

BAB II

INSTALASI ANDROID DAN HELLO WORLD

1. Latar Belakang

Setiap proses pembelajaran selalu membutuhkan media pembelajaran yang dapat membantu pelaksanaan proses pembelajaran. Modul ini merupakan salah satu solusi yang dapat membantu pelaksanaan proses pembelajaran sehingga modul ini diharapkan dapat membantu proses pembelajaran yang dilakukan peserta.

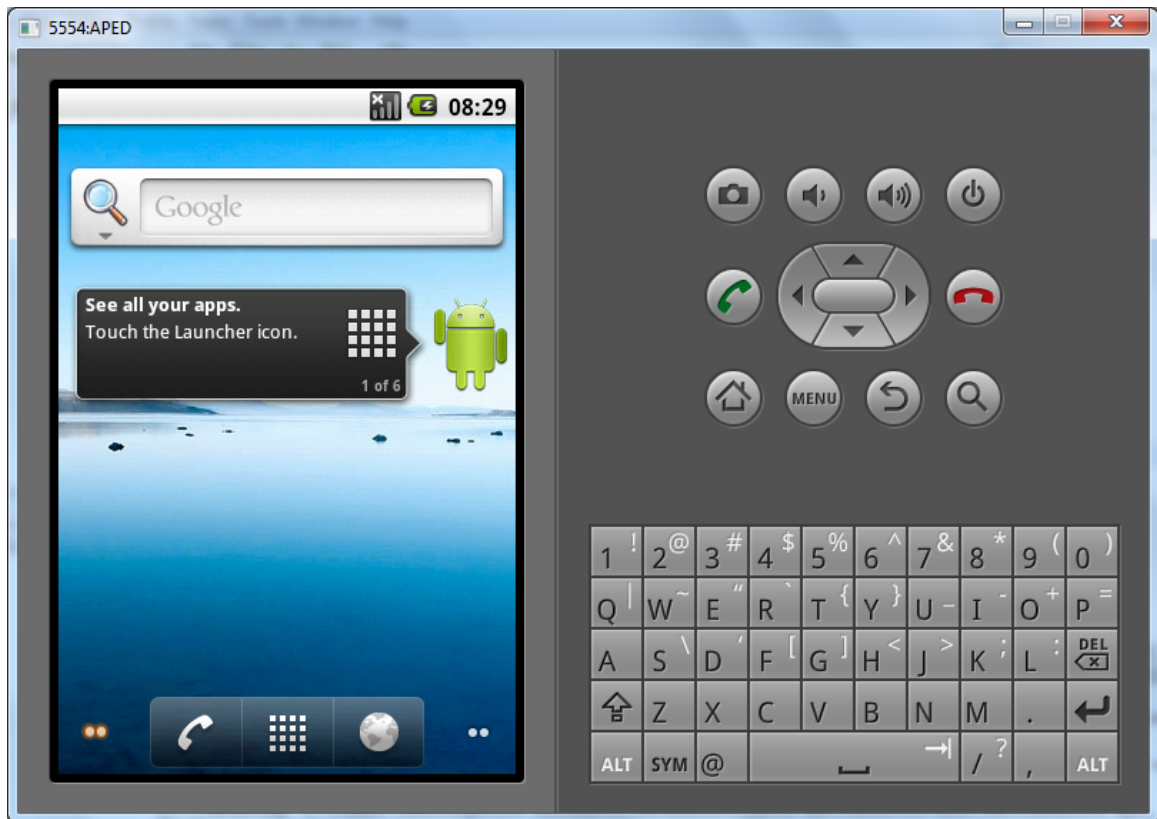
2. Tujuan

Tujuan pembuatan modul ini adalah :

- Peserta memahami dan mengerti langkah awal pemrograman dengan Android.
- Peserta mengetahui persiapan yang dilakukan sebelum melakukan pemrograman Android.
- Peserta dapat melakukan praktikum berdasarkan materi yang ada pada modul ini.

3. Pemrograman Android

Android adalah sebuah system yang merupakan system operasi berbasis java dan bekerja pada Linux kernel 2.6, dan sistemnya lebih ringan serta dengan fitur lebih lengkap.



Gambar 3.1 tampilan layar utama android pada emulator android

Aplikasi pada android dikembangkan dengan menggunakan bahasa pemrograman java dan lebih mudah diterapkan pada *platform* baru. Fitur yang lain terdapat pada Android adalah akselerasi pada *engine* grafis 3D (berdasarkan dukungan dan kecocokan pada *hardware*), dukungan fitur untuk *database* dengan SQL, serta terintegrasi dengan web browser.

Jika anda sudah terbiasa dengan bahasa pemrograman java, anda akan menyukai untuk melakukan pengembangan program pada *user interface* (UI), di mana tempat untuk UI ditangani tanpa menggunakan kode pemrogram. Android, sementara ini melakukan pengembangan program pada UI, dengan dukungan baru berbasis XML pada tampilan UI. Salah satu yang menarik dari fitur android yaitu pada *third-party* dari aplikasi dengan istilah “*Home Grown*”, yang dieksekusi pada prioritas sistem yang sama yaitu pada sistem *core*-nya, merupakan bagian utama dari beberapa sistem. Yang mana telah memberikan sebuah sistem aplikasi yang *embedded*, untuk mendapatkan sebuah eksekusi yang terbaik dan juga prioritas *thread* yang terdapat pada aplikasi yang

dibuat di bagian ketiga *developers*. Dan juga, aplikasi yang lain dijalankan tanpa menggunakan *thread*-nya sendiri sehingga sangat ringan untuk *virtual machine*.

Sisi lain dari generasi SDK-nya adalah dengan *libraries* yang telah disempurnakan dan disediakan untuk kita dalam mengembangkannya. Fitur yang menarik lainnya dari *Android Developers* adalah kita dapat mengakses apapun dari sistem operasi yang telah kita akses sebelumnya. Jika kita mau membuat sebuah aplikasi *phone dials*, kita mendapatkan akses *phone's dialer*, jika kita mau membuat sebuah aplikasi *utility GPS internal* telepon, kita dapat mengaksesnya.

4. Android SDK

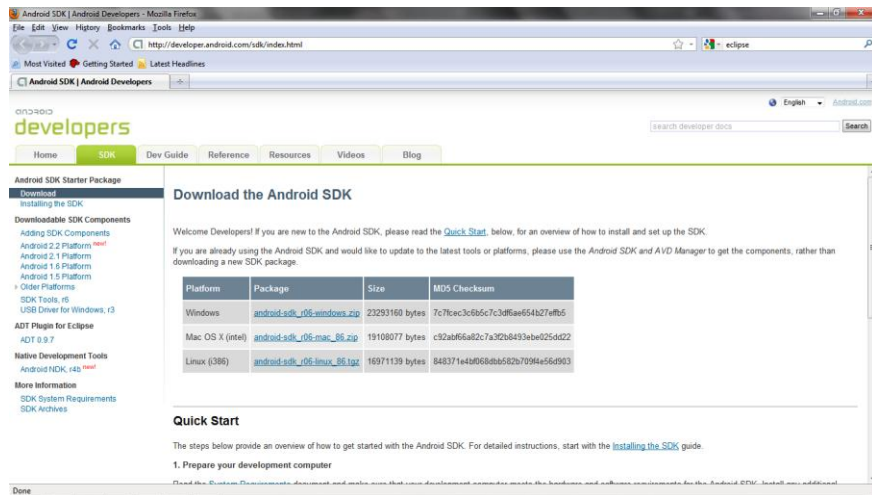
Pada pembahasan ini kita membahas tentang SDK Android, SDK Android merupakan sebuah *package* yang berupa *plugin* untuk IDE dan didalamnya terdapat banyak *file* dan *tool* untuk membantu anda dalam pembuatan sebuah aplikasi yang dijalankan pada Android.

Pada SDK Android juga terdapat sebuah *library* yang digunakan untuk mengikat aplikasi dengan *core Android* yang berhubungan dengan fungsi pada ponsel (melakukan panggilan telepon dan menerima panggilan telepon), fungsi pada GPS, dan SMS.

4.1 Dokumentasi Tentang SDK

Dokumentasi SDK terdapat pada folder Docs didalam SDK yaitu berada pada `../%sdk folder%/DOCS`. Dokumentasi SDK ini disediakan tentang tahapan download dan install SDK “Tahap Awal (*Getting Started*)”, dan cara cepat untuk mengembangkan aplikasi serta penjelasan tentang *Packge*.

Dokumentasi ini berupa *file* (.html) atau dalam bentuk *format* HTML dan dapat di akses melalui `documentation.html` pada *root folder* SDK, berikut tampilan halaman `documentation.html` :



Gambar 4.1 tampilan halaman *documentation.html*

Anda dapat melakukan navigasi pada halaman yang terdapat pada *documentation.html* dengan link-link yang tersedia pada halaman tersebut.

4.2 Android Sample

Folder Android Samples yang berada pada `../%sdk folder%/SAMPLES`, didalamnya terdapat beberapa aplikasi demo yang merupakan contoh yang mewakili keseluruhan fungsi Android, diantaranya :

- *API Demo.*
- *Hello, Activity!.*
- *Lunar Lander.*
- *Note Pad.*
- *Skeleton App.*
- *Snake.*
- *dll*

Contoh aplikasi ini yang tersedia dari Developer Android sebagai acuan dalam mengembangkan aplikasi Android.

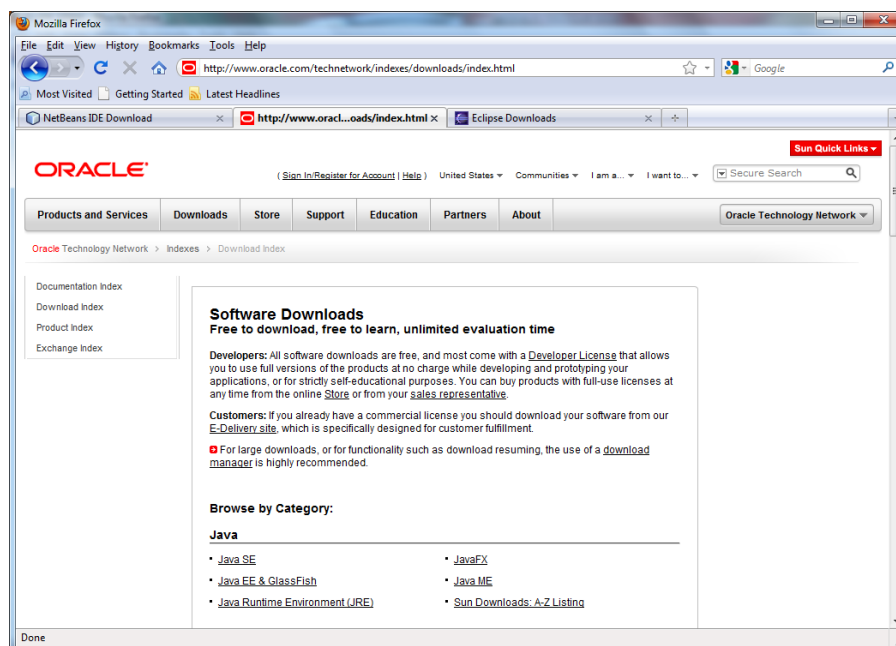
5. DOWNLOAD DAN INSTALASI JDK/JRE & IDE

Pada pembahasan ini dibahas pemasangan dan cara mendapatkan *file installer* Eclipse pada tahapan pemrograman Android.

5.1 Tahapan *Download* dan instalasi JRE

Sebelum anda melakukan *download* dan instalasi Eclipse, anda harus memiliki JRE (Java Runtime Environment) dan sudah terinstalasi pada komputer anda, karena Eclipse merupakan aplikasi yang bekerja dengan menggunakan java. Apabila JRE belum terpasang pada komputer anda, maka ketika anda menjalankan aplikasi Eclipse, anda akan melihat kotak dialog yang menyatakan *error* pada saat aplikasi Eclipse terbuka

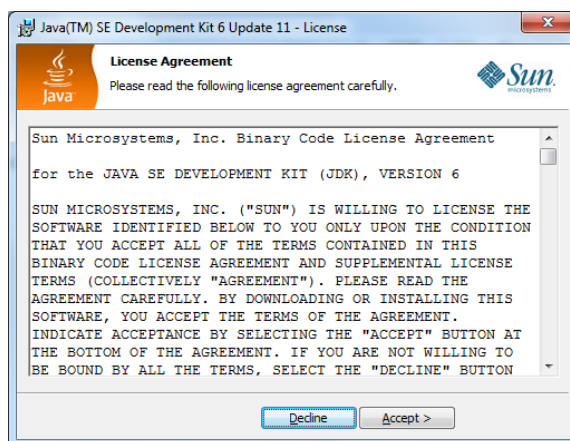
Untuk memulai *download* buka aplikasi browser anda kemudian ketikkan alamat <http://www.oracle.com/technetwork/indexes/downloads/index.html>, seperti pada tampilan gambar di bawah ini :



Gambar 5.1.1 tampilan *page web* dari Oracle

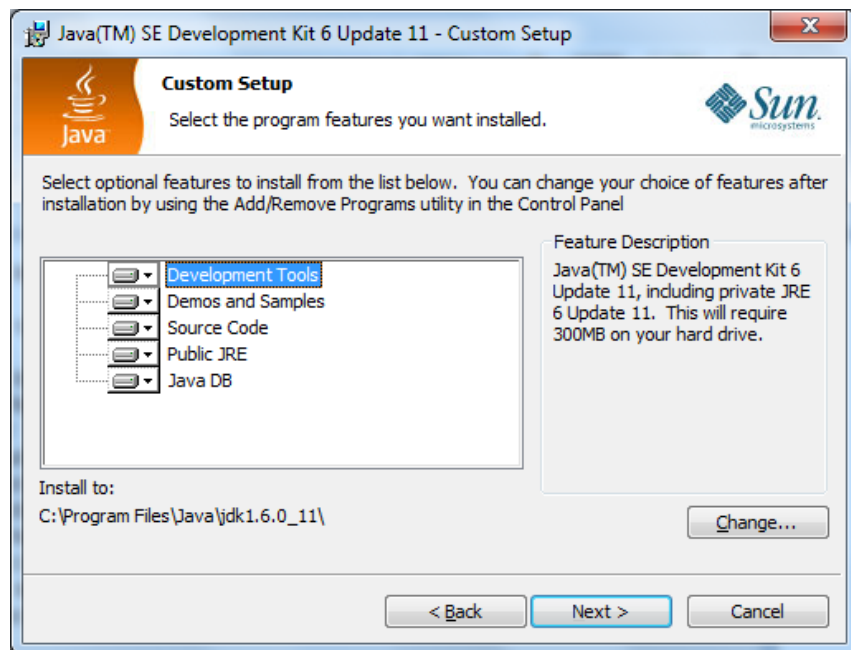
Sebenarnya kita membutuhkan JRE untuk menjalankan Eclipse, pada halaman web ini anda ditunjukkan untuk mendownload JDK yang sudah satu *bundle* (paket) dengan JRE. Alasan untuk mendownload full JDK tidak lain karena terdapat beberapa contoh cara membuat aplikasi Android diluar Eclipse hanya dengan menggunakan JDK *tools*, jika anda ingin mengikuti panduan (*tutorial*) tersebut kita harus menggunakan *full* JDK. Pada halaman SDN *download*, pilih JDK yang disediakan pada *page* download di situs Oracle kemudian *download* sesuai dengan kebutuhan sistem anda.

Pada saat *download* kotak dialog akan tampil, Klik *save* jika anda hendak menyimpan *file installer* JDK, atau anda bisa melakukan instalasi secara langsung dengan mengklik tombol *run*. Pada saat instalasi berlangsung pertama anda akan diberikan sebuah pernyataan persetujuan yang berupa *license agreement*, seperti pada tampilan gambar di bawah ini :



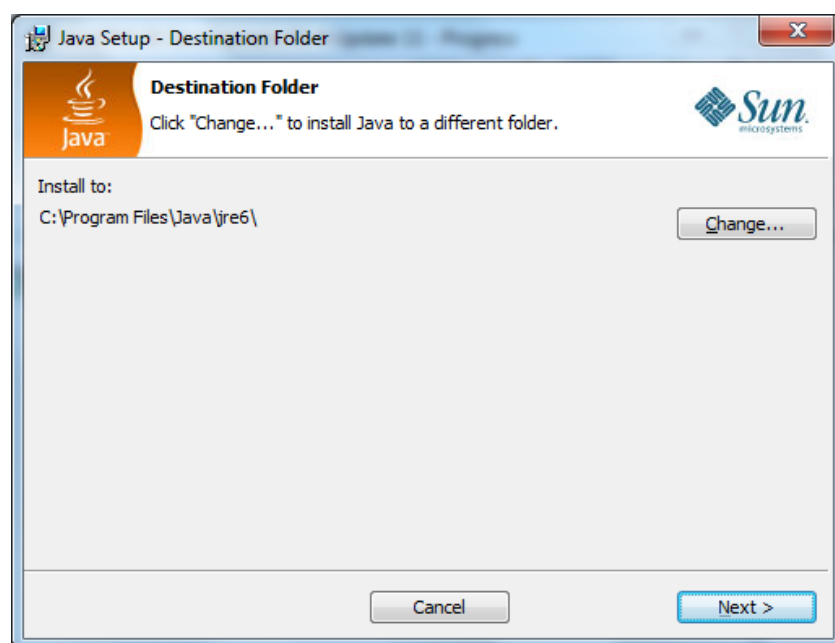
Gambar 5.1.2 *license agreement*

Klik pada “*I accept the terms in the license agreement*” untuk melanjutkan ke proses instalasi, selanjutnya klik tombol *next*. Setelah itu anda diberikan pilihan fitur pada instalasi JDK seperti pada gambar di bawah ini :



Gambar 5.1.3 pilihan fitur pada JDK

Pada tampilan di atas, anda dapat mengatur fitur pada JDK yang akan diinstal, kemudian klik next untuk melanjutkan. Setelah proses instalasi jdk selesai, akan tampil dialog Destination Setup JRE untuk instalasi JRE. Kemudian klik next jika telah selesai mengatur lokasi instalasi JRE.



Gambar 5.1.4 tampilan Java Setup - Destination Folder

Tunggu proses instalasi selesai, kemudian akan tampil dialog Instalasi Complete seperti pada gambar di bawah ini :



Gambar 5.1.5 tampilan instalasi selesai

5.2 Menggunakan Eclipse

Pada pembahasan ini, mungkin terdapat pertanyaan kenapa kita menggunakan eclipse untuk pemrograman Android? Alasan kenapa kita menggunakan eclipse untuk pemrograman Android karena :

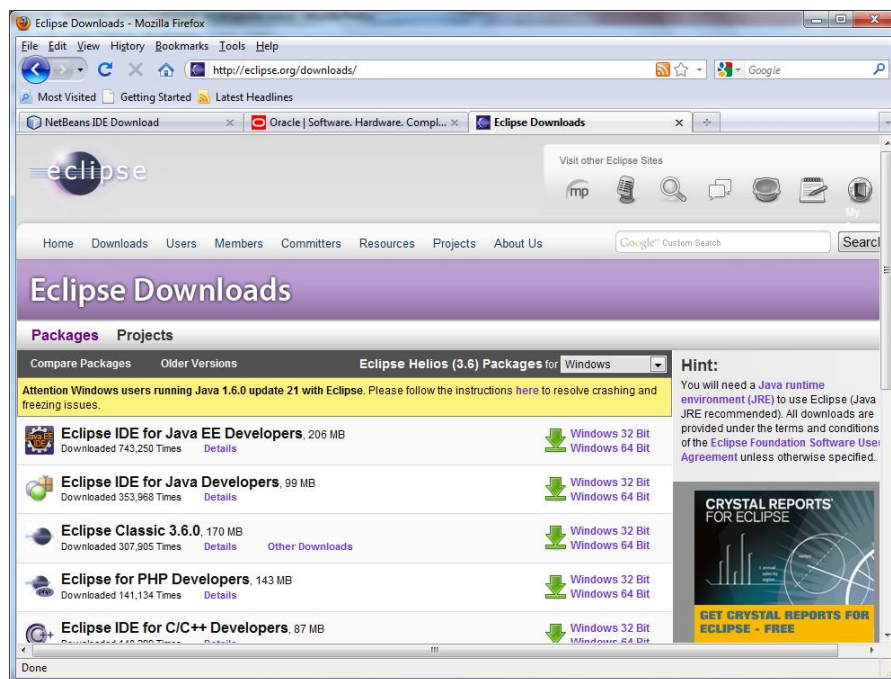
- Free (bebas digunakan).
- Terdapat IDE java didalamnya.
- Sangat mudah digunakan.
- Terbuka untuk para *programmer* pengembang program java.

- *Open Handset Alliance* menyediakan *plugin* Andriod untuk Eclipse ketika anda membuat *project* yang spesifik, melakukan *compile*, dan menggunakan *Android emulator* untuk menjalankannya

• Tahapan Download dan Instalasi Eclipse

Jika anda belum memiliki *installer* Eclipse, anda dapat mendownload terlebih dahulu *installer* Eclipse tersebut di alamat web “<http://www.eclipse.org/download>” caranya :

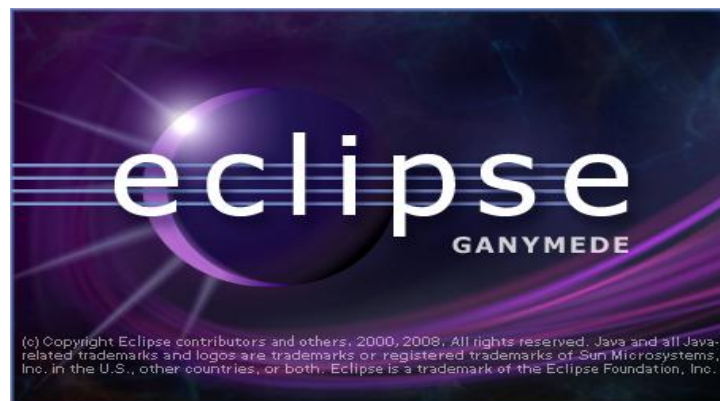
- Buka aplikasi *browser* pada komputer anda,
- Ketikan alamat di atas <http://www.eclipse.org/downloads>



Gambar 5.2.1 halaman web *download* Eclipse

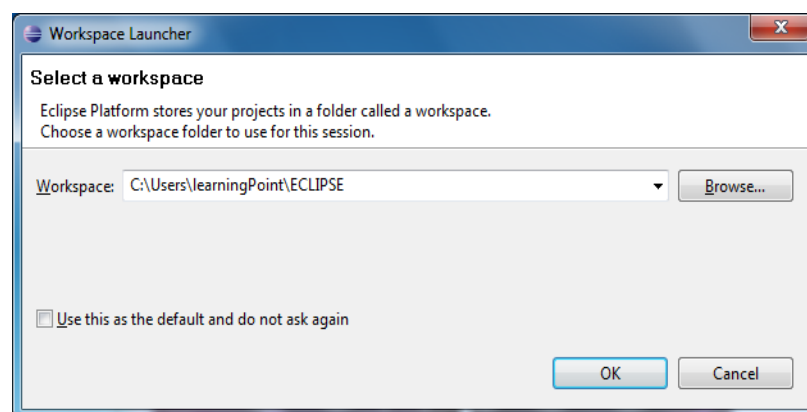
Pilih kategori Eclipse dan *download file installer* Eclipse sesuai dengan kebutuhan sistem operasi pada komputer anda, serta simpan *package* Eclipse pada *directory* yang anda inginkan untuk tempat menyimpan *file installer* , dan tunggu proses *download* selesai.

- Setelah anda selesai proses *download* Eclipse, buka *directory* tempat *file* Eclipse yang telah di *download* tadi kemudian *extract package installer* Eclipse yang berekstensi (.zip).
- Buka folder hasil dari *extract* tadi dan jalankan Eclipse.exe untuk menjalankan Eclipse.



Gambar 5.2.2 *splash screen Eclipse* pada saat *startup*

- Setelah tampil kotak dialog Workspace Launcher, anda dapat mengatur tempat/lokasi *workspace project* Eclipse. Seperti pada gambar dibawah ini:



Gambar 5.2.3 Dialog *workspace Launcher*

5.3 Menggunakan Netbeans

Selain menggunakan Eclipse, kita juga dapat menggunakan Netbeans sebagai *tools* dalam melakukan pemrograman di Android.

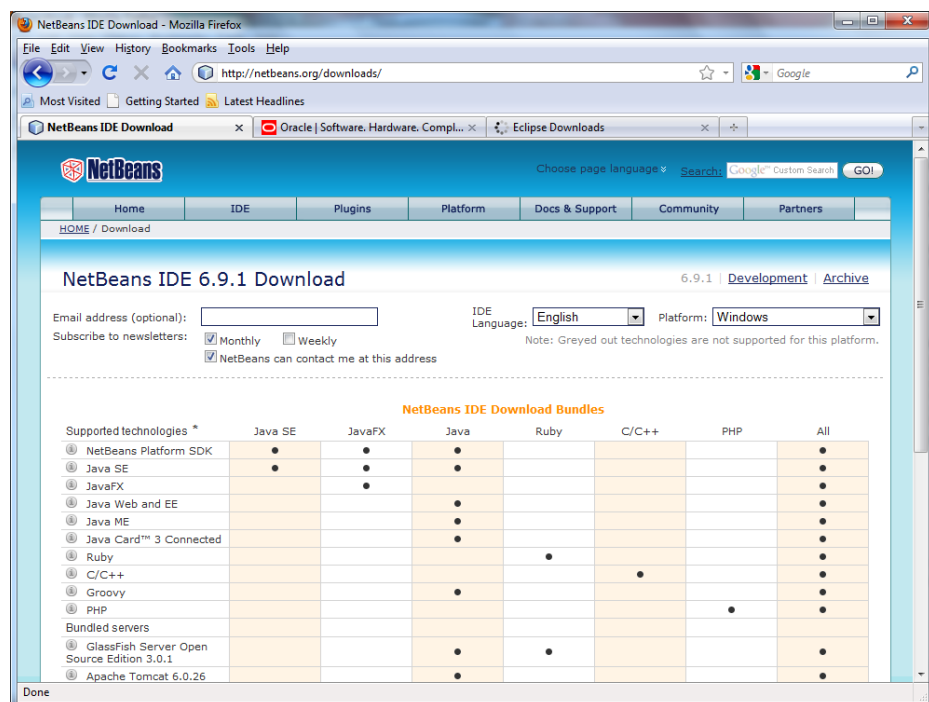
- **Download dan Instalasi Netbeans**

Apabila anda belum memiliki *installer* Netbeans, anda dapat men-*download installer* Netbeans pada <http://netbeans.org/downloads/>.

Pada saat modul ini ditulis, Netbeans IDE telah mencapai versi 6.9. tetapi untuk kedepannya kita akan menggunakan Netbeans versi 6.5

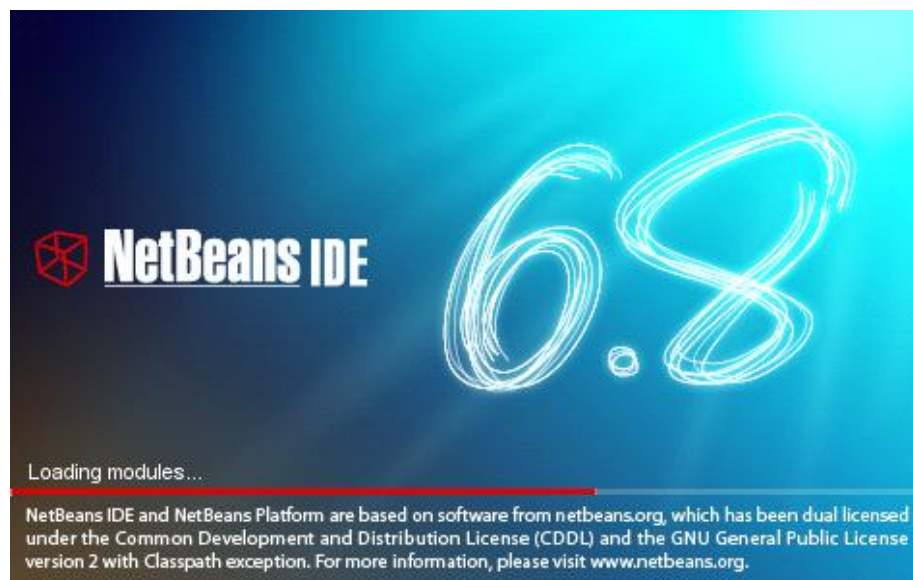
Berikut tata cara melakukan *download* Netbeans dari *browser* anda:

- Buka aplikasi *browser* anda.
- Masukkan URL <http://netbeans.org/downloads/>



Gambar 5.3.1 tampilan halaman <http://netbeans.org/downloads/>

- Pilih tipe Netbeans yang anda kehendaki dan lakukan *download* dengan menekan *button* “*download*” pada web browser. Simpan *installer* Netbeans tersebut pada tempat yang anda kehendaki.
- Setelah proses *download* selesai, anda bisa langsung menuju direktori tempat dimana *installer* Netbeans tersebut disimpan. Lalu *double click installer* tersebut untuk memulai instalasi.
- Ikuti petunjuk-petunjuk instalasi yang ada. Apabila anda telah selesai meng-*install* Netbeans, maka akan muncul shortcut untuk mengakses Netbeans pada *Start Menu* anda.
- Jalankan Program Netbeans anda untuk memulai menggunakan Netbeans.



Gambar 5.3.2 tampilan *Splashscreen* Netbeans

6. **DOWNLOAD DAN INSTALL ANDROID SDK**

Pada pembahasan di atas kita membahas tentang *download* dan *install primary development environment* anda. Dan *development environment* sudah terbentuk dengan menggunakan Eclipse dan Netbeans sebagai IDE java. Anda dapat menggunakannya untuk mengembangkan aplikasi java. Tetapi anda harus melakukan satu tahapan lagi sebelum anda memulai pemograman dalam membuat aplikasi handphone, anda harus melakukan konfigurasi sebagai jalan dalam memfasilitasi mengembangkan Android.

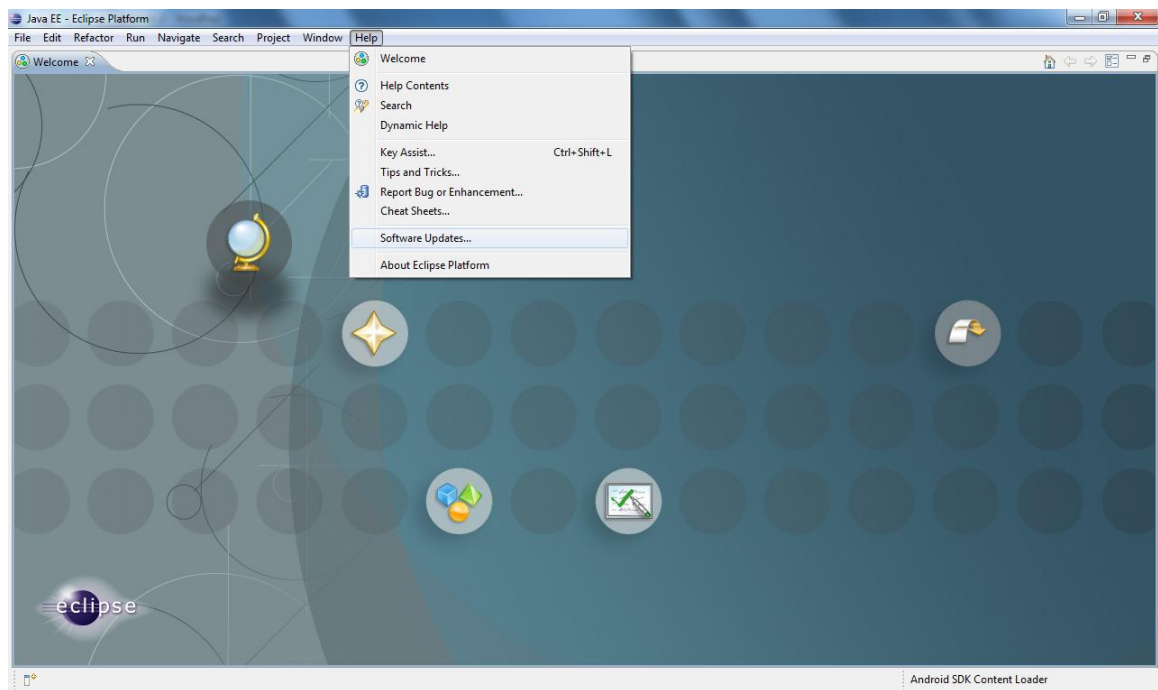
Karena Eclipse dan Netbeans merupakan media yang mengembangkan java, maka anda dapat membuat dan melakukan edit dengan mudah pada *project* java. Tetapi anda tidak bisa melakukan pengembangan Android jika *library* Eclipse dan Netbeans belum memiliki *library* Android untuk mengembangkan Android. Untuk membuat *project* Android anda harus memiliki SDK dan melakukan instalasi pada Eclipse dan Netbeans anda. Oleh karena itu anda harus melakukan *download* terlebih dahulu plugin Android untuk Eclipse dan Netbeans. Dengan menerapkan plugin ini anda dapat memulai untuk membuat aplikasi Android dengan menggunakan Eclipse dan Netbeans.

Sebagai permulaan, anda dapat men-*download* Android SDK dari <http://developer.android.com/sdk/index.html> atau anda bisa langsung melakukan *update* SDK dari IDE yang anda lakukan.

6.1 Melakukan *Download* dan *Install* SDK Android pada Eclipse.

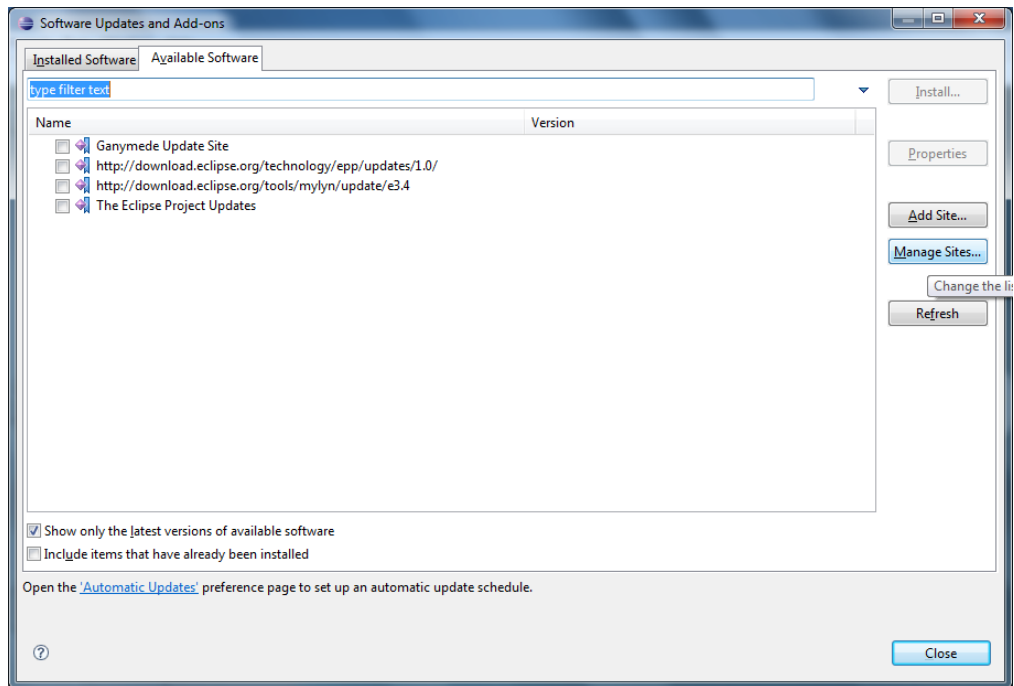
Download dan *install* SDK dilakukan apabila *programmer* belum memiliki atau memasang *plugin* SDK Android pada Eclipse, untuk melakukan *download* dan *install* berikut cara *download* dan *Install* SDK Android :

- Buka aplikasi Eclipse.
- Klik help > Software Update > Find and Install



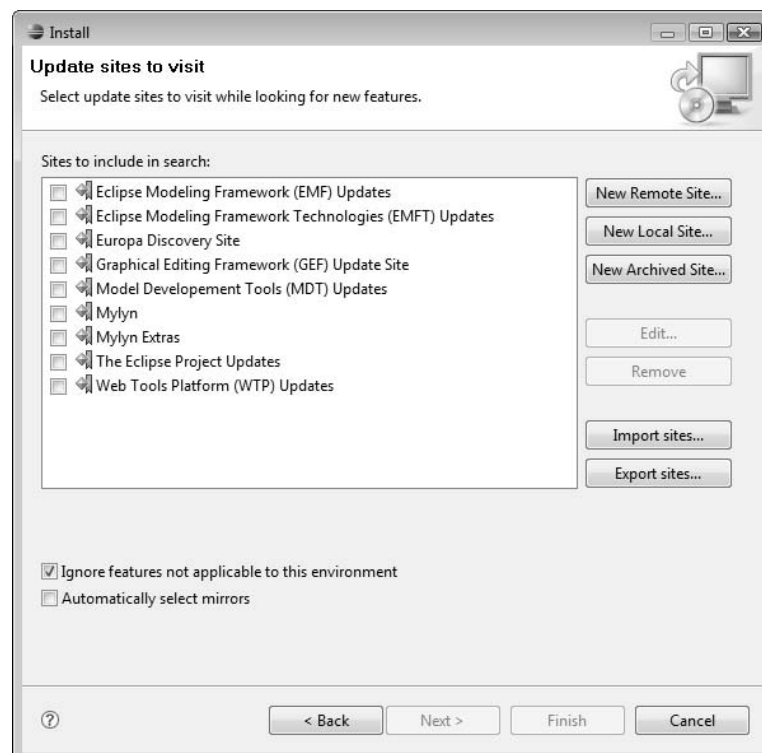
Gambar 6.1.1 jendela pembuka Eclipse

- Pada jendela *Software Update* atau *Update Install*, pada Eclipse versi Galileo terdapat dua pilihan *option* radio button anda pilih atau klik pada “Search or New Features to Install” sedangkan pada versi Gany Mayde akan muncul pilihan “Manage Sites” dan “Add Sites, kemudian pilih Add Sites untuk membuat Sites atau source baru” seperti pada gambar di bawah ini :



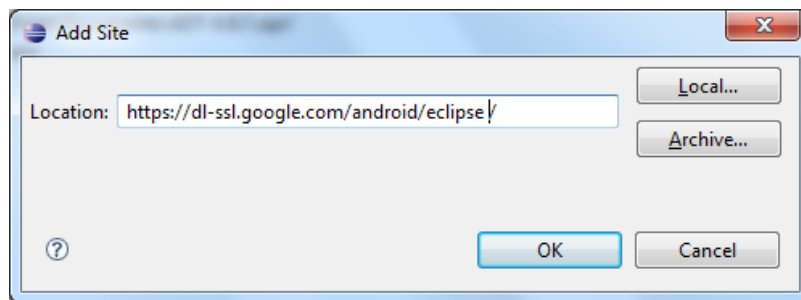
Gambar 6.1.2 jendela Software Update and Add-ons

- Sedangkan untuk versi Galileo akan tampil dialog install, untuk mendownload Android plugin anda klik pada bagian tombol *new remote site*, lihat gambar di bawah ini :



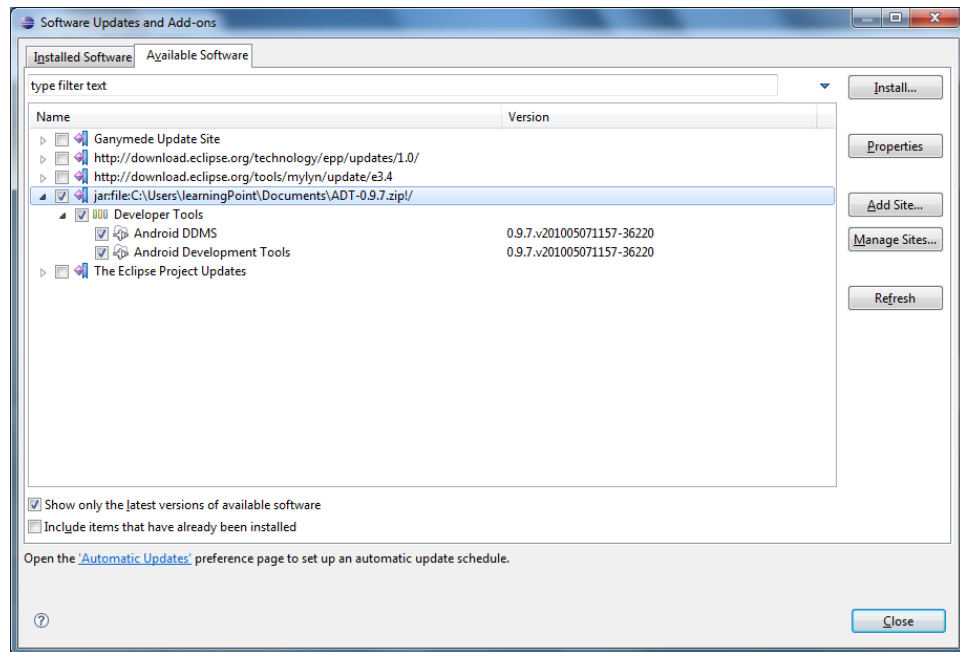
Gambar 6.1.3 jendela pilihan *site to visit*

- Pada tampilan berikutnya terdapat sebuah jendela dialog untuk mengisi *Field URL* dengan <https://dl-ssl.google.com/android/eclipse/>. Atau anda dapat mendownload file ADT dan memanggilnya menggunakan menu “Archive”. yang Kemudian klik OK, Untuk lebih jelasnya lihat gambar di bawah ini :



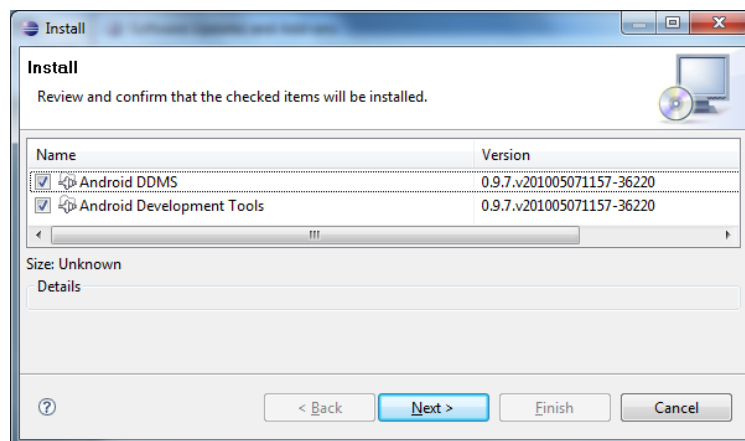
Gambar 6.1.4 jendela *New Update Site*

- Selanjutnya pada jendela *available software* terdapat *list* baru dengan berupa *site* baru dengan nama *Android Plugin*.



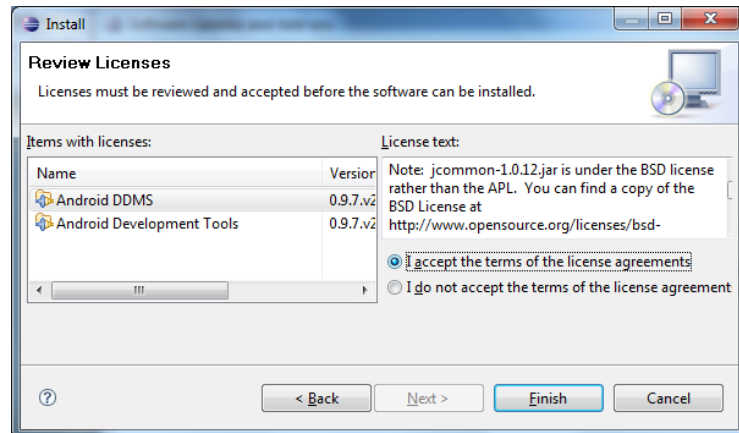
Gambar 6.1.5 list baru pada jendela *Update site to visit*

- Berikan cek list pada *Android Plugin* kemudian klik *install*
- Kemudian akan tampil jendela *Install*, pada jendela ini berikan ceklis pada item yang dibutuhkan *Android*, seperti pada gambar dibawah ini :



Gambar 6.1.6 kotak dialog *Install*

- Selanjutnya akan tampil kotak dialog *license agreement*, klik “*I accept the term of the license agreements*” kemudian klik tombol *finish*, lihat gambar di bawah ini :



Gambar 6.1.7 kotak dialog *Review License*

- Setelah selesai instalasi plugin, selanjutnya anda akan masuk pada pembahasan konfigurasi plugin.

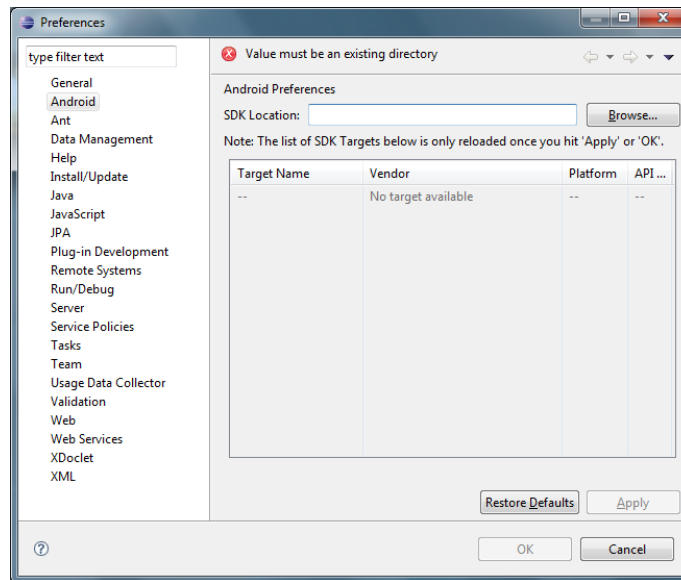
6.2 Melakukan Konfigurasi *Plugin* pada Eclipse.

Setelah anda selesai memasukkan atau memasang plugin pada Eclipse, anda mendapatkan pemberitahuan dan anda diminta untuk melakukan *restart* aplikasi Eclipse. Jika tidak ada pemberitahuan maka lakukan *restart* aplikasi segera. Melakukan restart akan membuat aplikasi Eclipse diberikan kesempatan untuk melakukan inisialisasi dan meyakinkan bahwa program tersebut telah mendapat pemasangan plugin Android.

Plugin Android untuk Eclipse di konfigurasi melalui jendela *preference* Eclipse

Tahapan melakukan konfigurasi adalah sebagai berikut :

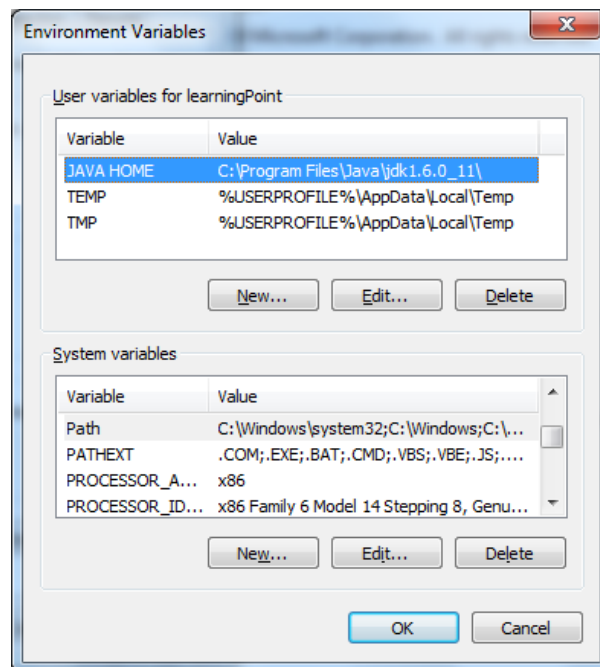
- Pada layar utama tampilan jendela Eclipse pilih menu *bar* Window > *Preference*.
- Selanjutnya tampil jendela *Preference*, klik Android pada menu sebelah kiri, kemudian pada sebelah kanan klik browse, lihat gambar di bawah ini :



Gambar 6.2.1 Kotak dialog *preference*

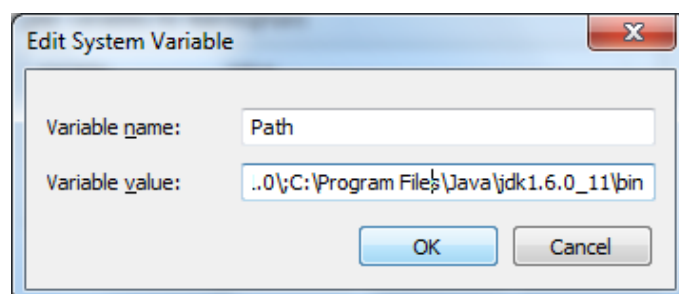
Pada tahap ini Eclipse mencari lokasi SDK untuk menjalankan emulator sebagai media aplikasi Android yang akan dibuat, kemudian klik *apply*.

- Pada tahap terakhir pada setingan SDK, yaitu pada PATH statement. Jika anda menggunakan windows lakukan langkah berikut : klik kanan pada *my computer*, pilih *properties* untuk membuka kotak dialog pada *system properties*, kemudian klik pada *advance tab*.
- Klik pada *environment variables* untuk menampilkan jendela *environment variables*, seperti terlihat pada gambar dibawah ini, anda bisa mengedit PATH *statement*



Gambar 6.2.2 *Environment Variables*

- Pada bagian bawah *System Variables*, cari dan temukan *variable* PATH kemudian klik ganda.
- Pada kotak dialog *Edit System Variables*, masukkan lokasi Android SDK lalu klik OK untuk menerapkannya. Dan klik OK lagi pada jendela *Environment Variables*.



Gambar 6.2.3 Setting Path

6.3 Melakukan Konfigurasi *Plugin* pada Netbeans.

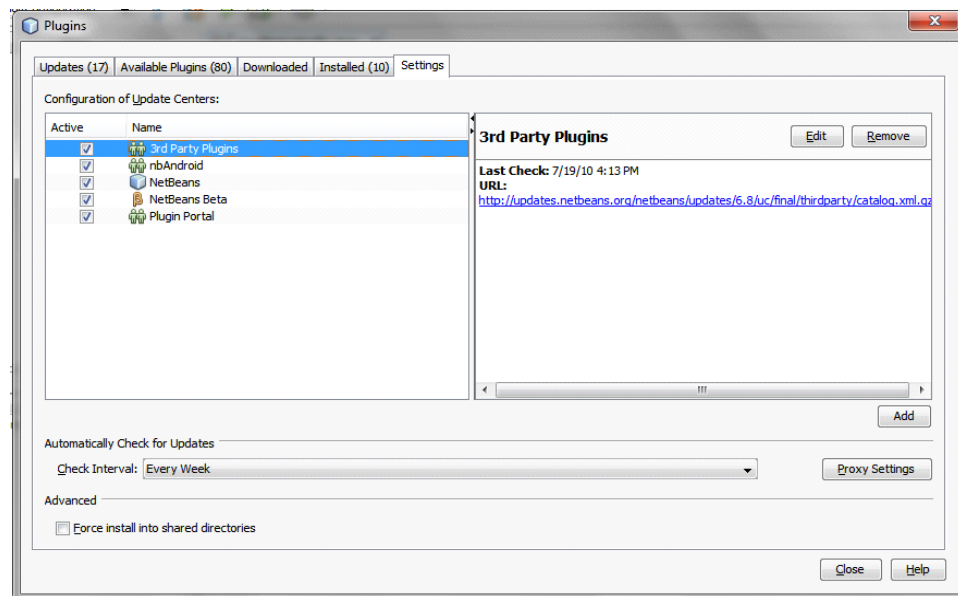
Untuk mempermudah instalasi *plugin* pada Netbeans, anda diharapkan untuk terlebih dahulu men-*download* Android SDK dari <http://developer.android.com/sdk/index.html>.

Apabila anda sudah menyelesaikan proses *download* Android SDK, maka anda dapat meng-*extract* Android SDK tersebut ke direktori yang anda kehendaki.

Pada nantinya direktori tersebut akan menjadi tempat acuan dari Netbeans pada saat membutuhkan *emulator* Android.

Apabila anda sudah selesai dengan proses *extract* anda, anda dapat langsung mengikuti langkah-langkah berikutnya:

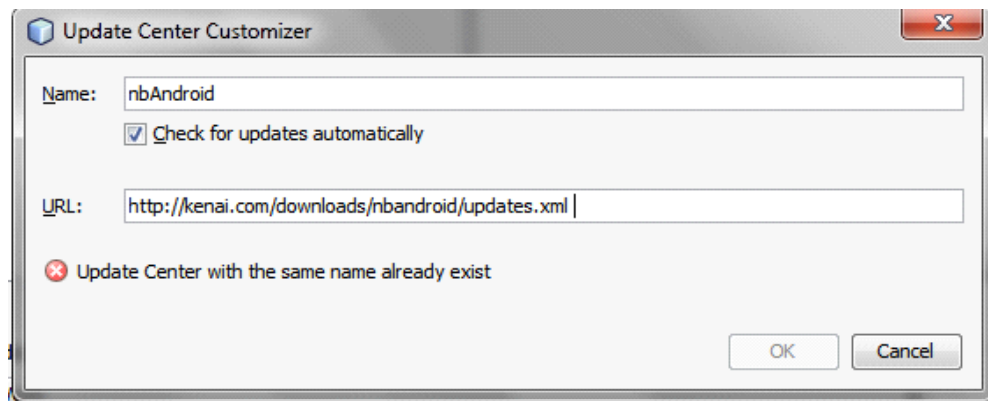
- Buka Netbeans anda, lalu klik Tools – Plugins – Settings



Gambar 6.3.1 Tampilan dari *window Tools*

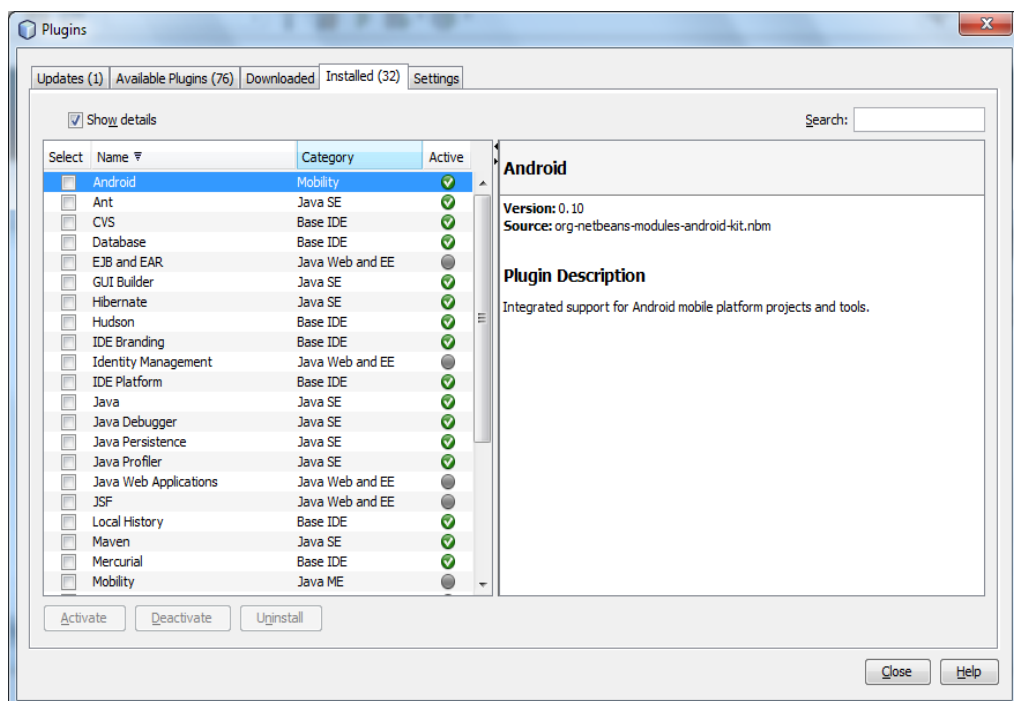
- Setelah itu teka tombol *Add*, pada *window Update Center Customizer* isikan nama yang anda kehendaki sebagai nama plugin Android anda dan masukkan URL tempat Netbeans akan mencari update.

Disini anda diharuskan memasukkan url <http://kenai.com/downloads/nbandroid/updates.xml>.



Gambar 6.3.2 tampilan *window Update Center Customizer*

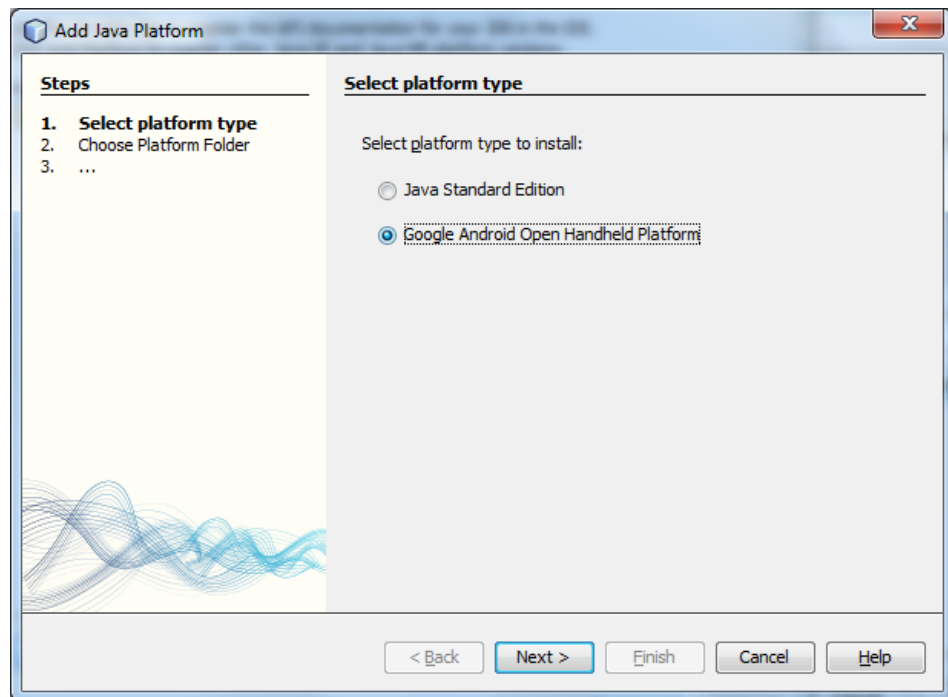
- Klik ok dan lanjutkan ke *window Available Plugins* dan carilah *update* yang tersedia untuk Android



Gambar 6.3.3 tampilan *Available Plugins*

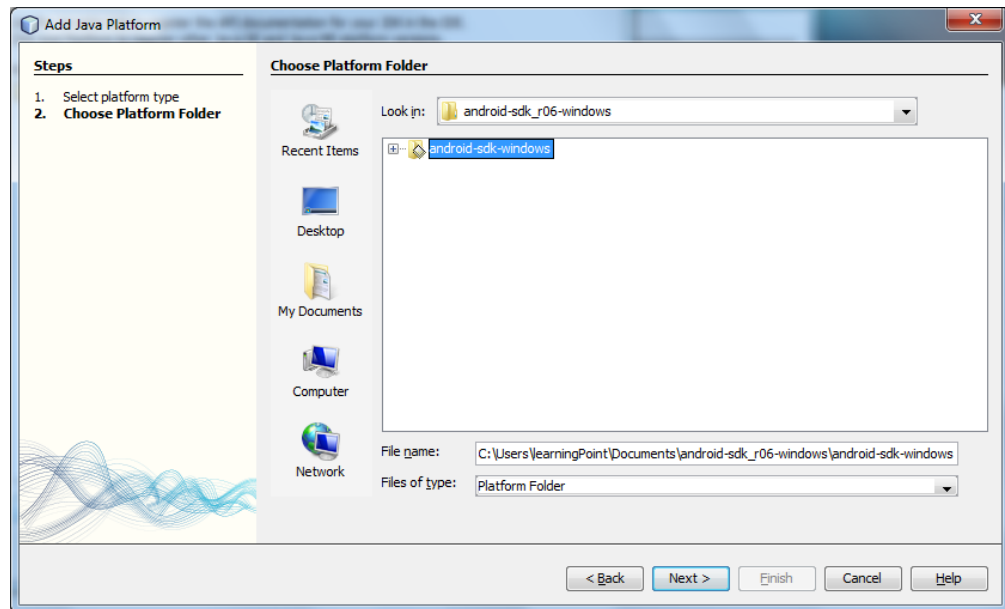
- Klik Install lalu ikuti langkah – langkah instalasi yang disediakan.

- Setelah selesai meng-*install update* untuk Android, maka hal selanjutnya yang harus dilakukan adalah menambah *platform* untuk Android pada *list platform* yang ada pada Netbeans
- Klik *Tools – Java Platform*, maka akan muncul tampilan *Java Platform Manager*.
- Klik *Add Platform*, lalu pilih *Google Android Open Handheld Platform*, lalu klik *Next*



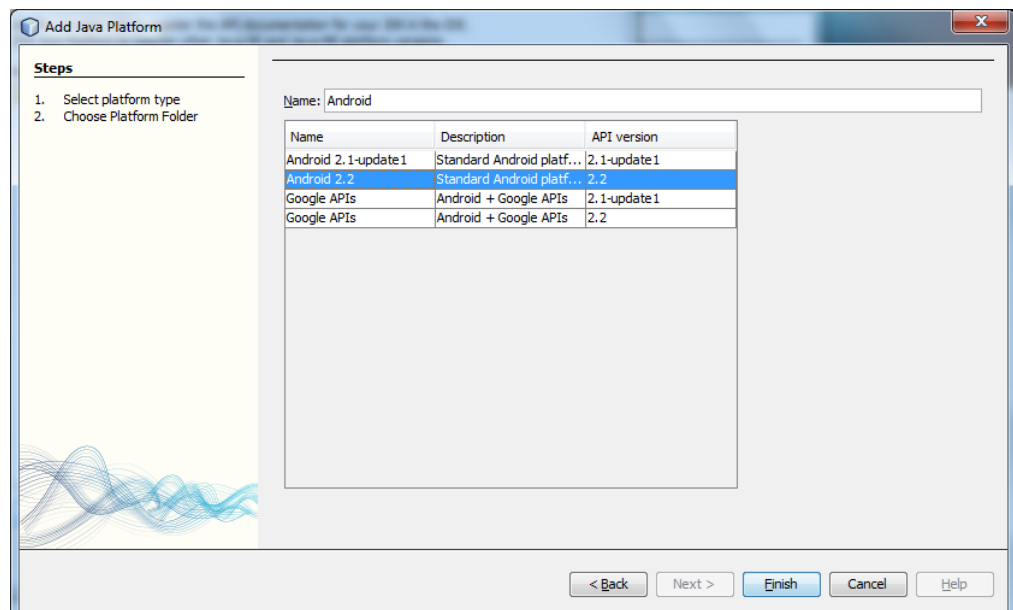
Gambar 6.3.4 tampilan *Add Platform*

- Kemudian masukkan direktori tempat di mana terdapat Android SDK, lalu klik *Next*



Gambar 6.3.5 tampilan Choose Platform Folder Android

- Kemudian pilih Android Platform yang tersedia pada SDK, seperti pada contoh gambar dibawah ini :



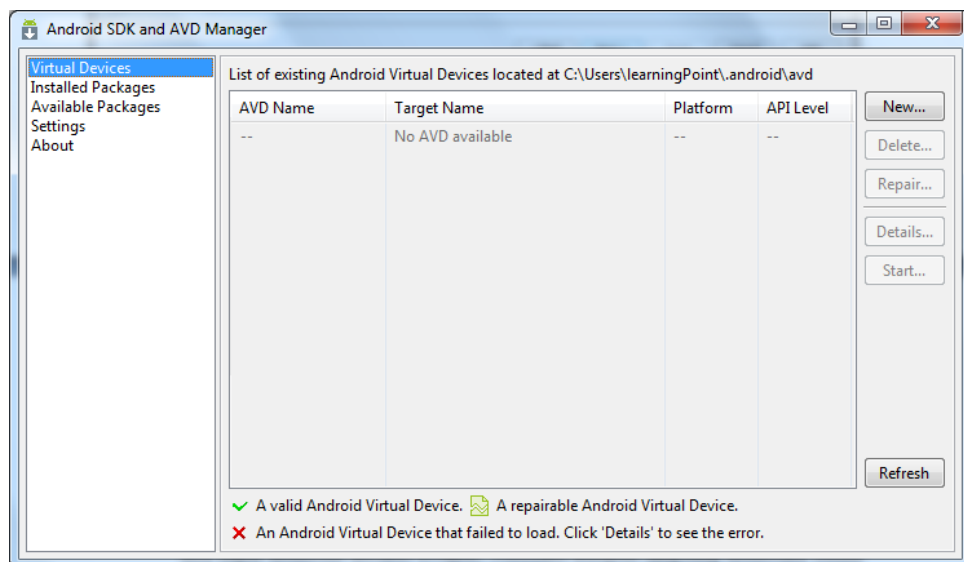
Gambar 6.3.6 tampilan Add Platform

- Untuk mengakhiri konfigurasi dan Instalasi *platform* pada Netbeans, pilih Finish.

6.4 Membuat AVD / Android Virtual Device pada SDK

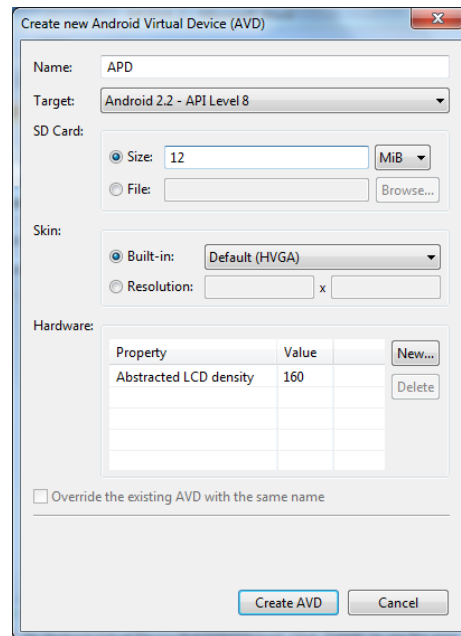
Agar emulator yang telah di-*setting* dapat digunakan ketika aplikasi dijalankan dibutuhkan sebuah Virtual Device pada SDK Android. Untuk membuat AVD ini kita dapat membuat melalui Eclipse, Netbeans maupun langsung membuka folder SDK tersebut dan menjalankan SDK Setup. Untuk langkah-langkah selanjutnya adalah sebagai berikut :

1. Setelah Android and AVD Manager tampil, anda dapat memilih Virtual Devices. Seperti pada gambar di bawah ini :



Gambar 6.4.1 Jendela Android SDK and AVD Manager

2. Kemudian untuk membuat AVD baru, dapat dipilih button “New”. Selanjutnya akan tampil Dialog Create New AVD.



Gambar 6.4.2 Dialog Create New AVD

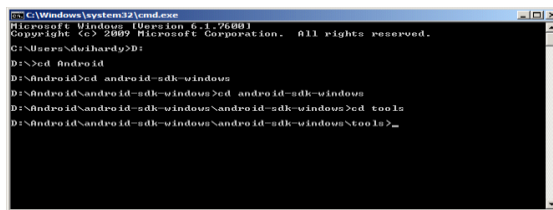
3. Selanjutnya, anda dapat memberikan nama AVD, memilih target API-Android yang tersedia dan filed lainnya, setelah itu klik “Create AVD”.
4. Maka Android Virtual Device pun telah dapat digunakan oleh Eclipse maupun Netbeans.

7. MEMBUAT *HELLO WORLD* DENGAN *COMMAND LINE INTERFACE*

Membuat aplikasi Android menggunakan Command Line memiliki beberapa tahapan diantaranya adalah :

7.1 Membuat Project Android

Untuk dapat membuat project android, kita harus masuk ke folder <folder android sdk windows>/tools dengan command prompt (cmd).



Gambar 7.1.1

- Untuk membuat project android , kita menggunakan

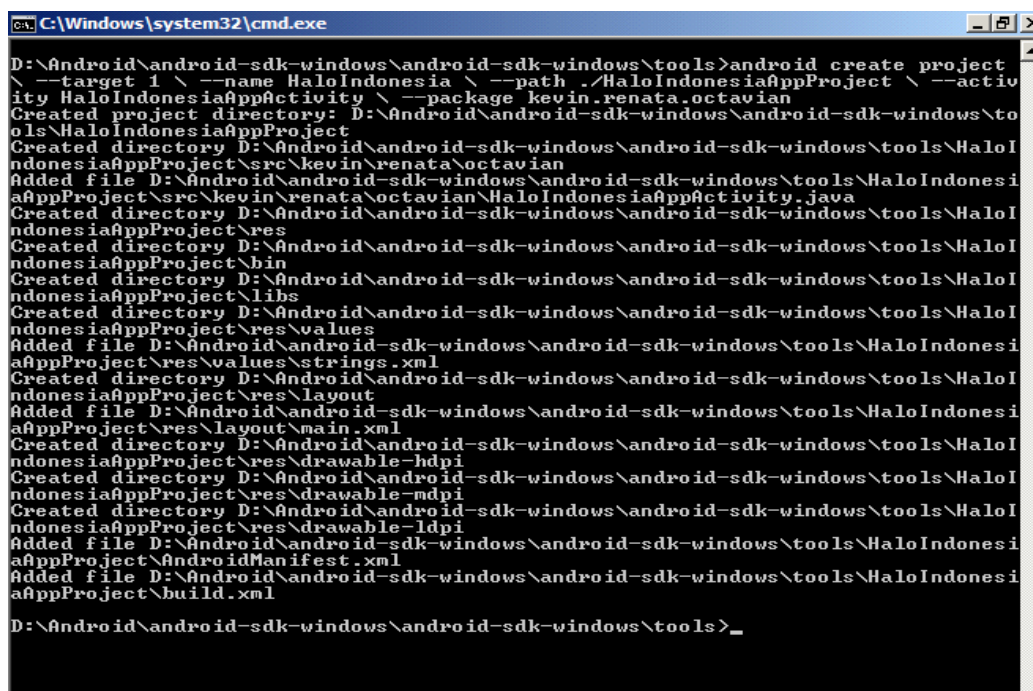
android create project \
--target <ID target> \
--name <nama project> \
--path /path/to/your/project \
--activity <nama activity> \
--package <nama package>

- target untuk membangun “bulid target” terhadap aplikasi. Hal ini sesuai dengan Android platform library (including any add-ons, such as Google APIs) untuk membuat suatu project. Untuk melihat daftar target tersedia dan ID yang sesuai mereka, lakukan : android list targets.

- name adalah nama untuk proyek Anda. Ini adalah opsional. Nama ini akan sama dengan nama file .apk

- path adalah lokasi direktori proyek. Jika direktori tidak ada, maka akan dibuat kan direktori baru.
- activity adalah nama default untuk [Activity](#) class. Class ini akan dibuat di `<path_to_your_project>/src/<your_package_namespace_path>/`.
- package adalah nama package untuk project anda, mengikuti aturan yang sama seperti untuk package dalam bahasa pemrograman Java.
- Misalnya kita akan membuat project android dengan nama project nya HaloIndonesia. Buka command prompt / terminal kemudian masuk ke `<folder android sdk>/tools`. Kemudian ketikkan sintaks berikut :

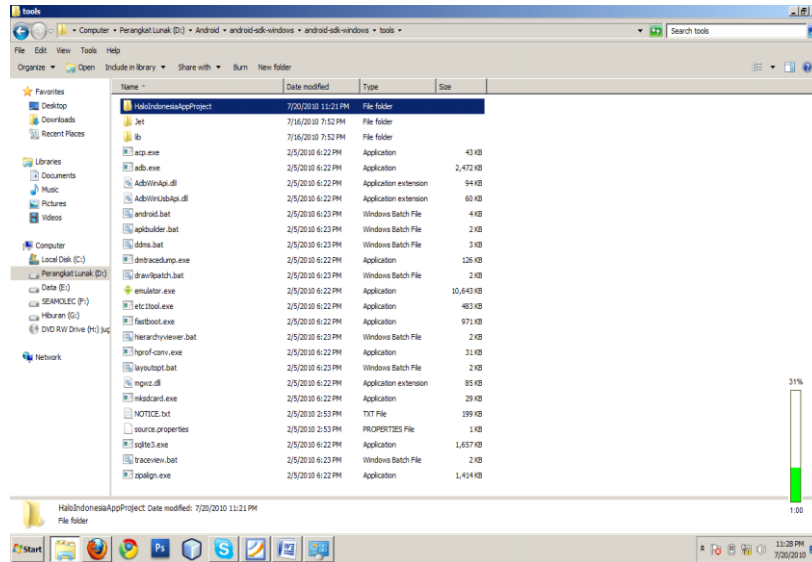
```
android create project \
--target 1 \
--name HaloIndonesia \
--path ./HaloIndonesiaAppProject \
--activity HaloIndonesiaAppActivity \
--package kevin.renata.octavian
```



```
C:\Windows\system32\cmd.exe
D:\Android\android-sdk-windows\android-sdk-windows\tools>android create project
--target 1 --name HaloIndonesia --path ./HaloIndonesiaAppProject --activ
ity HaloIndonesiaAppActivity --package kevin.renata.octavian
Created project directory: D:\Android\android-sdk-windows\android-sdk-windows\to
ols\HaloIndonesiaAppProject
Created directory D:\Android\android-sdk-windows\android-sdk-windows\tools\HaloI
ndonesiaAppProject\src\kevin\renata\octavian
Added file D:\Android\android-sdk-windows\android-sdk-windows\tools\HaloIndonesi
aAppProject\src\kevin\renata\octavian\HaloIndonesiaAppActivity.java
Created directory D:\Android\android-sdk-windows\android-sdk-windows\tools\HaloI
ndonesiaAppProject\res
Created directory D:\Android\android-sdk-windows\android-sdk-windows\tools\HaloI
ndonesiaAppProject\bin
Created directory D:\Android\android-sdk-windows\android-sdk-windows\tools\HaloI
ndonesiaAppProject\libs
Created directory D:\Android\android-sdk-windows\android-sdk-windows\tools\HaloI
ndonesiaAppProject\res\values
Added file D:\Android\android-sdk-windows\android-sdk-windows\tools\HaloIndonesi
aAppProject\res\values\strings.xml
Created directory D:\Android\android-sdk-windows\android-sdk-windows\tools\HaloI
ndonesiaAppProject\res\layout
Added file D:\Android\android-sdk-windows\android-sdk-windows\tools\HaloIndonesi
aAppProject\res\layout\main.xml
Created directory D:\Android\android-sdk-windows\android-sdk-windows\tools\HaloI
ndonesiaAppProject\res\drawable-hdpi
Created directory D:\Android\android-sdk-windows\android-sdk-windows\tools\HaloI
ndonesiaAppProject\res\drawable-mdpi
Created directory D:\Android\android-sdk-windows\android-sdk-windows\tools\HaloI
ndonesiaAppProject\res\drawable-ldpi
Added file D:\Android\android-sdk-windows\android-sdk-windows\tools\HaloIndonesi
aAppProject\AndroidManifest.xml
Added file D:\Android\android-sdk-windows\android-sdk-windows\tools\HaloIndonesi
aAppProject\build.xml
D:\Android\android-sdk-windows\android-sdk-windows\tools>
```

Gambar 7.1.2

- Apabila berhasil , maka akan muncul folder baru bernama HaloIndonesiaAppProject



Gambar 7.1.3

Di dalam folder HaloIndonesiaAppProject akan berisi :

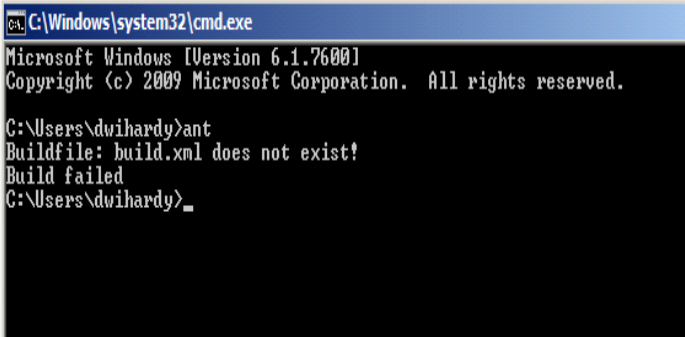
- AndroidManifest.xml - Aplikasi File manifest, disinkronisasikan ke Class Activity untuk proyek tersebut.
- build.xml – untuk membuild project.
- default.properties - Properties untuk mem build system. *Jangan mengubah file ini.*
- build.properties - Customizable properties untuk mem-build system. Anda dapat mengedit file ini untuk mengganti default membangun pengaturan yang digunakan oleh Ant dan memberikan pointer ke keystore Anda dan alias kunci sehingga membuat perangkat dapat menandatangani aplikasi Anda ketika dibangun dalam release mode.

- `src/your/package/namespace/ActivityName.java` – Class Activity yang ditentukan selama pembuatan project.
- `bin/` - Output direktori untuk mem-build script.
- `gen/` - berisi Ant-generated files, seperti `R.java`.
- `libs/` - berisi private libraries.
- `res/` - berisi project resources.
- `src/` - berisi source code.
- `tests/` - berisi salinan dari semua data di atas.

7.2 Build project

Untuk membuild aplikasi , terlebih dahulu kita harus memiliki Apache Ant versi terbaru. Jika belum memilikinya, silahkan unduh di <http://ant.apache.org/>. Jika anda telah memilikinya, pastikan set classpath telah dilakukan seperti pada pembahasan konfigurasi Plugin.

Untuk mengetahui apakah set classpath-nya berhasil atau tidak, buka command prompt kemudian ketikkan `ant` . Apabila gagal, maka akan muncul tampilan seperti berikut :



```

C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\dwihardy>ant
Buildfile: build.xml does not exist!
Build failed
C:\Users\dwihardy>_

```

Gambar 7.2.1

Untuk melakukan build aplikasi android, langkah yang harus dilakukan adalah buka command prompt kemudian masuk ke folder project android yang tadi (HaloIndonesiaAppProject). Kemudian ketikkan ant debug. Apabila berhasil akan muncul kotak dialog seperti di bawah ini dan pada folder bin akan muncul HaloIndonesiaAppProject-debug.apk.



```
C:\Windows\system32\cmd.exe
D:\Android\android-sdk-windows\android-sdk-windows\tools\HaloIndonesiaAppProject
>ant debug
Buildfile: D:\Android\android-sdk-windows\android-sdk-windows\tools\HaloIndonesiaAppProject\build.xml
[setup] Project Target: Android 2.1
[setup] API level: 7
[setup] WARNING: No minSdkVersion value set. Application will install on all
Android versions.

-compile-tested-if-test:

-dirs:
[echo] Creating output directories if needed...

-resource-src:
[echo] Generating R.java / Manifest.java from the resources...

-aidl:
[echo] Compiling aidl files into Java classes...

compile:
[javac] D:\Android\android-sdk-windows\android-sdk-windows\platforms\android
-7\templates\android_rules.xml:248: warning: 'includeantruntime' was not set, de
faulting to build.sysclasspath=last; set to false for repeatable builds
[javac] Compiling 1 source file to D:\Android\android-sdk-windows\android-sd
k-windows\tools\HaloIndonesiaAppProject\bin\classes

-dex:
[echo] Converting compiled files and external libraries into D:\Android\and
roid-sdk-windows\android-sdk-windows\tools\HaloIndonesiaAppProject\bin\classes.d
ex...
[echo]

-package-resources:
[echo] Packaging resources
[aptexcl] Creating full resource package...

-package-debug-sign:
[apkbuilder] Creating HaloIndonesia-debug-unaligned.apk and signing it with a de
bug key...
[apkbuilder] Using keystore: C:\Users\dwi Hardy\.android\debug.keystore

debug:
[echo] Running zip align on final apk...
[echo] Debug Package: D:\Android\android-sdk-windows\android-sdk-windows\to
ols\HaloIndonesiaAppProject\bin\HaloIndonesia-debug.apk

BUILD SUCCESSFUL
Total time: 6 seconds
D:\Android\android-sdk-windows\android-sdk-windows\tools\HaloIndonesiaAppProject
>
```

Gambar 7.2.2

7.3 Menjalankan aplikasi android

Membuat AVD (Android Virtual Device)

- Buka command prompt / terminal kemudian masuk ke (folder android sdk)/tools , kemudian ketikkan “*android list targets*”

```
C:\Windows\system32\cmd.exe
D:\Android\android-sdk-windows\android-sdk-windows\tools>android list target
Available Android targets:
id: 1 or "android-7"
  Name: Android 2.1
  Type: Platform
  API level: 7
  Revision: 1
  Skins: HUGA (default), QUGA, WQUGA400, WQUGA432, WUGA800, WUGA854
id: 2 or "Google Inc.:Google APIs:7"
  Name: Google APIs
  Type: Add-On
  Vendor: Google Inc.
  Revision: 1
  Description: Android + Google APIs
  Based on Android 2.1 (API level 7)
  Libraries:
    * com.google.android.maps (maps.jar)
      API for Google Maps
  Skins: WUGA854, WQUGA400, HUGA (default), WQUGA432, WUGA800, QUGA
D:\Android\android-sdk-windows\android-sdk-windows\tools>
```

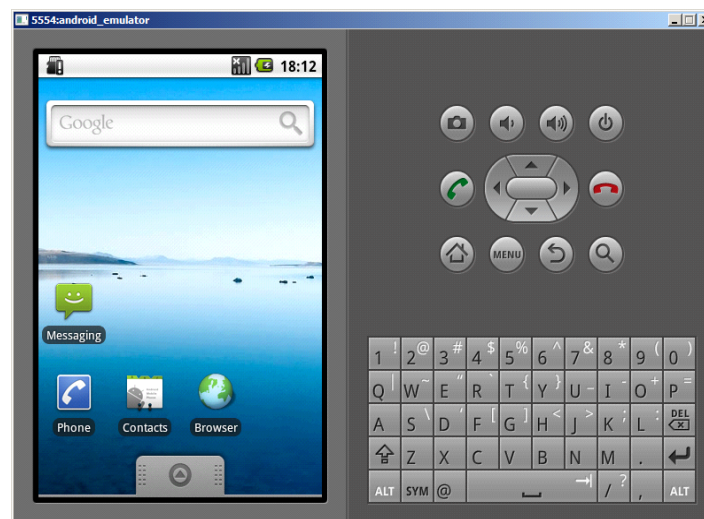
Gambar 7.3.1

- Kemudian untuk membuat AVD :
 - `android create avd --name <your_avd_name> --target <target_ID>` target_id diisi dengan id pada hasil output diatas.
 - Kemudian ketikkan sintak berikut pada *command prompt*: “*android create avd --name android_emulator --target 1*”.
 - Kemudian ditampilkan pertanyaan “*create a custom hardware profile*” . Jika anda merespon nya ketikkan yes jika tidak tekan *enter*.Jika berhasil akan muncul tampilan seperti berikut :

```
C:\Windows\system32\cmd.exe
D:\Android\android-sdk-windows\android-sdk-windows\tools>android create avd --na
me android_emulator --target 1
Android 2.1 is a basic Android platform.
Do you wish to create a custom hardware profile [no]
Created AVD 'android_emulator' based on Android 2.1, with the following hardware
config:
hw.lcd.density=160
D:\Android\android-sdk-windows\android-sdk-windows\tools>_
```

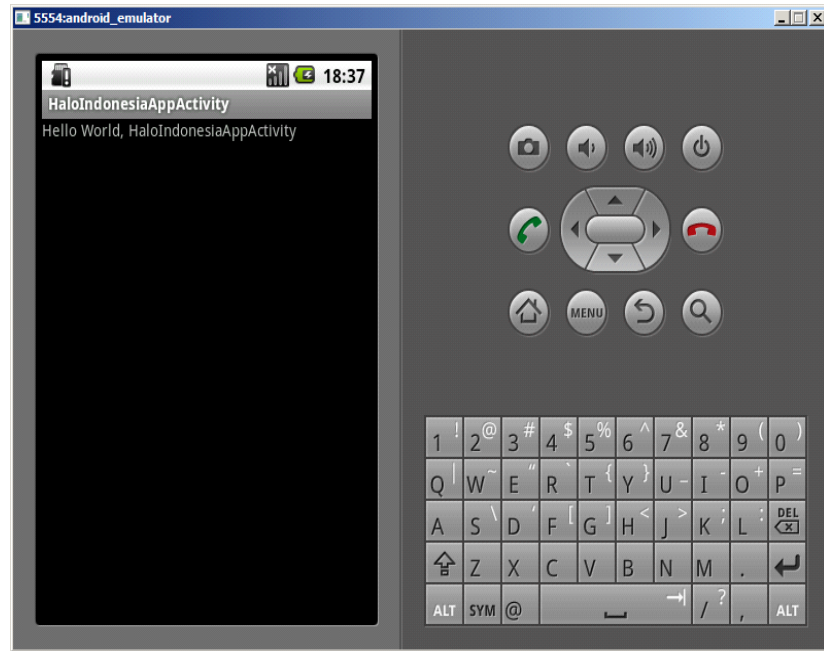
Gambar 7.3.2

- Menjalankan emulator
- buka command prompt atau terminal kemudian ketikkan kode “*emulator - avd android_emulator*” jika berhasil maka akan muncul emulator android seperti yang ditunjukkan oleh gambar 6.9: >> bisa juga 2.6.9 >> konsistensi



Gambar 7.3.3

- Menginstall aplikasi android di emulator
- Untuk menjalankan:adb install /path/to/your/application.apk, buka command prompt atau terminal yang baru kemudian ketikkan kode”adb install HaloIndonesiaAppProject/bin/HaloIndonesia-debug.apk” apabila berhasil maka akan muncul seperti berikut :



Gambar 7.3.4

8. Membuat *Hello World* dengan Eclipse

Pada dasarnya, kita akan lebih sering, atau bahkan selalu menggunakan IDE pada saat *men-develop* aplikasi Android.

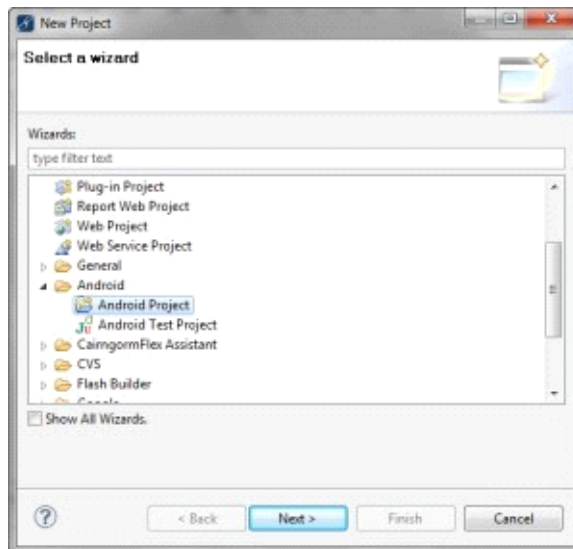
Maka dari itu, kita akan membahas tentang bagaimana membuat project di Eclipse, yang pada kali ini kita akan membuat sebuah program Hello World sederhana.

8.1 Membuat Hello World

1. Jalankan Eclipse, pilih File -> New -> Project.

Jika plugin ADT sudah ter-install, akan tampil di layar sebuah dialog dengan pilihan folder bernama “Android” yang di dalamnya terdapat informasi mengenai “Android Project” yang akan Anda bangun.

2. Pilih Android Project, dan klik tombol Next.



Gambar 8.1.1

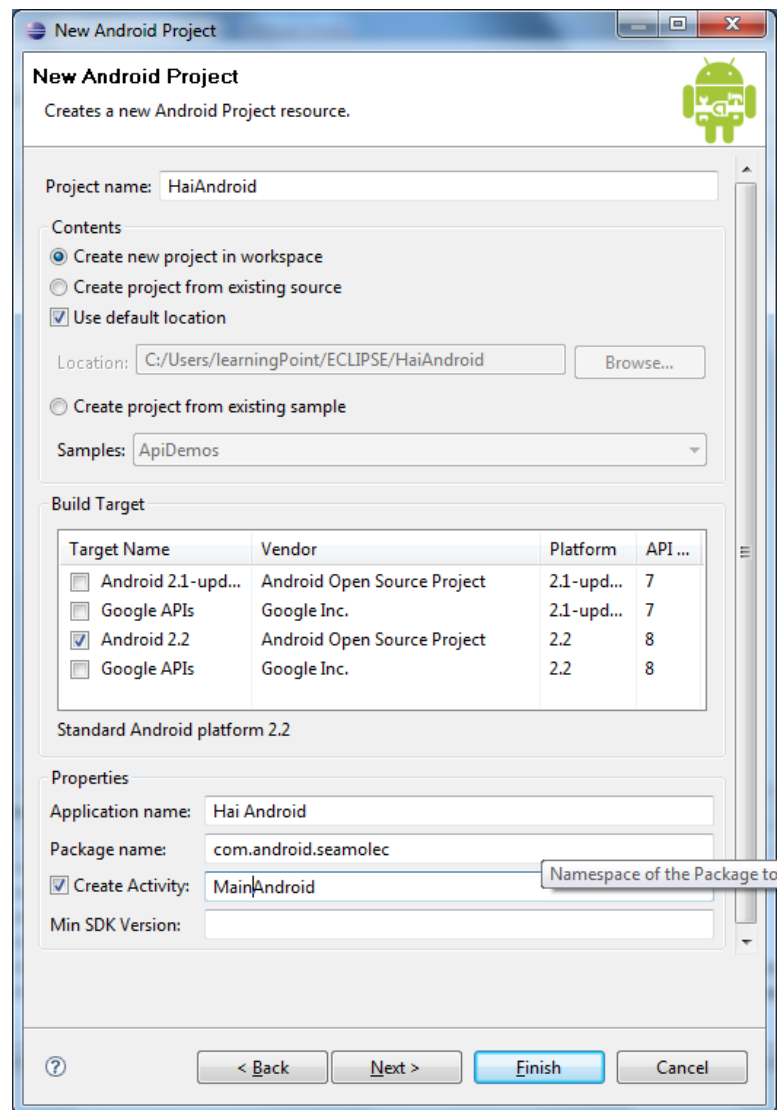
3. Masukkan beberapa informasi di dalamnya dengan informasi berikut:

Project name: HaiAndroid

Application name: Hai, Android

Package name: com.android.seamolec

Create Activity: MainAndroid



Gambar 8.1.2

Jangan lupa untuk memilih SDK Android minimal versi 2 pada bagian “Build Target”. Kemudian klik Finish.

MainAndroid.java, yang ada di *seamolec > src > com.android.seamolec*. Berikut isi dari file tersebut:

```
package com.android.seamolec;
```

```
import android.app.Activity;

import android.os.Bundle;


public class MainAndroid extends Activity {

    /** Called when the activity is first created. */

    @Override

    public void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.main);

    }

}
```

Ubah kode program di atas dengan kode berikut:

```
package com.android.seamolec;


import android.app.Activity;

import android.os.Bundle;

import android.widget.TextView;


public class MainAndroid 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("Hai Android!");  
  
    setContentView(tv);  
  
}  
}
```

Setelah Anda mengikuti langkah-langkah diatas, saat ini Anda sudah selesai membangun aplikasi “Hai Android” menggunakan Android. Langkah selanjutnya adalah menjalankan aplikasi yang baru saja Anda bangun.

1. Pilih menu Run -> Run.
2. Pilih bagian Android Application.

Eclipse akan melakukan kompilasi aplikasi, dan kemudian akan menampilkan hasilnya pada emulator Android.



Gambar 8.1.3 Hasil Emulator Android

8.2 Membuat Hello Word dengan XML

Setelah anda membuat "Hello, World" dengan IDE Eclipse / Netbeans akan muncul pada folder project UI Layout secara langsung. Android menyediakan model pembangunan alternatif UI: file layout berbasis XML. Cara termudah untuk menjelaskan konsep ini adalah untuk menunjukkan contoh. Berikut ini adalah tata letak file XML yang identik dalam perilaku dengan contoh pemrograman yang dibangun:

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

<TextView xmlns: android = "http://schemas.android.com/apk/res/android"

    android: id = "@ + id / textview"

    android: layout_width = "fill_parent"

    android: layout_height = "fill_parent"

    android: text = "@ string / halo" />
```

Struktur umum dari file XML Android adalah: *tree of XML elements*, dimana setiap node adalah nama kelas View. Anda dapat menggunakan nama dari setiap kelas yang sifatnya global. Lihat sebagai unsur dalam tata letak XML Anda, termasuk kelas custom view yang ditetapkan dalam kode. Struktur ini memudahkan untuk membangun UIs, menggunakan struktur yang lebih sederhana dan sintaks yang akan digunakan dalam tata letak program. Model ini terinspirasi oleh model pengembangan web, di mana dapat memisahkan presentasi dari aplikasi (UI), dari logika aplikasi yang digunakan untuk mengambil dan mengisi data.

Pada contoh XML di atas, ada salah satu elemen Lihat: yang TextView, yang memiliki lima atribut XML. Berikut adalah ringkasan dari apa yang mereka maksud:

(nomor dan judul table, di posisi atas)

Atribut	Pengertian
xmlns: android	Ini merupakan deklarasi namespace XML yang menginformasikan alat Android bahwa yang mengacu pada atribut umum didefinisikan dalam ruang nama Android. Tag terluar di setiap file layout Android harus memiliki atribut ini.
android: id	Atribut ini menetapkan pengenalan unik untuk elemen TextView. Anda dapat menggunakan ID yang ditugaskan untuk referensi ini Lihat dari kode sumber Anda atau dari deklarasi XML sumber daya lainnya.
android: layout_width	Atribut ini mendefinisikan berapa banyak lebar yang tersedia pada layar ini Lihat harus mengkonsumsi. Dalam hal ini, itu adalah Lihat sehingga Anda hanya ingin mengambil seluruh layar, yang adalah apa nilai "fill_parent" berarti.
android: layout_height	Ini seperti android: layout_width, kecuali bahwa itu merujuk pada tinggi layar yang tersedia.
android: text	Ini menetapkan bahwa teks TextView harus ditampilkan. Dalam contoh ini, Anda menggunakan sumber daya string, bukan nilai string keras-kode. Halo string didefinisikan di res / nilai / file strings.xml. Ini adalah praktek yang disarankan untuk memasukkan string ke aplikasi Anda, karena membuat lokalisasi aplikasi Anda ke bahasa lain anggun, tanpa perlu susah-perubahan kode ke file layout. Untuk informasi lebih lanjut, lihat Sumber dan Internasionalisasi.

File-file ini termasuk layout XML di dalam folder res /layout/direktori pada proyek yang telah dibuat. "res" adalah singkatan dari "resources" dan direktori yang berisi semua

aktiva non-kode yang dibutuhkan aplikasi / program yang telah dibuat. Selain file layout, resource juga mencakup file seperti gambar, suara, dan string lokal.

Plugin Eclipse secara otomatis menciptakan salah satu layout file-file yaitu main.xml. Pada aplikasi “Hello World” yang baru saja selesai, file ini diabaikan dan menciptakan sebuah layout pemrograman. Ini dimaksudkan untuk mengajar lebih lanjut tentang kerangka Android, tetapi harus hampir selalu mendefinisikan layout dalam file XML bukannya dalam kode. Prosedur berikut ini akan menginstruksikan bagaimana untuk mengubah aplikasi yang sudah ada, untuk menggunakan layout XML.

1. Dalam Paket Explorer di Eclipse, buka folder /res/ layout, dan buka main.xml. Ubah isi file dengan XML berikut :

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

<TextView xmlns: android = "http://schemas.android.com/apk/res/android"

    android: id = "@+id/textview"

    android: layout_width = "fill_parent"

    android: layout_height = "fill_parent"

    android: text = "@ string/app_name" />
```

Simpan file.

2. Di dalam folder res/nilai/, buka file strings.xml. Di sinilah file harus menyimpan semua string teks standar untuk interface/tampilan pengguna. Jika menggunakan Eclipse, maka ADT akan mulai dengan dua string, *hello* dan *app_name*. Ubah “hello” ke teks yang lain. Contoh "Halo, saya Android!". Untuk seluruh file akan terlihat seperti ini:

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

<resources>
```

```
<string name="halo"> Hello, Android ! </String >
<string name="app_name"> Assalamualaikum, saya android </string>
</ resources >
```

3. Sekarang membuka dan memodifikasi kelas HelloAndroid Anda menggunakan layout XML.

Ubah isi file agar seperti ini:

```
package com.example.helloandroid;

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

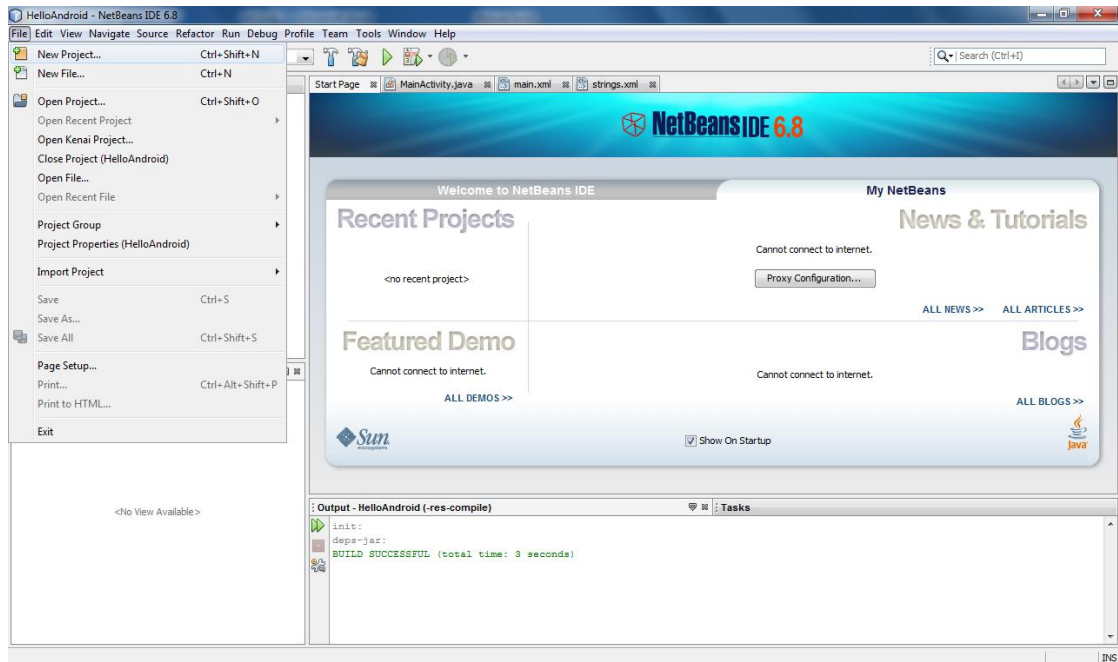
public class HelloAndroid extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```

Sekarang kembali menjalankan aplikasi, jika telah selesai membuat semua konfigurasi yang perlu dilakukan dengan cara klik ikon panah hijau untuk menjalankan aplikasi, atau pilih **Run > Run History > Android Activity**.

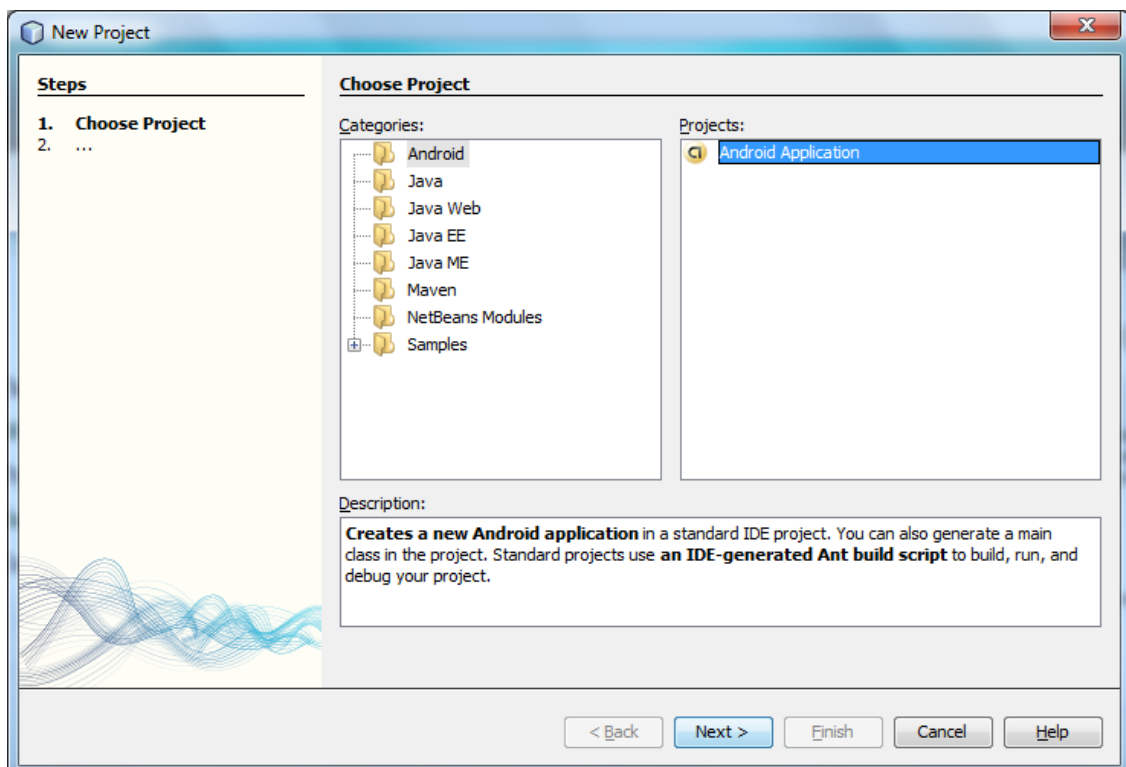
8.3 Membuat Hello Word dengan Netbeans

Membuat Android project baru

Sekarang, Pilih File -> New Project -> Kemudian akan tampil Dialog New Project, Pili Categories Android, yang selanjutnya akan menampilkan entri baru untuk Android :

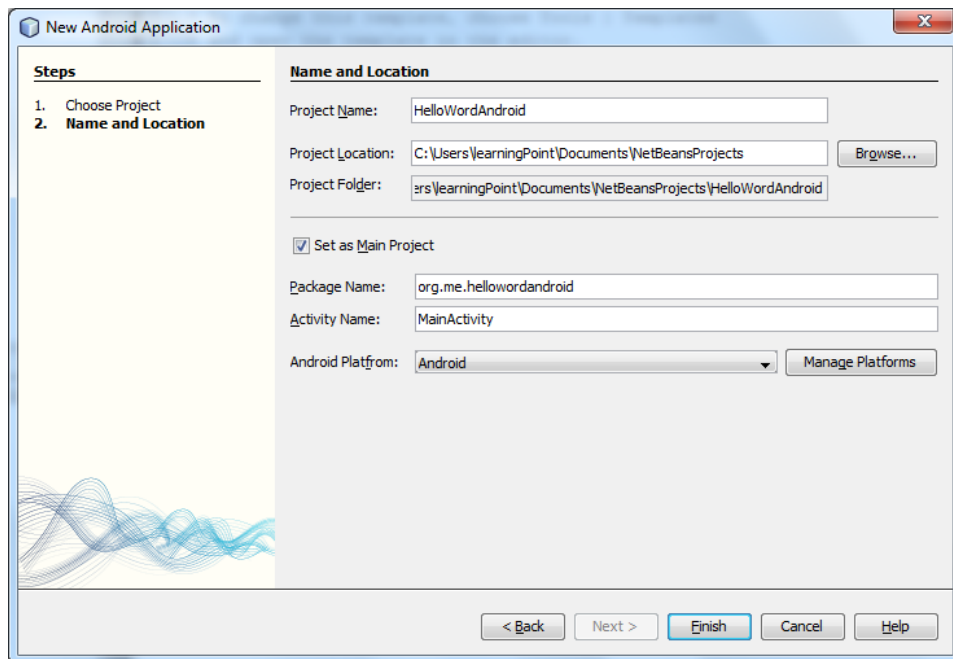


Gambar 8.3.1



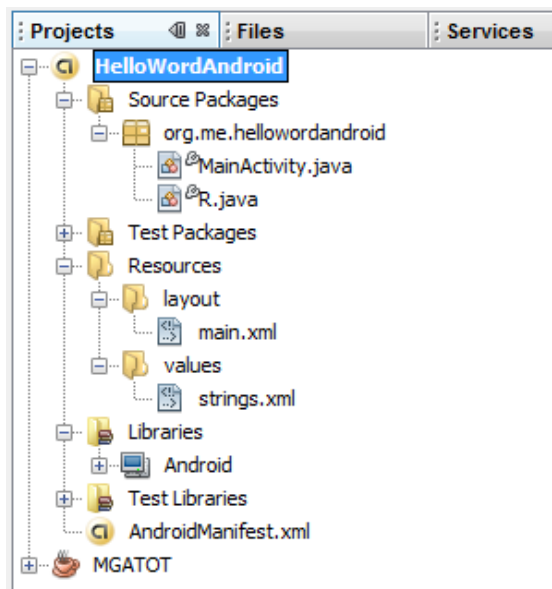
Gambar 8.3.2

Kemudian untuk melanjutkan pembuatan Project baru, pilih Next untuk mengubah Nama Aplikasi Android yang ingin dibuat atau tetap melanjutkan / default dengan memilih Next.



Gambar 8.3.3

Anda dapat melihat Struktur project yang baru dibuat, seperti pada gambar di bawah ini :



Gambar 8.3.4

Untuk isi dari File MainActivity.java, seperti yang ditunjukkan di bawah ini :

```
/*  
  
 * To change this template, choose Tools | Templates  
 * and open the template in the editor.  
 */  
  
package org.me.helloworldandroid;  
  
import android.app.Activity;  
import android.os.Bundle;  
  
/**  
 *  
 * @author PerajutHati  
 */  
public class MainActivity extends Activity {
```

```
/** Called when the activity is first created. */  
  
@Override  
  
public void onCreate(Bundle savedInstanceState) {  
  
    super.onCreate(savedInstanceState);  
  
    // TODO: Add your GUI initialization code here  
  
}  
  
}
```

Untuk Menampilkan Tulisan Hello Word pada aplikasi, tambahkan :

```
setContentView(R.layout.main);
```

Seharusnya Script diatas telah dihasilkan oleh plugin, karena file main.xml telah berisi konfigurasi tampilan utama yang akan ditampilkan. Jika tidak ada, maka dapat dibuat dengan menambahkan Script dibawah ini :

```
TextView tv = new TextView(this);  
  
tv.setText("Hello, Android");  
  
setContentView(tv);
```

Untuk Hasil File MainActivity.java yang telah di ubah secara keseluruhan adalah sebagai berikut :

```
package org.me.androidapplication1;

import android.app.Activity;

import android.os.Bundle;

import android.widget.TextView;

/**
 *
 * @author amit
 */
public class MainActivity 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, Android");

        setContentView(tv);
    }
}
```

```
}  
  
}
```

Kemudian jalankan / Run, maka akan muncul seperti tampilan pada gambar xxx :



Gambar 8.3.5