

Tutorial

Pengantar Pengembangan Aplikasi Android

oleh: [Mohamad Sani \(sani@msani.net\)](mailto:sani@msani.net)
17 Oktober 2010

Android adalah sistem operasi untuk *mobile device* yang awalnya dikembangkan oleh Android Inc., Perusahaan ini kemudian dibeli oleh Google pada tahun 2005. Android dibuat berdasarkan kernel Linux yang dimodifikasi.

Aplikasi Android ditulis dengan bahasa Java, menggunakan Java Core Libraries. Aplikasi Android dijalankan di atas VM bernama Dalvik Virtual Machine.

Pada tutorial ini, akan dijelaskan langkah-langkah membuat aplikasi Android mulai dari instalasi perangkat-perangkat lunak yang diperlukan sampai dapat menjalankan program Hello World pada Android *device*.

1 Mempersiapkan Komputer

Sebelum mengunduh Android SDK, pastikan di komputer anda sudah terinstall JDK versi 5 atau 6 dan Eclipse versi 3.4 atau 3.5.

Eclipse hanya dibutuhkan apabila kita ingin mengembangkan aplikasi dengan bantuan sebuah plugin Eclipse yang bernama Android Development Tools (ADT). Sebenarnya kita bebas memilih penyunting teks (*text editor*) apapun, tetapi menggunakan Eclipse adalah teknik yang paling cepat dan direkomendasikan. Eclipse dapat diunduh dari <http://www.eclipse.org/downloads>. Versi yang direkomendasikan adalah “Eclipse for Java Developers” atau “Eclipse for RCP Developers”. Untuk versi 3.5, versi yang direkomendasikan adalah “Eclipse Classic”. Pada tutorial ini, kita akan menggunakan Eclipse.

Tidak direkomendasikan menggunakan Eclipse 3.6 Helios karena ada sedikit masalah dengan ADT plugin. Gunakan Eclipse 3.4 Ganymede atau 3.5 Galileo. Pastikan juga Eclipse memiliki Java Development Tools (JDT), biasanya JDT sudah terdapat pada kebanyakan paket Eclipse.

JDK bersifat wajib. JRE saja tidak mencukupi. JDK dapat diunduh dari <http://java.sun.com/javase/downloads/index.jsp>.

2 Menginstall Android SDK

Unduh Android SDK starter package dari http://dl.google.com/android/android-sdk_r07-windows.zip Kemudian unzip.

Android SDK starter package belum merupakan lingkungan pengembangan (*development environment*) yang lengkap. Ia hanya berisi alat-alat (*tools*) utama saja. Dengan Android SDK starter package inilah nantinya kita akan mengunduh alat yang lainnya.

Setelah membongkar (*unzip*), simpan direktori di tempat yang nyaman, misalkan C:\Program Files\android-sdk-windows. Kita perlu untuk merujuk kembali alamat direktori ini nanti.

3 Menginstall ADT untuk Eclipse

1. Buka Eclipse, klik **Help > Install New Software**.
2. Pada dialog Available Software yang terbuka klik **Add...**
3. Pada dialog Add Site yang muncul, masukan nama remote site (contoh: Android Plugin) pada Name field. Pada Location field, masukkan URL berikut: <https://dl-ssl.google.com/android/eclipse/>. Klik **OK** dan tunggu sampai “pending” selesai.
4. Kembali ke Available Software view, seharusnya kita bisa melihat “Developer Tools” tertambah pada daftar. Centang checkbox pada **Developer Tools** yang secara otomatis mencentang pula Android DDMS dan Android Development Tools. Klik **Next** dan tunggu sebentar.
5. Pada dialog Install Detail yang muncul, Android DDMS and Android Development Tools features seharusnya terdaftar. Klik **I accept ...** untuk menyetujui perjanjian lisensi kemudian klik **Finish**. Tunggu sampai proses *fetching* selesai.
6. Mulai ulang (*restart*) Eclipse.

Jika anda mengalami masalah saat mengunduh plugin pada langkah nomor 3, coba ganti https dengan http. Jika muncul warning pada langkah 5, klik bahwa kita mempercayai paket yang diunduh.

4 Menambahkan Platform Android

1. Buka Android SDK and AVD Manager (Android SDK.exe pada direktori SDK starter package).
2. Pada bagian Available Package, centang **SDK Platform Android 2.2 API 8**.
3. Klik **Install Selected** untuk mengunduh dan menginstall.

Setidaknya anda harus menginstall 1 platform agar dapat mengembangkan aplikasi Android. Tergantung untuk Android veri berapa anda ingin mengembangkan aplikasi. Anda dapat mencentang platform lain dan komponen lain (Documentation, Samples, USB Driver, Google APIs) untuk diunduh. Meskipun hal ini direkomendasikan, namun tidak wajib.

Jika tersedia API yang lebih baru dari level 8, set nilainya ke yang paling baru. Biasanya API yang levelnya lebih tinggi mendukung API yang di bawahnya namun belum tentu sebaliknya. Pada satu dua kasus saja (dengan kata lain jarang sekali), cocok ke bawah (*backward compatible*) tidak dipertahankan.

Jika kotak “Choose Package to Install” muncul secara otomatis, pastikan anda mencentang hanya yang anda perlukan. Anggapannya (*default*-nya), semua platform akan tercentang. Mengunduh semua platform akan memakan waktu yang lama.

5 Membuat AVD

1. Pada Eclipse, pilih **Window > Preferences**.
2. Masukkan jalur (**path**) dimana anda mengekstrak Android SDK pada langkah no 2 di atas. Klik OK
3. Pada Eclipse, pilih **Window > Android SDK and AVD Manager**.
4. Pilih **Virtual Devices** di panel kiri.
5. Klik **New**. Dialog Create New AVD muncul.
6. Ketik nama AVD, misal "avd_saya".
7. Pilih sebuah target platform (versi Android SDK, misal 2.2) yang ingin anda jalankan pada emulator. Anda dapat mengacuhkan field yang lain untuk saat ini.
8. Klik **Create AVD**.

Pada tutorial ini, kita akan menjalankan aplikasi pada Emulator Android. Sebelum kita dapat menjalankan Emulator, kita harus membuat Android Virtual Device (AVD). AVD mendefinisikan *system image* dan setting *device* yang digunakan Emulator.

6 Membuat Proyek Android Baru

1. Dari Eclipse, pilih **File > New > Project**.
2. Pilih **Android Project** dan klik **Next**.
3. Isikan field-field dengan nilai berikut:
 - Project name : HelloWorld
 - Application name : Hello, World
 - Package name : com.example.helloworld (atau namespace pribadi anda)
 - CreateActivity : HelloWorld
4. Centang versi platform pada **Build Target**, misal: Android 2.2
5. Klik **Finish**.

Pilihan Android Project pada langkah nomor 2 hanya ada bila plugin ADT untuk Eclipse terpasang dengan sukses.

Berikut penjelasan dari masing-masing isian:

- **Project Name**

Ini adalah nama proyek Eclipse: nama direktori yang akan berisi berkas-berkas proyek.

- **Application Name**

Ini adalah judul aplikasi anda: nama yang akan muncul pada perangkat Android.

- **Package Name**

Ini adalah **namespace** paket (mengikuti aturan yang sama untuk paket dalam bahasa pemrograman Java) dimana semua kode sumber (*source code*) anda diletakkan. Nama ini juga menjadi nama paket di mana stub Activity akan dihasilkan.

Nama paket Anda harus unik terhadap semua paket yang terinstal pada sistem Android, oleh karena itu, penting untuk menggunakan gaya penulisan yang sesuai standar untuk aplikasi Anda. Contoh di atas menggunakan "com.example" *namespace*, dimana *namespace* dicadangkan untuk dokumentasi contoh. Ketika Anda mengembangkan aplikasi sendiri, Anda biasanya menggunakan *namespace* sesuai dengan

nama perusahaan/organisasi Anda.

- **Create Activity**

Ini adalah nama untuk *class stub* yang akan dihasilkan oleh plugin. Ini akan menjadi sebuah sub kelas dari kelas Android Activity. Sebuah Activity adalah sebuah kelas yang dapat menjalankan dan melakukan pekerjaan. Activity dapat membuat UI jika mau, tetapi tidak diharuskan. Activity hampir selalu digunakan sebagai dasar untuk aplikasi sebagaimana direkomendasikan oleh checkbox, tetapi bersifat optional.

- **Min SDK Version**

Nilai ini menentukan tingkat API minimum yang dibutuhkan oleh aplikasi Anda. Biasanya API yang levelnya lebih tinggi mendukung API yang di bawahnya namun belum tentu sebaliknya. Pada satu dua kasus saja (dengan kata lain jarang sekali), cocok ke bawah (*backward compatible*) tidak dipertahankan. Oleh karena itu, anda dianjurkan untuk mempelajari tentang API Level pada Android.

7 Menulis Program HelloWorld

1. Buka berkas HelloWorld.java yang terletak pada HelloWorld > src > com.example.helloworld
2. Ganti isi HelloWorld.java yang ada dengan kode berikut:

```
package com.example.helloworld;

import android.app.Activity;
import android.os.Bundle;
import android.widget.TextView;

public class HelloWorld extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        TextView tv = new TextView(this);
        tv.setText("Hello, World!");
        setContentView(tv);
    }
}
```

Perhatikan bahwa kelas `HelloWorld` adalah turunan dari kelas `Activity`. Sebuah `Activity` adalah sebuah entiti aplikasi yang digunakan untuk melakukan serangkaian aksi. Sebuah aplikasi dapat terdiri dari banyak activity, namun pengguna berinteraksi dengan satu activity pada satu waktu. Method `onCreate()` akan dipanggil oleh sistem Android ketika activity dijalankan pertama kali. Di sini lah tempat anda melakukan semua inisialisasi dan menyiapkan UI. Sebuah activity biasanya memiliki *user interface*, tetapi tidak harus.

User interface Android terdiri dari hierarki objek yang disebut `Views`. Sebuah `View` adalah sebuah *drawable object* yang digunakan sebagai sebuah elemen pada *layout* UI seperti tombol, gambar, atau [dalam contoh ini] label teks. Setiap objek-objek tersebut adalah sub kelas dari kelas `View` dan sub kelas yang menangani teks adalah `TextView`.

Pada contoh di atas, kita membuat sebuah `TextView` dengan kelas *constructor* yang menerima

instance Context sebagai parameternya. **Context** adalah sebuah *handle* ke sistem. **Context** menyediakan layanan-layanan seperti mengurus *resource*, mendapatkan akses ke basis data dan *preferences*, dll. Kelas **Activity** adalah turunan dari **Context**, karena aplikasi di atas adalah sub kelas dari **Activity**, ia juga menjadi sub kelas dari **Context**. Oleh karena itu, kita dapat melemparkan **this** sebagai referensi **Context** ke **TextView**.

Terakhir, kita melemparkan **TextView** ke **setContentview()** untuk menampilkannya sebagai konten untuk **Activity** UI. Jika aplikasi tidak memanggil *method* ini, tidak akan ada UI yang ditampilkan dan aplikasi hanya akan menampilkan layar kosong.

8 Menjalankan Program

1. Klik **Run > Run**
2. Pilih “Android Application”

Plugin Eclipse akan secara otomatis membuat konfigurasi untuk proyek anda dan menjalankan Emulator Android. Emulator biasanya butuh waktu beberapa menit untuk *booting*, jadi yang sabar ya... :)