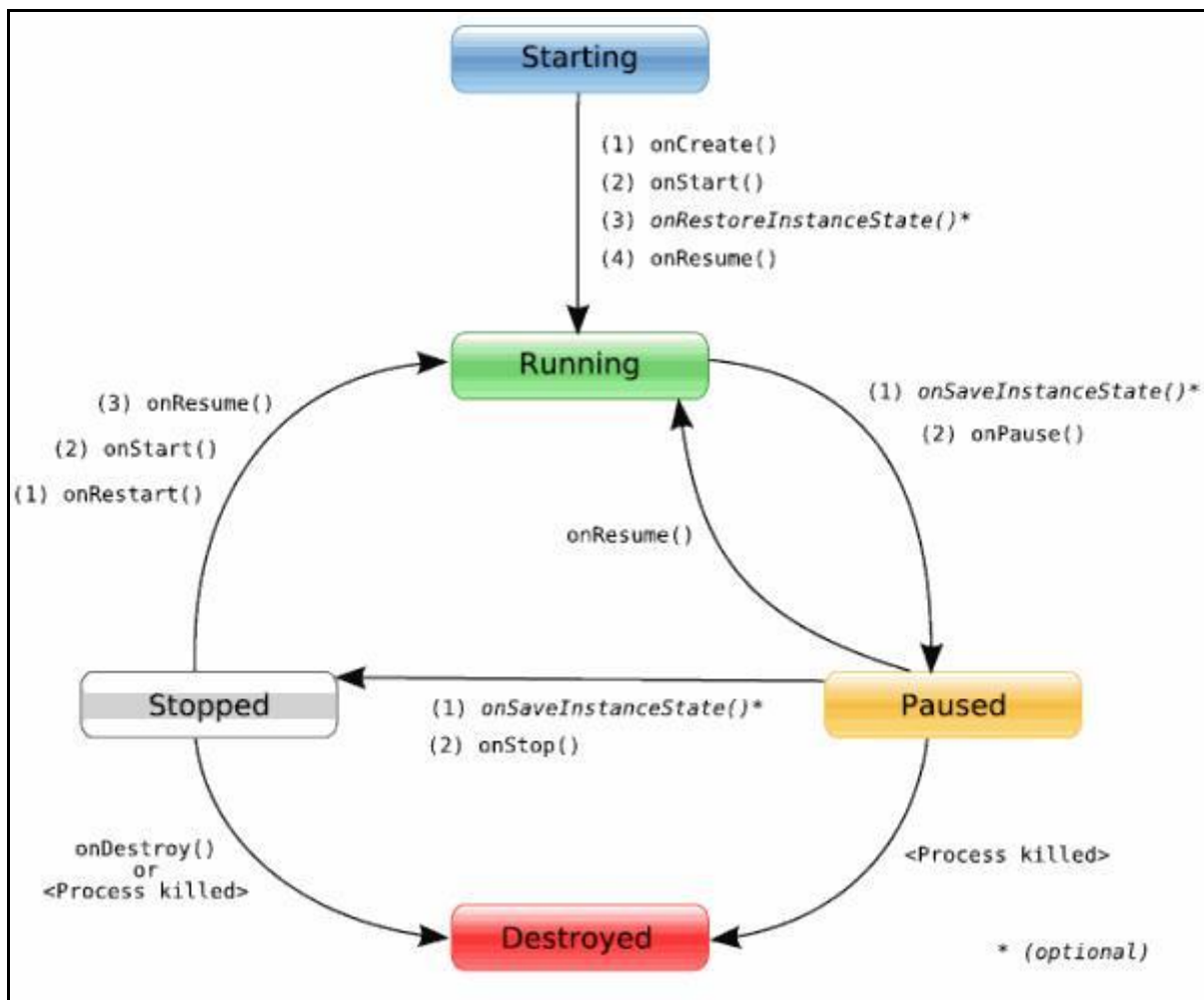
Linux kernel adalah layer dimana ini dari sistem operasi dari android itu berada. Berisi file-file sistem yang mengatur sistem processing, memory, resource, drivers, dan sistem-sistem lainnya. Linux kernel yang digunakan android adalah linux kernel 2.6 (www.kernel.org).

1.2. Komponen Android

Terdapat beberapa jenis komponen android diantaranya:

1. Activity

Setiap User Interface diwakili oleh kelas Activity (*activity class*). Setiap activity mempunyai siklus, dapat dilihat di gambar 2. Sebuah aplikasi dapat terdiri dari satu atau lebih activity yang diproses dalam Linux.



Gambar 2. Activity Lifecycle

Selama siklus ini berjalan, activity bisa mempunyai lebih dari 2 status seperti yang terlihat pada gambar 2. Kita tidak bisa mengontrol setiap status karena semuanya sudah ditangani oleh sistem. Namun kita akan mendapat pesan saat terjadi perubahan status melalui method onXX(). Berikut penjelasan setiap status.

- a. onCreate(Bundle) Dipanggil saat pertama kali aplikasi dijalankan. Kita dapat menggunakan ini untuk deklarasi variabel atau membuat user interface.
- b. onStart() Mengindikasikan activity yang ditampilkan ke pengguna (user).

- c. `onResume()` Dipanggil saat aplikasi kita mulai berinteraksi dengan pengguna. Disini sangat cocok untuk meletakkan animasi ataupun musik.
- d. `onPause()` Dipanggil saat aplikasi yang Kita jalankan kembali ke halaman sebelumnya atau biasanya karena ada activity baru yang dijalankan. Disini cocok untuk meletakkan algoritma penyimpanan (save).
- e. `onStop()` Dipanggil saat aplikasi Kita berjalan di belakang layar dalam waktu cukup lama.
- f. `onRestart()` Activity kembali menampilkan user interface setelah status stop.
- g. `onDestroy()` Dipanggil saat aplikasi benar-benar berhenti.
- h. `onSaveInstanceState(Bundle)` Method ini mengijinkan activity untuk menyimpan setiap status intance. Misalnya dalam mengedit teks, kursor bergerak dari kiri ke kanan.
- i. `onRestoreInstanceState (Bundle)` Diapanggil saat activity kembali menginisialisasi dari status sebelumnya yang disimpan oleh `onSaveInstanceState(Bundle)`.

2. Service

Service tidak memiliki user interface, namun berjalan di belakang layar. Misalnya music player, sebuah activity digunakan untuk memilih lagu kemudian diputar/dimainkan. Agar music player bisa berjalan dibelakang aplikasi lain maka harus menggunakan service.

3. Content Provider

Menyediakan cara untuk mengakses data yang dibutuhkan oleh suatu activity, misalnya kita menggunakan aplikasi berbasis peta (MAP). Activity membutuhkan cara untuk mengakses data kontak untuk prosedur navigasi dan disinilah peran content providers. Intinya content provider digunakan dalam aplikasi untuk sharing data (berhubungan dengan database).

4. Resource

Resource digunakan untuk menyimpan file-file non-coding yang diperlukan pada sebuah aplikasi misalnya file icon, file gambar, file audio, file video atau yang lain. Gambar berformat JPG atau PNG sebuah aplikasi biasanya disimpan dalam folder `res/drawable`, icon aplikasi disimpan dalam `res/drawable-ldpi` dan file audio disimpan dalam folder `res/raw`. File XML untuk membentuk sebuah user interface disimpan dalam folder `res/layout`.

5. Views

Digunakan untuk membangun antarmuka pengguna untuk komponen activity yang akan digunakan.

6. Notification

Digunakan untuk mekanisme sinyal ke pengguna secara insidental sesuai dengan waktu yang telah ditentukan. Contohnya pemberitahuan dalam bentuk getar atau lampu ketika zona waktu berubah, baterai low dan lain-lain.

1.3. Versi Android

Pengembangan berbagai versi android yang telah dirilis dipasaran, terlihat seperti pada tabel 1 di bawah ini.

Tabel 1. Versi Android

Versi	Nama Kode
Android 1.0	
Android 1.1	
Android 1.5	Cupcake
Android 1.6	Donut
Android 2.0	Eclair
Android 2.0.1	Eclair
Android 2.1	Eclair
Android 2.2–2.2.3	Froyo
Android 2.3–2.3.2	Gingerbread
Android 2.3.3–2.3.7	Gingerbread
Android 3.0	Honeycomb
Android 3.1	Honeycomb
Android 3.2	Honeycomb
Android 4.0–4.0.2	Ice Cream Sandwich
Android 4.0.3–4.0.4	Ice Cream Sandwich
Android 4.1	Jelly Bean
Android 4.2	Jelly Bean
Android 4.3	Jelly Bean
Android 4.4	KitKat
Android 5.0	Lollipop
Android 5.1	Lollipop
Android 6.0	Marshmallow

II. Instalasi Perangkat Pembangunan Aplikasi Android

Pada bab ini kita akan mempersiapkan perangkat yang diperlukan untuk membangun aplikasi android. Ada beberapa perangkat dasar yang harus kita instalasi untuk dapat membangun aplikasi android, diantaranya:

1. Java JDK

2. Android SDK
3. IDE Eclipse
4. Android ADT

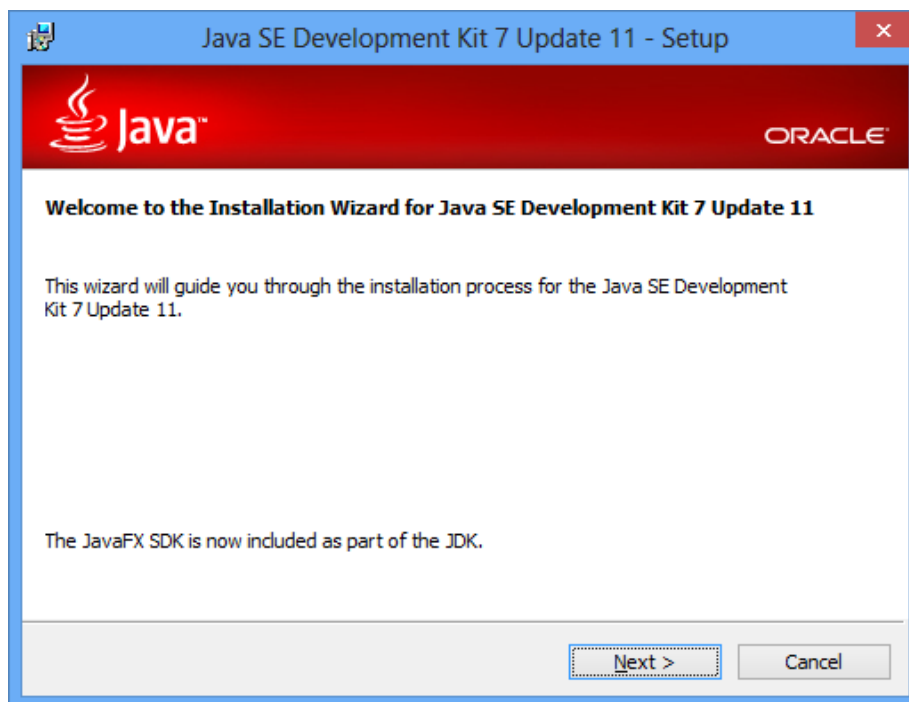
Sekarang kita akan melakukan instalasi semua paket diatas secara berurutan.

2.1. Instalasi Java JDK

JDK (*Java Development Kit*) adalah Perangkat lunak yang digunakan untuk melakukan proses kompilasi dari kode java ke bytecode yang dapat dimengerti dan dapat dijalankan oleh JRE (*Java Runtime Envirotment*). JDK wajib terinstall pada komputer yang akan melakukan proses pembuatan aplikasi berbasis java, namun tidak wajib terinstall di komputer yang akan menjalankan aplikasi yang dibangun dengan java.

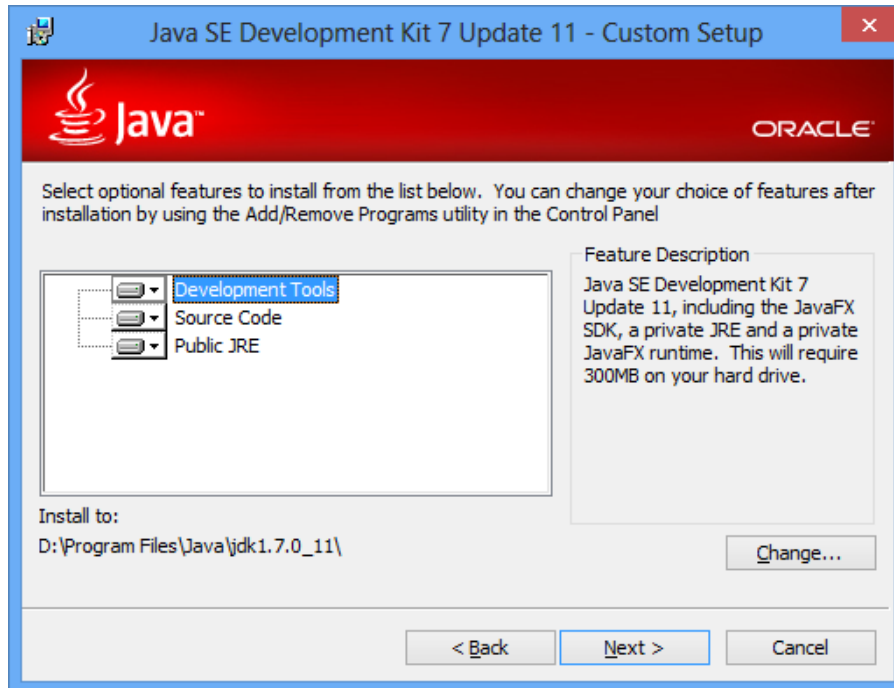
JRE adalah perangkat lunak yang digunakan untuk menjalankan aplikasi yang dibangun menggunakan java. Versi JRE harus sama atau lebih tinggi dengan JDK yang digunakan untuk membangun aplikasi java.

1. Download terlebih dahulu paket Java JDK di alamat situs: <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>
2. Jalankan file excutable Java JDK yang sudah kita download, maka pertama kali akan tampil halaman seperti pada gambar 3 di bawah ini.



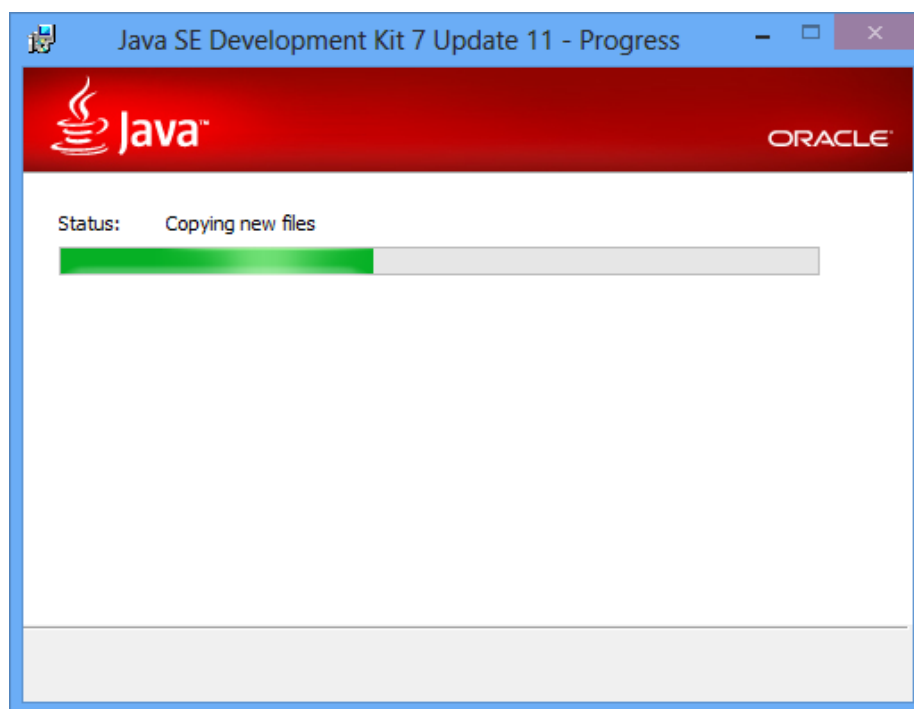
Gambar 3. Kotak dialog instalasi Paket Java SE

3. Klik button Next sehingga akan tampil deskripsi fitur yang akan diinstal seperti pada gambar 4 di bawah ini. Anda bias memilih opsi mana saja yang akan diinstal. Secara default, lokasi instalasi adalah dalam folder Program Files. Apabila kita akan mengubah lokasi instalasi, klik button Change dan pilih folder yang akan digunakan.



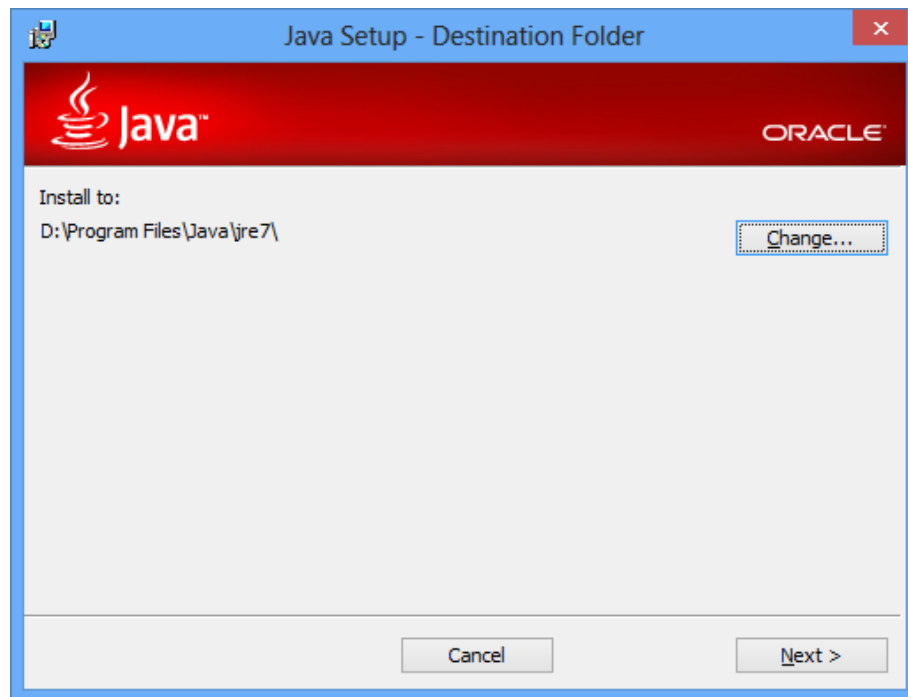
Gambar 4. Deskripsi Fitur dan Lokasi Instalasi

4. Klik button Next untuk menuju proses selanjutnya yang tampilannya seperti pada gambar 5 di bawah ini.



Gambar 5. Proses Ekstraksi

- Selanjutnya akan tampil kotak dialog untuk instal JRE seperti pada gambar 6 di bawah ini. Anda bisa mengubah lokasi penyimpanan pada folder lain dengan cara klik button Change dan pilih folder yang akan digunakan.



Gambar 6. Lokasi Instalasi JRE

- Klik button Next untuk menyelesaikan proses instalasi yang tampilannya seperti pada gambar 7 di bawah ini.



Gambar 7. Proses Instalasi

7. Apabila proses instalasi berhasil maka akan tampil halaman seperti pada gambar 8 di bawah ini.



Gambar 8. Instalasi Selesai

Untuk dapat mengoperasikan JDK dari folder mana saja, kita perlu melakukan setting alamat path-nya (untuk caranya silahkan cari melalui internet).

2.2. Instalasi Android SDK

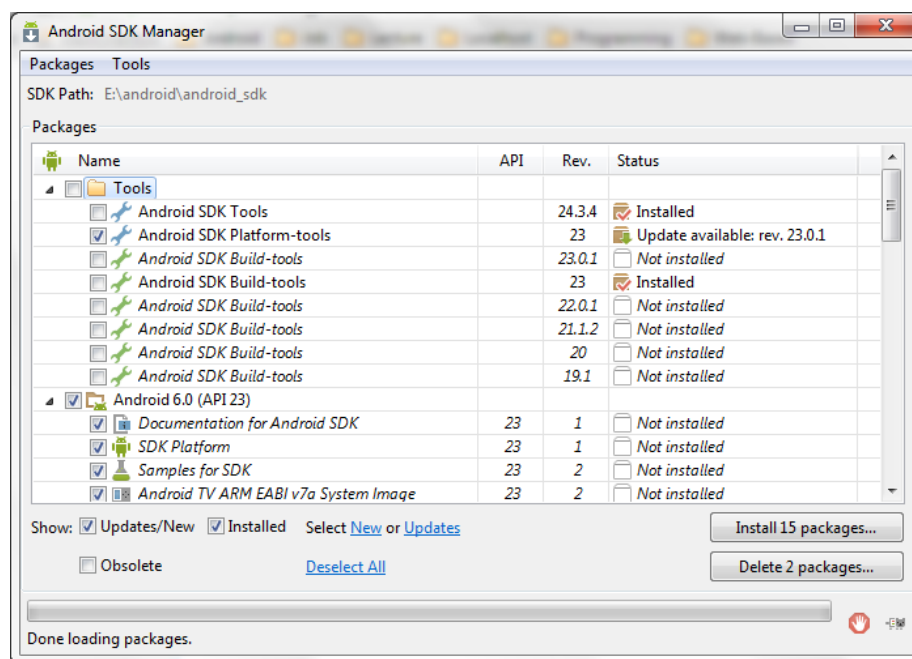
Android SDK (*Software Development Kit*) tools API (Application Programming Interface) yang digunakan untuk mulai mengembangkan aplikasi pada platform android menggunakan bahasa pemrograman Java. Android merupakan subset perangkat lunak untuk ponsel yang meliputi sistem operasi, middleware dan aplikasi kunci yang di release oleh Google. Saat ini disediakan android SDK sebagai alat bantu dan API untuk mulai mengembangkan aplikasi pada platform android menggunakan bahasa pemrograman Java. Sebagai platform aplikasi netral, android memberi anda kesempatan untuk membuat aplikasi yang kita butuhkan yang bukan merupakan aplikasi bawaan handphone/smartphone. Beberapa fitur android yang paling penting adalah :

1. Framework aplikasi yang mendukung penggantian komponen dan reusable.
2. Mesin Virtual Dalvik dioptimalkan untuk perangkat mobile.
3. Integrated browser berdasarkan engine open source webkit.
4. Grafis yang dioptimalkan dan didukung oleh libraries grafis 2D, grafis 3D berdasarkan spesifikasi openGL ES 1.0 (Opsional Akselerasi Hardware).

5. SQLite untuk penyimpanan data.
6. Media Support yang mendukung audio, video dan gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF), GSM Telephony (tergantung hardware).
7. Bluetooth, EDGE, 3G, WiFi (tergantung hardware).
8. Kamera, GPS, kompas dan Accelerator (tergantung hardware).
9. Lingkungan development yang lengkap dan kaya, termasuk perangkat emulator, tools untuk debugging, profil dan kinerja memori, dan plug in untuk IDE Eclipse.

Untuk cara instalasinya adalah sebagai berikut:

1. Download terlebih dahulu paket Android SDK di alamat situs: <https://developer.android.com/sdk/index.html>. Tersedia untuk sistem operasi Windows, Linux, Mac OS.
2. Jalankan file executable Android SDK yang sudah kita download, lalu ikuti proses instalasi sampai selesai. Setelah proses instalasi selesai maka akan tampil halaman Android SDK seperti pada gambar 9 di bawah ini.



Gambar 9. Android SDK

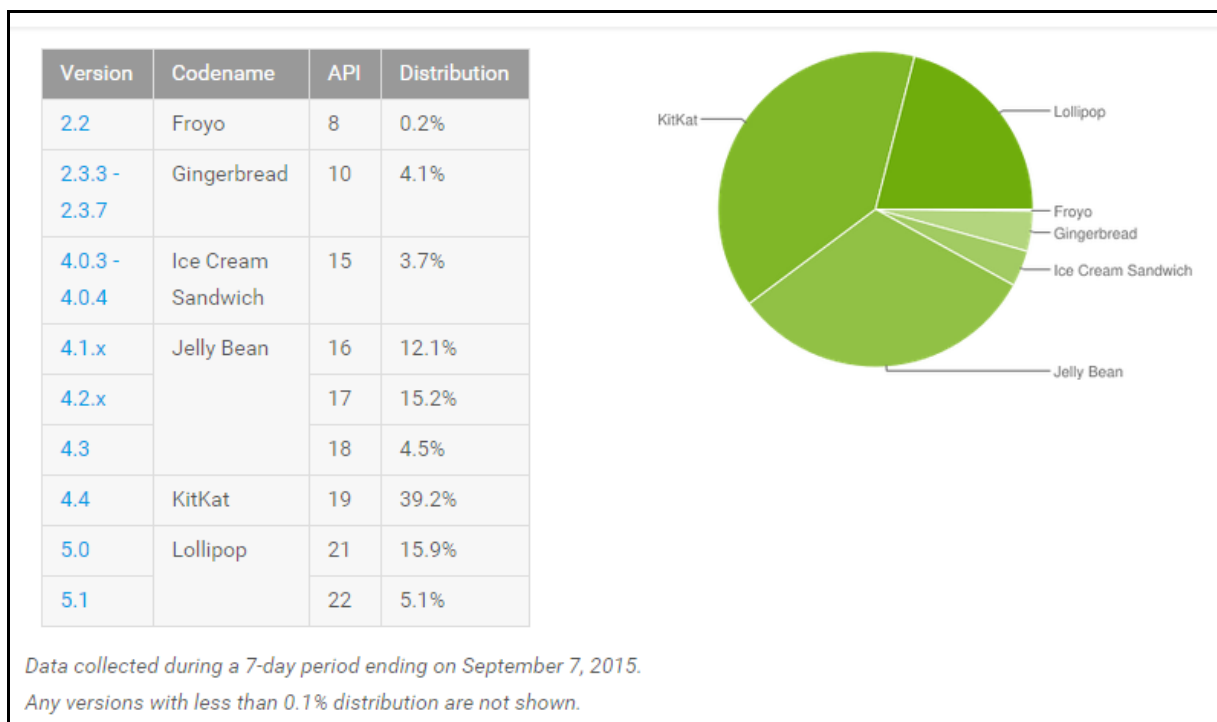
3. Saat ini versi android atau level API yang terbaru adalah Android 6.0 (API 23). Untuk informasi API setiap versi android dapat kita lihat pada tabel 2 di bawah ini.

Tabel 2. Level API Android

Versi	Nama Kode	Level API
Android 1.0		API level 1
Android 1.1		API level 2
Android 1.5	Cupcake	API level 3

Android 1.6	Donut	API level 4
Android 2.0	Eclair	API level 5
Android 2.0.1	Eclair	API level 6
Android 2.1	Eclair	API level 7
Android 2.2–2.2.3	Froyo	API level 8
Android 2.3–2.3.2	Gingerbread	API level 9
Android 2.3.3–2.3.7	Gingerbread	API level 10
Android 3.0	Honeycomb	API level 11
Android 3.1	Honeycomb	API level 12
Android 3.2	Honeycomb	API level 13
Android 4.0–4.0.2	Ice Cream Sandwich	API level 14
Android 4.0.3–4.0.4	Ice Cream Sandwich	API level 15
Android 4.1	Jelly Bean	API level 16
Android 4.2	Jelly Bean	API level 17
Android 4.3	Jelly Bean	API level 18
Android 4.4	KitKat	API level 19
Android 5.0	Lollipop	API level 21
Android 5.1	Lollipop	API level 22
Android 6.0	Marshmallow	API level 23

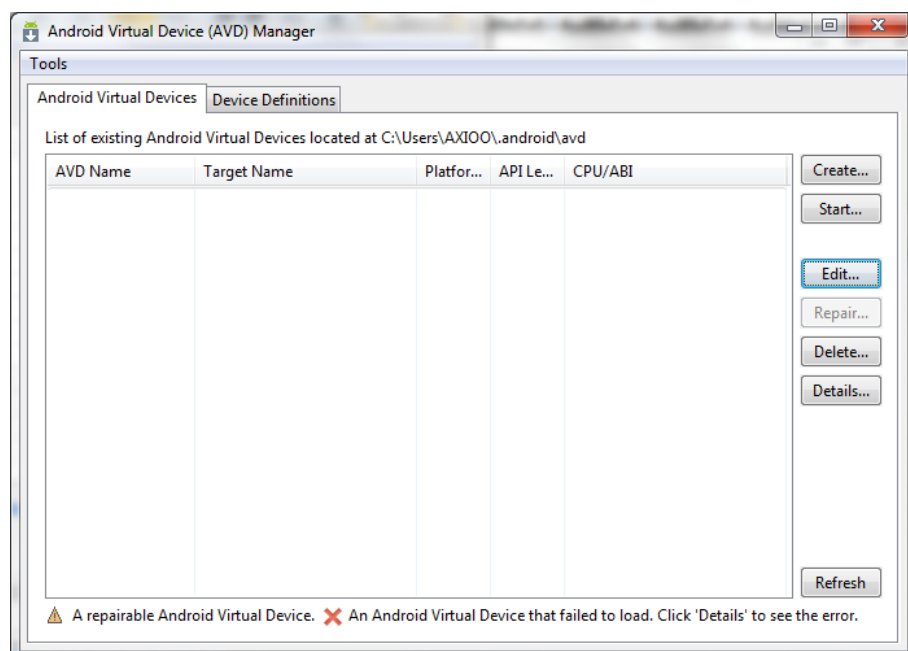
4. Kita dapat memilih versi android yang akan digunakan. Sebagai pertimbangan dalam memilih versi, kita dapat melihat penggunaan versi android yang ada di pasaran. Untuk melihat persentase penggunaan versi android, dapat membuka alamat website: <https://developer.android.com/about/dashboards/index.html>.



Gambar 10. Persentase Penggunaan Versi Android

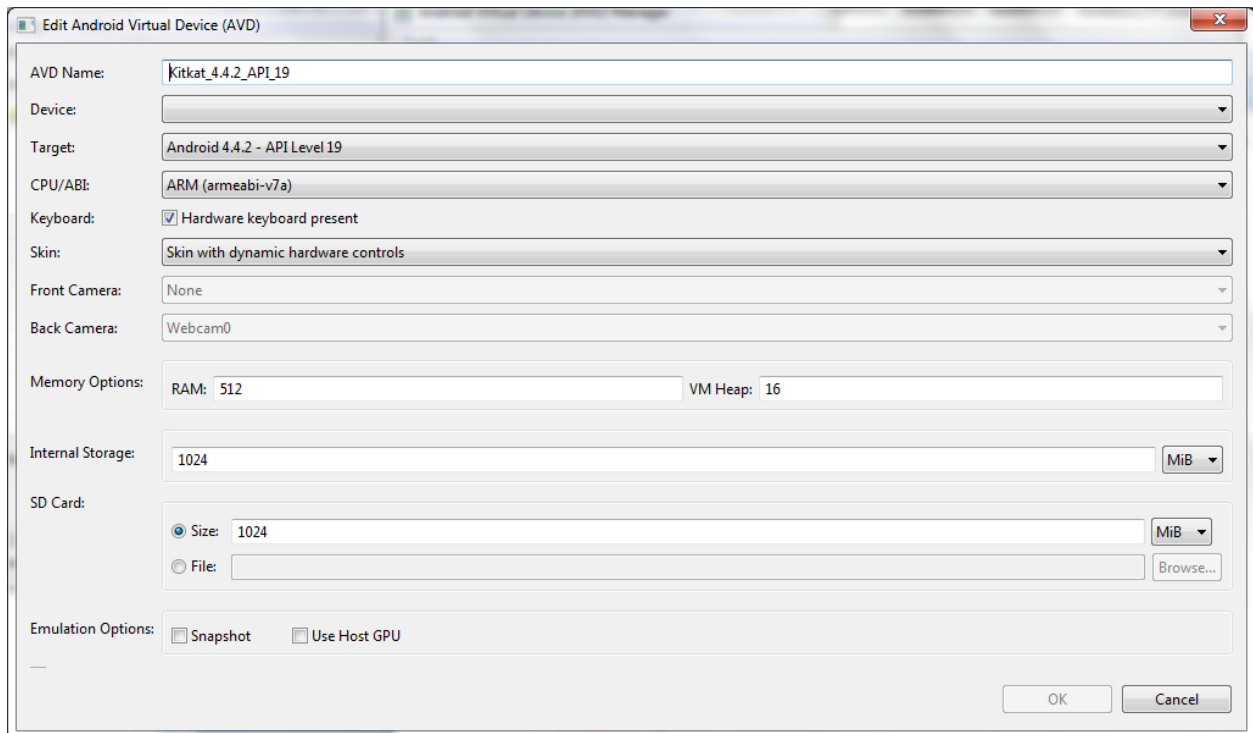
Pada gambar 10 di atas, kita melihat versi android yang paling banyak digunakan saat ini adalah Android 4.4 KitKat (API 19).

5. Saat ini kita akan menggunakan versi Android 4.4.2 (API 19). Pada kotak dialog Android SDK seperti pada gambar 9, kita centang pada bagian Android 4.4.2 (API 19).
6. Selanjutnya klik pada bagian Extras dan centang pada bagian Google USB Driver. Pilihan ini diperlukan ketika kita menggunakan perangkat android sebenarnya, tinggal menyambungkan USB computer dan tidak perlu melakukan instalasi driver, karena akan terdeteksi secara otomatis. Lalu klik button Install ... packages.
7. Ketika proses instalasi selesai maka akan terlihat status paket yang ada telah terinstal (Installed) pada kolom Status. Untuk dapat mengoperasikan Android SDK dari folder mana saja, kita perlu melakukan setting alamat path-nya (untuk caranya silahkan cari melalui internet).
8. Selanjutnya kita akan membuat AVD pada aplikasi Android SDK yang sudah terinstal. AVD (Android Virtual Device) merupakan simulator untuk mengoperasikan aplikasi android yang nanti akan kita bangun. Pada halaman Android SDK seperti pada gambar 9, kita pilih menu Tools lalu pilih Manage AVDs. Maka akan tampil halaman seperti pada gambar 11 di bawah ini.



Gambar 11. AVD Manager

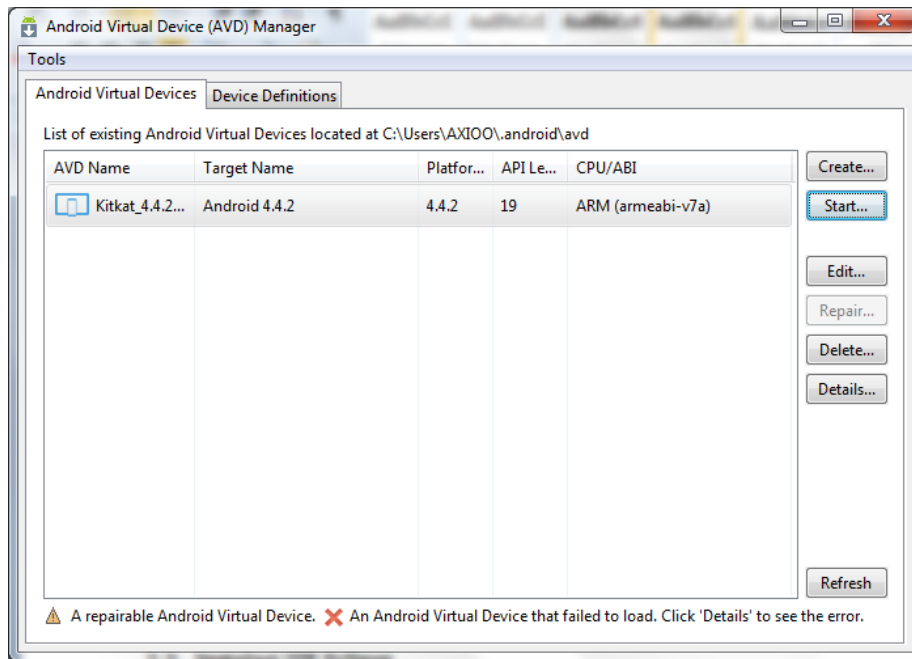
9. Lalu tekan button Create, dan akan tampil halaman seperti pada gambar 12.



Gambar 12. Create AVD

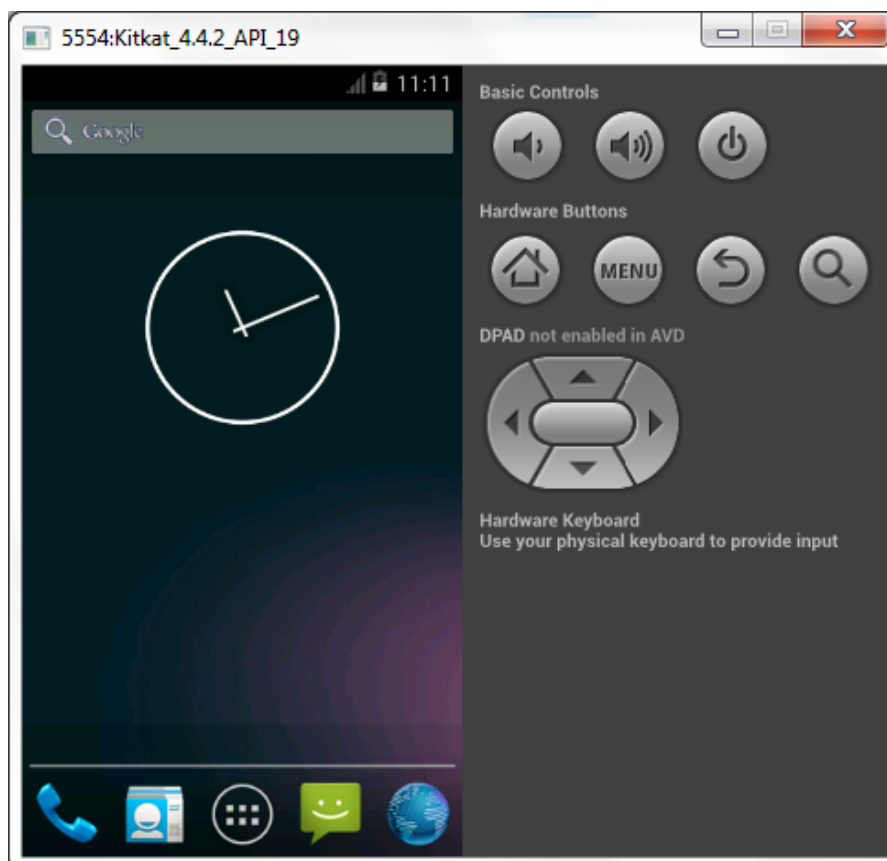
Keterangan:

- a. AVD Name : nama AVD/emulator yang akan kita buat.
 - b. Device : model device android yang akan kita buat
 - c. Target : versi android yang hendak kita gunakan.
 - d. Keyboard : konfigurasi apakah kita akan menggunakan keyboard atau tidak pada emulator yang akan kita buat.
 - e. Back Camera : konfigurasi apakah kita akan menggunakan camera atau tidak pada emulator yang akan kita buat.
 - f. Internal Storage : kapasitas memori internal pada emulator kita.
 - g. SD Card: kapasitas kartu memori pada emulator kita.
10. Setelah selesai menekan button OK maka akan muncul halaman seperti pada gambar 13 di bawah ini.



Gambar 13. AVD Berhasil Dibuat

11. Untuk menjalankan emulator, kita tekan button Start.



Gambar 14. Emulator Android

Saat pertama kali dijalankan memang emulator akan loading dengan waktu yang cukup lama. Untuk itu jika emulator sudah tampil jangan di tutup dulu, karena emulator akan loading dari awal lagi.

2.3. Instalasi IDE Eclipse

Eclipse adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (platform-independent). Eclipse dikembangkan dengan bahasa pemrograman Java, akan tetapi Eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti C/C++, Cobol, Python, Perl, PHP, dan termasuk Android. 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.

1. Download terlebih dahulu paket IDE Eclipse di alamat situs: <https://eclipse.org/downloads/index-developer.php>
2. Setelah download selesai, ekstrak hasil unduhan ke folder lokal computer kita. Lalu jalankan file executable Eclipse dan nanti kita akan meilih folder sebagai folder kerja.

2.4. Instalasi Android ADT

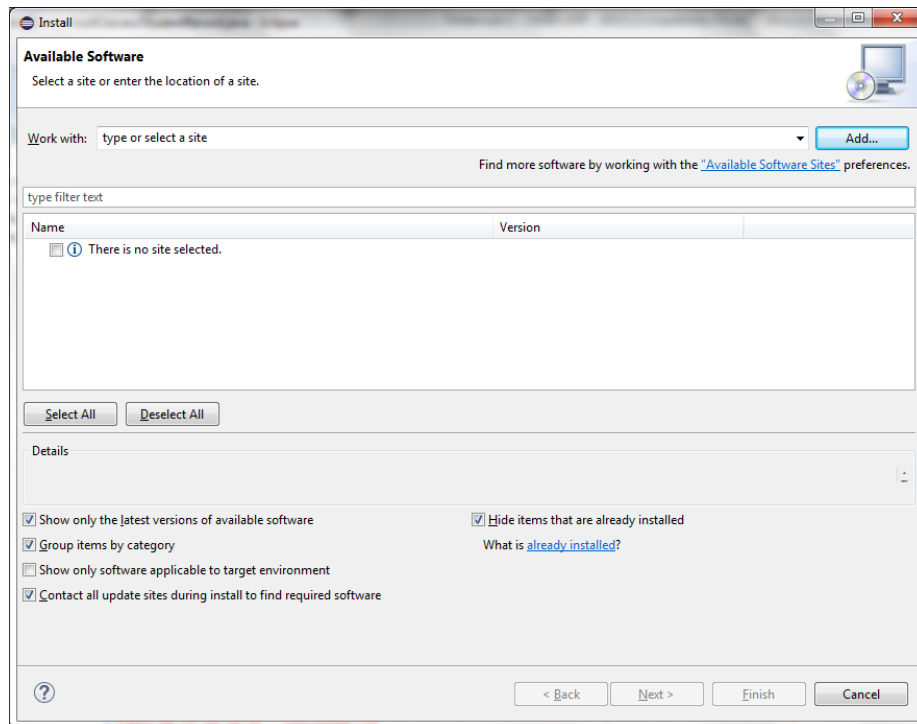
Android Development Tools (ADT) adalah plugin yang didesain untuk IDE Eclipse yang memberikan kita kemudahan dalam mengembangkan aplikasi android. Contoh kemudahan tersebut adalah dalam membuat GUI aplikasi, dan menambahkan komponen-kompenen yang lainnya. Dengan ADT juga kita dapat melakukan pembuatan package android (.apk) yang digunakan untuk distribusi aplikasi android yang kita rancang.

Semakin tinggi platform android yang kita gunakan, dianjurkan menggunakan ADT yang lebih terbaru, karena biasanya munculnya platform baru diikuti oleh munculnya versi ADT yang terbaru. Untuk melakukan instalasi ADT di-eclipse dapat dilakukan secara online maupun offline.

Developing Android di eclipse sangat direkomendasikan karena banyak kemudahan-kemudahan sebagai tools terintegrasi seperti custom XML editor, debugging

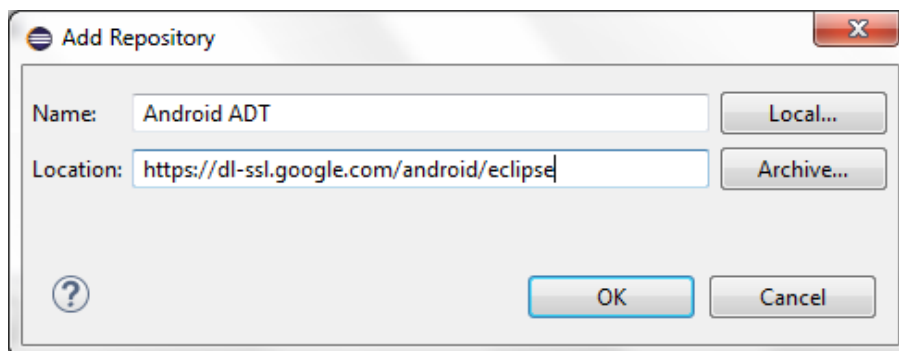
dan banyak hal-hal lain yang mempercepat pembuatan aplikasi. Berikut tahapan-tahapan instalasi ADT plugin ke eclipse IDE:

1. Sebelum melakukan instalasi ADT, persiapkan terlebih dahulu versi eclipse yang kompetibel. Sebaiknya download the latest eclipse yang terdapat pada websitenya. Perlu dicatat sebelum intalisasi ADT, Android SDK starter package harus terlebih dahulu di install.
2. Download ADT plugin dari eclipse menu Help>Install new software.



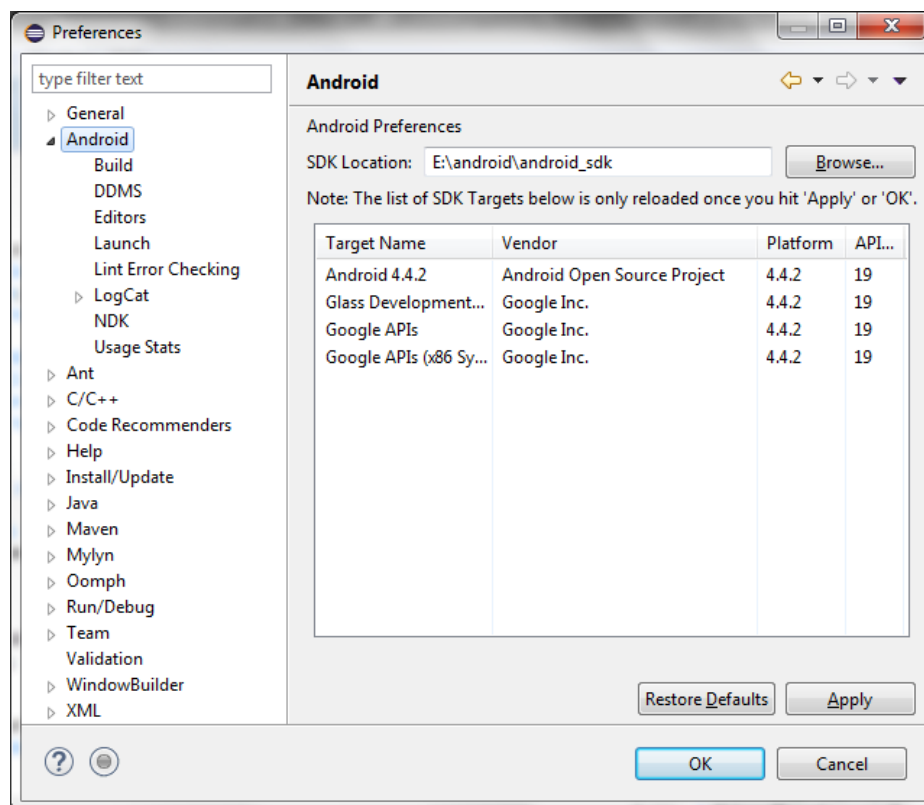
Gambar 15. Install New Software di Eclipse

3. Pada available software dialog, click Add button. Pada Add site dialog masukkan Nama (bebas, contohnya kita kasih nama Android) dan pada Location field masukkan link <https://dl-ssl.google.com/android/eclipse/>. Kalau terjadi error untuk koneksinya, silahkan diubah disetting tab dan memilih http connection.



Gambar 16. Install ADT Secara Online

4. Setelah tersambung dan terinstall, maka kembali ke dialog window pertama (Install new software), pilih work with field yaitu Android, dan pada list akan terlihat nested tools android DDMS dan Android development tools. Check semua pilihan nested tersebut, kemudian click tombol Next sampai finish. Kemudian restart Eclipse.
5. Perlu disetting terlebih dahulu untuk mereferensikan Android SDK directory. Dari menu Windows>preferences, select Android pada panel sebelah kiri. Pada SDK Location di panel utama, browse dan locate SDK directory.

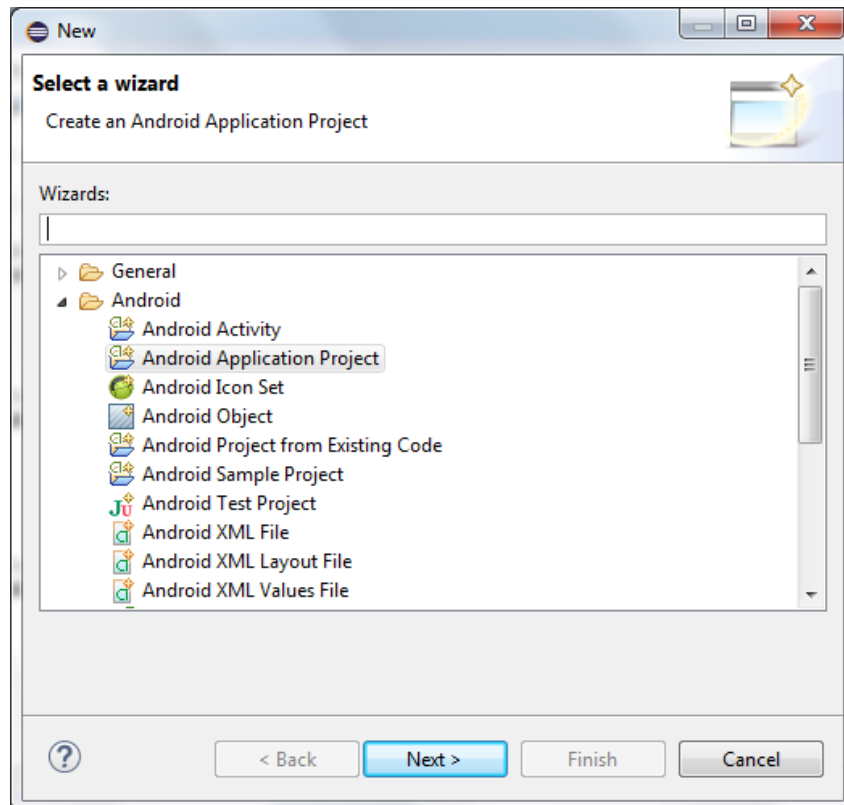


Gambar 17. Mereferensikan Android SDK

III. Membuat Aplikasi Sederhana

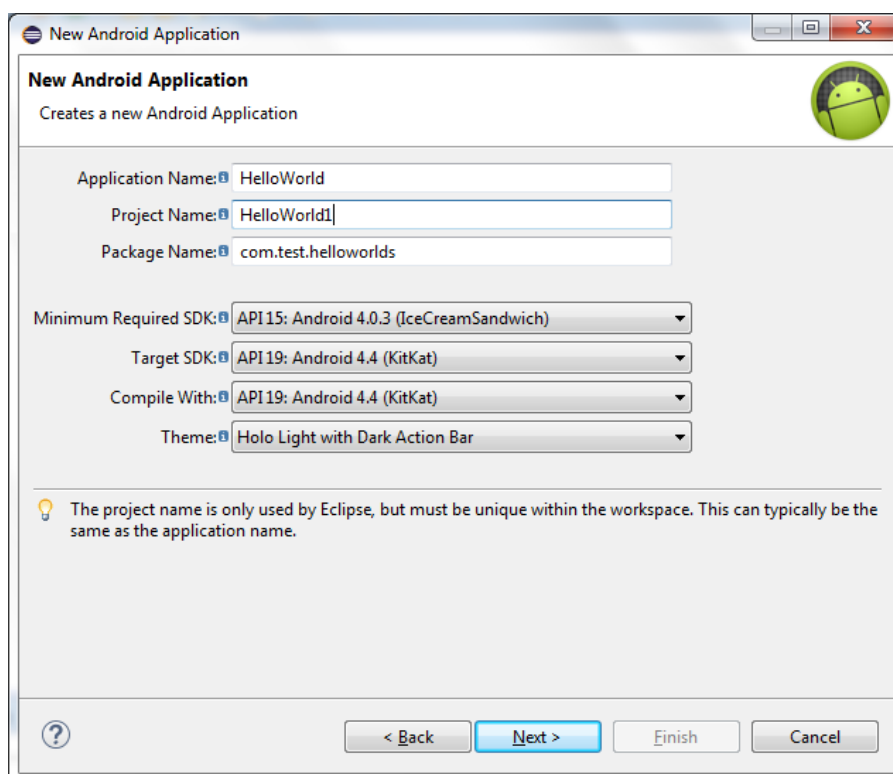
Sekarang kita akan membuat program pertama kita pada android dengan menggunakan perangkat yang sudah kita instal. Seperti belajar bahasa pemrograman yang lainnya, biasa program pertama yang dibuat adalah program untuk menampilkan tulisan “Hello World”, atau sering dikenal dengan sebutan program hello world.

Pertama jalankan IDE Eclipse, pilih File > New > Android Application Project (biasanya terdapat pada pilihan Others) sehingga muncul halaman seperti pada gambar 18 di bawah ini.



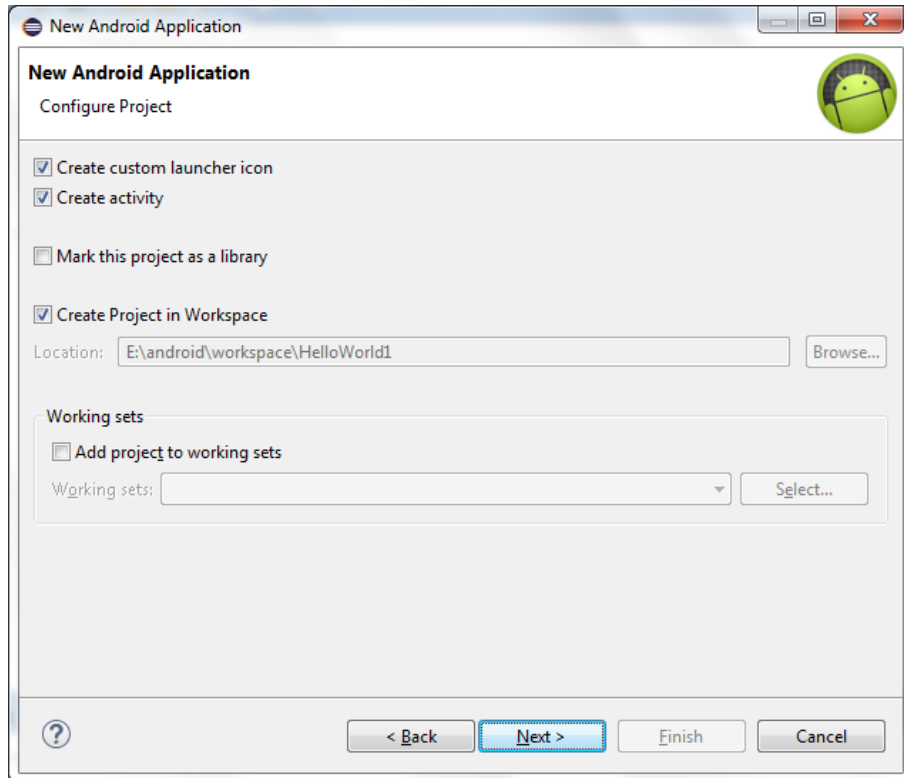
Gambar 18. *Create New Android Application Project*

Selanjutnya klik button Next dan akan muncul halaman seperti pada gambar 19 di bawah ini.



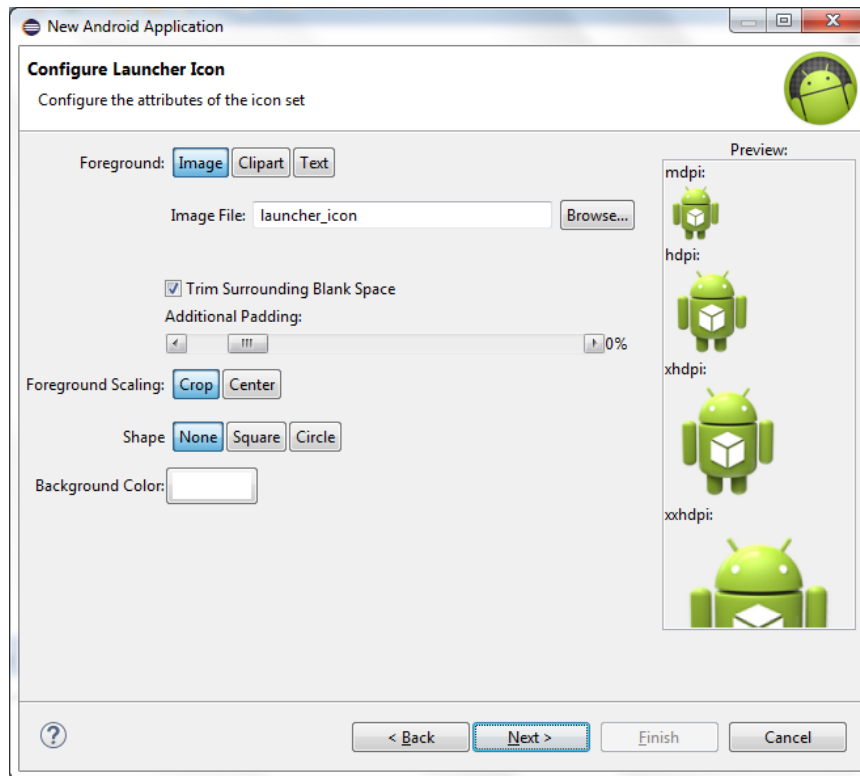
Gambar 19. Menentukan Nama Aplikasi dan Memilih SDK

Pada halaman gambar 19 kita akan mengisi nama aplikasi, nama project eclipse dan nama package. Selain itu kita juga akan memilih minimum SDK/API, target SDK/API dan API yang digunakan untuk proses *compile*. Selanjutnya klik button Next dan akan muncul halaman seperti pada gambar 20 di bawah ini.



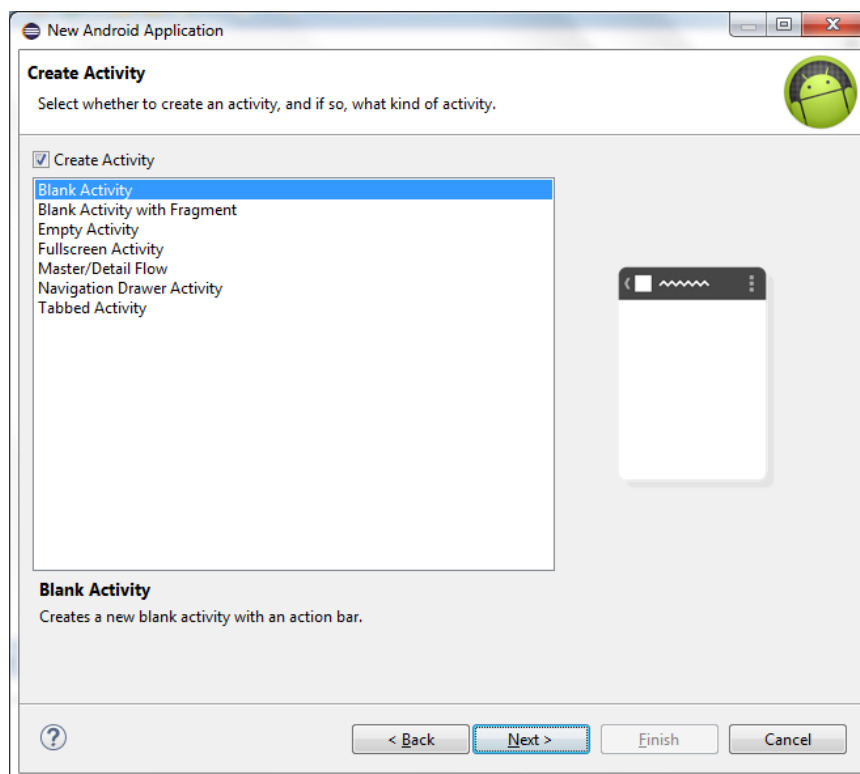
Gambar 20. Menentukan Folder Project

Pada halaman gambar 20 kita akan menentukan folder penyimpanan project. Selanjutnya klik button Next dan akan muncul halaman seperti pada gambar 21 di bawah ini.



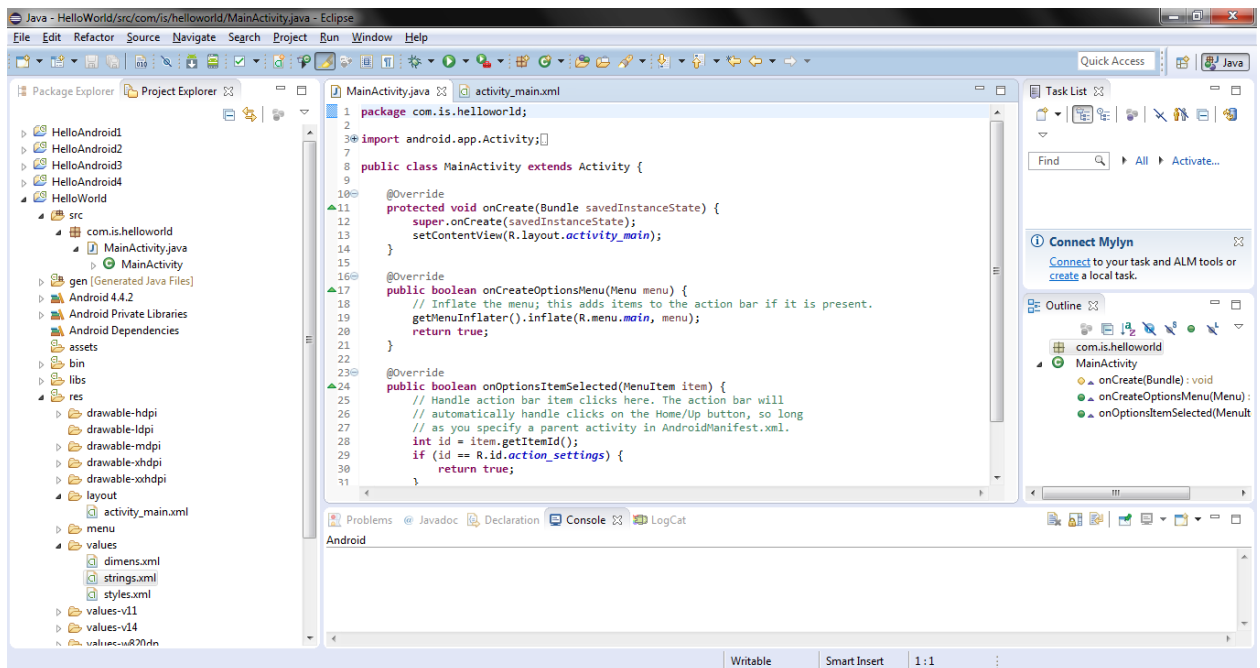
Gambar 21. Configure Launcher Icon

Pada halaman gambar 21 kita akan menentukan icon aplikasi dan warna background tampilan. Selanjutnya klik button Next dan akan muncul halaman seperti pada gambar 22 di bawah ini.



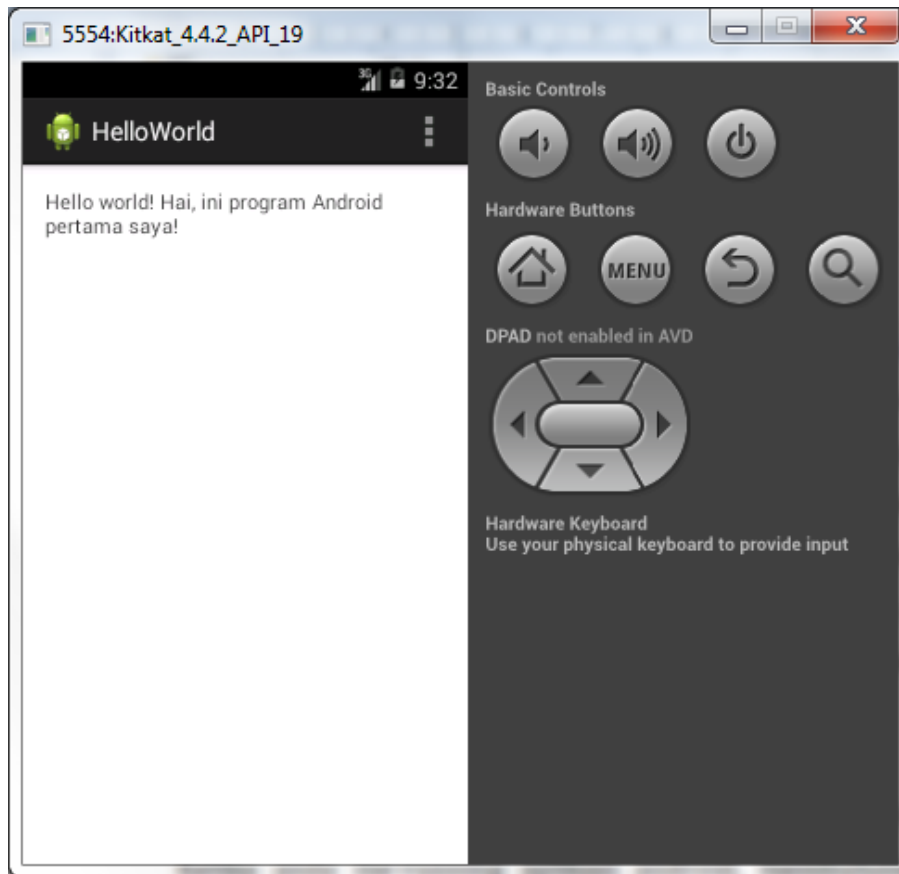
Gambar 22. Create Activity

Selanjutnya kita akan dihadapi pada halaman pengerjaan project eclipse seperti pada gambar 23 di bawah ini.



Gambar 23. Halaman Android Project pada Eclipse

Kita sudah bisa melakukan running pada program yang baru kita buat ini dengan cara klik kanan pada nama project kita yang ada di dalam Project Explorer lalu pilih Run As > Android Application. Proses run juga bisa dilakukan dengan klik tombol run yang berbentuk segitiga ke kanan (warna hijau putih) lalu pilih Run As > Android Application. Apabila aplikasi android HelloWorld yang baru kita buat, dilakukan proses run maka akan tampil seperti gambar 24 di bawah ini.



Gambar 24. Hasil Run Program Hello World

Ketika anda me-running aplikasi android, membutuhkan waktu untuk proses loading emulatoanya, dianjurkan setelah anda berhasil running program tidak perlu menutup emulatoanya apabila ada program lain yang akan di-running.

MainActivity.java

```
package com.is.helloworld;

import android.app.Activity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;

public class MainActivity extends Activity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

Activity merupakan basic class untuk sebuah interface. Dimana kita lihat bahwadi dalam Acitivity terdapat sebuah method onCreate yang berfungsi untuk memanggil setContentView. Sementara setContentView memanggil sebuah file main.xml

(R.layout.activity_main), activity_main.xml ini yang bertindak sebagai user interface. Sekarang kita coba lihat isi dari file activity_main.xml (folder res/layout) di bawah ini.

activity_main.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.is.helloworld.MainActivity" >
<!--
* android:text="@string/hello_world" bermaksud bahwa text diambil dari file
strings.xml
* yang ada pada folder res/values/
-->
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/hello_world" />

</RelativeLayout>
```

Dari source code activity_main.xml di atas, dapat kita lihat bahwa untuk menampilkan text adalah dengan coding android:text="@string/hello_world", dimana kode ini menunjukkan bahwa android:text berasal dari string.xml pada directory res/values, melalui pointer yang berada pada file R.java (folder gen/). Sekarang kita coba lihat isi dari file string.xml (folder res/values) di bawah ini.

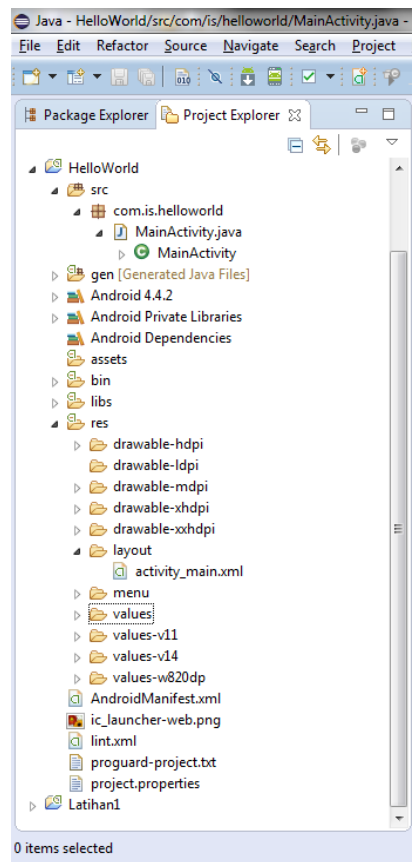
string.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>

    <string name="app_name">HelloWorld</string>
    <string name="hello_world">Hello world! Hai, ini program Android pertama
saya!</string>
    <string name="action_settings">Settings</string>

</resources>
```


Pada halaman pengerjaan pemrograman android di eclipse terdapat bagian Project Explorer seperti pada gambar 25 di bawah ini.



Gambar 25. Project Explorer Eclipse

Berikut penjelasan beberapa folder yang ada pada struktur project aplikasi android:

1. src/, src atau source memiliki fungsi sebagai tempat utama yang di dalamnya terdapat file-file Activity bertipe Java. File Activity ini terdiri dari beberapa Class yang menjadi logic dasar pemrograman aplikasi Android. Contohnya pada gambar diatas di folder src/ terdapat package yang di dalam package berisi file MainActivity.java.
2. gen/, folder ini secara otomatis telah dibuat oleh ADT, berisi berbagai ID yang digunakan untuk mengakses resources aplikasi pada folder res/. Sebaiknya kita tidak mengubah secara langsung isi file Java yang ada dalam folder ini.
3. Android 4.4.2/, folder ini menunjukkan versi android yang digunakan dan berisi berbagai class yang dapat digunakan.

4. assets/, folder ini secara default kosong. Anda dapat memasukkan berbagai file yang mendukung jalannya aplikasi, bisa berupa file audio, teks, video, atau file yang berhubungan dengan database.
5. bin/, folder ini berisikan hasil dari proses compile aplikasi yang kita buat. File *.apk ini otomatis terbuat ketika proses compile berhasil. File *.apk ini dapat kita pasang atau install di perangkat mobile android.
6. libs/, folder ini berisi file berbentuk *.jar sebagai tempat library untuk fitur tambahan aplikasi. Folder ini biasanya dipakai saat aplikasi kita misalnya akan menggunakan sebuah banner dari Admob. Maka sebelumnya kita harus memiliki file *.jar untuk Admob dan dimasukkan langsung ke folder atau melalui menu Project Properties -> Java Build Path -> Libraries tab -> Add JARs.
7. res/, folder ini berisi beragam file seperti file string, layout, audio, gambar. Folder ini terbagi menjadi beberapa subfolder yang terstruktur sesuai dengan kebutuhan yang diperlukan ketika proses development aplikasi. Beberapa subfolder tersebut diantaranya:
 - a. drawable-hdpi: folder ini digunakan untuk penyimpanan gambar yang terkompilasi untuk layar dengan resolusi tinggi.
 - b. drawable-ldpi: folder ini digunakan untuk penyimpanan gambar yang terkompilasi untuk layar dengan resolusi rendah.
 - c. drawable-mdpi: folder ini digunakan untuk penyimpanan gambar yang terkompilasi untuk layar dengan resolusi sedang.
 - d. drawable-xhdpi: folder ini digunakan untuk penyimpanan gambar yang terkompilasi untuk layar dengan resolusi lebih tinggi dari gambar yang ada di drawable-hdpi.
 - e. drawable-xxhdpi: folder ini digunakan untuk penyimpanan gambar yang terkompilasi untuk layar dengan resolusi lebih tinggi dari gambar yang ada di drawable-xhdpi.
 - f. layout: folder ini berisi berbagai file xml yang digunakan untuk mendefinisikan tampilan antarmuka dari aplikasi yang kita buat.
 - g. values: folder ini berisi nilai yang digunakan dalam aplikasi. Ada beberapa file standar yang dapat dijadikan acuan untuk disimpan pada folder values ini, yaitu:

- 1) style.xml: file ini untuk merepresentasikan style aplikasi yang dibuat. Jika pada file HTML, file ini hampi sama dengan file CSS.
- 2) string.xml: file ini untuk merepresentasikan nilai string yang dapat diakses melalui class R.strings.

Selain folder-folder di atas, tetdapat beberapa file yang penting untuk diperhatikan, antara lain Android Manifest dan Project Properties. File AndroidManifest.xml merupakan file yang berisi informasi nama paket java pada aplikasi, daftar komponen aplikasi seperti activies, informasi minimum platform, izin aplikasi, daftar libraries, dan kode versi serta nama versi. File project.properties akan menunjukkan target platform yang digunakan dalam aplikasi (versi android yang digunakan).

IV. Daftar Pustaka

4.1. Buku Utama

- [1] Michael, I.S. 2011. Membongkar Source Code Berbagai Aplikasi Android. Yogyakarta: Gava Media.
- [2] Priyo E.U. 2012. From Newbie to Advanced – Mudahna Membuat Aplikasi Android. Yogyakarta: Andi.
- [3] Safaat N.H. 2014. Android: Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android. Bandung: Informatika.

4.2. Referensi

<https://developer.android.com/index.html> [September 2015]
<http://androidforums.com> [September 2015]

V. Materi Berikutnya

Pokok Bahasan	Android Widget Bagian 1
Sub Pokok Bahasan	<ol style="list-style-type: none"> 1. Komponen user interface 2. Konsep widget 3. Text view 4. Edit text 5. Button 6. Checkbox 7. Radio button and radio group 8. Image view and image button 9. Bekerja dengan kumpulan widget