

MATERI 1

PENDAHULUAN

- **Alasan Membangun Aplikasi Android**
 - **Dasar Pemrograman Android**
 - **Fitur Perangkat Keras Android**
 - **Fitur Perangkat Lunak Android**
-

1.1. Alasan Membangun Aplikasi Android

Terdapat beberapa alasan mengapa Android berkembang dengan pesat dan kenapa *developer* sebaiknya membangun Aplikasi Android.

a. Market Share

Developer memiliki kesempatan untuk mengembangkan aplikasi untuk pasar yang relatif baru dan sedang berkembang pesat. Keberadaan *Android Market*akan menempatkan aplikasi yang dibuat oleh *developer*langsung kepada pengguna.Pengguna tidak perlu mencari melalui internet untuk menemukan dan meng-*install* aplikasi yang dibuat oleh *developer*.Pengguna cukup membuka*Android Market*yang sudah ter-*install* di perangkat android pengguna dan mengakses aplikasi yang pengguna butuhkan dan meng-*install*kedalam perangkat android pengguna.

b. Time to Market

Terdapat Android APIs (*Application Programming Interfaces*) yang memudahkan *developer*untuk membangun aplikasi dengan mudah dan waktu yang singkat.

c. Open Platform

Sistem operasi Android merupakan platform terbuka.Sehingga tidak merujuk pada suatu perusahaan *hardware*atau suatu *provider*. Platform terbuka android memungkinkan perkembangan *market*dengan sangat cepat, karena semua perusahaan *hardware*dan *provider*dapat membuat dan menjual perangkat android.Source codeandroid dapat di akses melalui at <http://source.android.com>, untuk digunakan dan dimodifikasi sesuai kebutuhan perusahaan *hardware*, *provider* atau *developer*aplikasi.

d. Cross Compatibility

Android dapat berjalan di berbagai perangkat dengan ukuran dan resolusi layar yang berbeda.Android memiliki fitur yang membantu *developer*untuk mengembangkan aplikasi yang *compatible*untuk berbagai perangkat (*cross-compatible application*). Google memiliki *feature detection*yang mengatur agar aplikasi yang dikembangkan oleh *developer*hanya berjalan di perangkat yang *compatible*. Sebagai contoh, jika suatu aplikasi membutuhkan

kamera depan (*front-facing camera*) maka hanya perangkat android dengan kamera depan yang akan dapat melihat aplikasi tersebut di *Android Market*.

Suatu perangkat android harus mengikuti suatu aturan untuk mendapatkan sertifikat kesesuaian terhadap Android Market untuk memastikan aplikasi dapat berjalan diatasnya, diantaranya keberadaan: (a) Kamera, (b) Kompas, (c) GPS (*Global Positioning System*), (d) *Bluetooth Transceiver*.

e. *Mashup Capability*

Mashup Capability merupakan kemampuan untuk mengkombinasikan dua atau lebih layanan untuk mengembangkan suatu aplikasi. Sebagai contoh, *developer* dapat membuat *mashup* dengan menggunakan kamera dan GPS, sehingga terbangun aplikasi pengambilan photo dengan dilengkapi secara tepat lokasi pengambilannya.

Beberapa contoh lain *mashup* yang mungkin dikembangkan dalam suatu aplikasi, yaitu:

1. *Geolocation* dan *Social Networking*

Suatu aplikasi yang secara otomatis melakukan *tweet* posisi pengguna setiap 10 menit.

2. *Geolocation* dan *Gaming*

Location-based gaming merupakan model game yang saat ini popular. Game memiliki *background services* untuk mengecek posisi pengguna dan membandingnya dengan posisi pengguna yang lain di area yang sama. Jika jarak antara pengguna kurang dari 1 km misal, maka akan diberikan notifikasi dan memungkinkan mereka untuk bertarung melalui game tersebut.

1.2. Dasar Pemrograman Android

Membangun Aplikasi Android sangat mudah, *default* untuk bahasa pemrograman yang digunakan oleh Android adalah java. Tidak semua fitur java yang ada digunakan akan tetapi sebagian dari java yang sering disebut dengan *Dalvik Virtual Machine*. Beberapa bagian kecil dari *framework* Android menggunakan bahasa XML untuk *scripting*-nya.

a. *Activity*

Activity merupakan *container* untuk *User Interface (UI)*. Sebuah Aplikasi Android terbangun dari satu atau beberapa *Activity*.

b. *Intents*

Intent merupakan sistem pesan utama yang menjalankan Android. *Intent* terdiri dari *Action* yang harus dijalankan (Tampil, Ubah, Dial, dll) dan Data.

Intent digunakan untuk memulai aktivitas dan komunikasi antar bagian dari sistem Android. Suatu aplikasi dapat mengirimkan atau menerima *intent*.

1. Mengirimkan pesan dengan *Intent*

Ketika suatu aplikasi mengirimkan suatu *intent*, aplikasi mengirimkan pesan memberitahukan Android untuk menjalankan sesuatu. Misal: memberitahukan Android untuk menjalankan suatu *Activity* dari aplikasi atau menjalankan aplikasi lain.

2. Mendaftarkan *Intent Receiver*

Mengirimkan pesan dengan *Intent* bukan berarti sesuatu selanjutnya akan terjadi secara otomatis. Aplikasi harus mendaftarkan *intent receiver* yang memperhatikan *intent* dan memberitahukan Android apa yang harus dilakukan.

Contoh, jika pengguna menekan lama suatu *image* di *image gallery*, maka akan memunculkan *context menu* yang berkaitan dengan *image sharing*. Dikarenakan terdapat beberapa *receivers* yang didaftarkan berkaitan dengan *image sharing* (*email*, *messaging*, *Bluetooth*, dll) maka *context menu* dimunculkan untuk memberikan opsi keputusan kepada pengguna.

c. *Cursorless Controls*

Perangkat Android menggunakan jari pengguna sebagai *input*. Jika pada komputer biasa digunakan pointer mouse, kemudian *left-click* untuk berinteraksi dengan suatu tombol misal, maka pengguna Android dapat menggunakan jarinya langsung ke tombol. Untuk mengantikan *right-click*, pengguna Android dapat menekankan jarinya agak lama ke layar, kemudian *context menu* akan muncul.

d. *Views dan Widgets*

View merupakan elemen dasar UI (*User Interface*). Sebagai contoh area kotak pada layar yang bertanggung jawab untuk tampilan dan menerima event (*event handling*). Beberapa contoh *view* antara lain: (1) *ContextMenu*, (2) *Menu*, (3) *View*, (4) *SurfaceView*.

Widgets adalah elemen UI yang lebih canggih. Merupakan control untuk interaksi antara sistem dengan pengguna. Beberapa contoh *Widgets* antara lain: (1) *Button*, (2) *CheckBox*, (3) *DatePicker*, (4) *DigitalClock*, (5) *Gallery*, (6) *FrameLayout*, (7) *ImageView*, (8) *RelativeLayout*, (9) *PopupWindow*.

Untuk mempelajari *widgets* lebih lanjut dapat dilihat melalui

url: <http://developer.android.com/reference/android/widget/package-summary.html>

e. *Asynchronous Calls*

Android memiliki sebuah class yang *AsyncTask* yang memungkinkan aplikasi menjalankan beberapa operasi pada waktu yang bersamaan, tanpa harus mengatur bagaimana *thread* berjalan secara khusus. *AsyncTask* memungkinkan developer untuk membangun model program yang bersih untuk proses-proses asinkron.

Proses asinkron biasa digunakan untuk proses-proses yang membutuhkan waktu lama, misal: *Network Communication (Internet)*, *Media Processing*, dan berbagai proses lain yang mengharuskan pengguna menunggu. Jika pengguna harus menunggu, maka dapat digunakan proses asinkron untuk menampilkan UI yang memberitahukan pengguna apa yang terjadi.

f. *Background Services*

Services merupakan aplikasi yang berjalan di belakang dan tidak terlalu penting memiliki UI, sebagian contoh: anti-virus. Sebagian besar pemutar music (*music player*) dari Android

Market berjalan sebagai *Background Services*, yang mana pengguna dapat mendengarkan musik sambil mengecek e-mail atau melakukan tugas lain yang membutuhkan penggunaan tampilan layar.

1.3. Fitur Perangkat Keras Android

Perangkat Android memiliki beberapa fitur perangkat keras didalamnya, yang dapat dimanfaatkan *developer* dalam membangun aplikasi.

a. *Touchscreen*

Perangkat Android memiliki fitur layar sentuh (*touchscreen*) yang memberikan beberapa kemungkinan bagi pengguna untuk berinteraksi dengan aplikasi dengan menggunakan jari. Pengguna dapat melakukan *swipe*, *flip*, *drag*, dan *pinch* untuk *zoom*. Android juga mendukung *multitouch* yang berarti keseluruhan layar dapat disentuh dengan satu atau lebih jari pada saat yang bersamaan.

b. GPS

Sistem operasi Android mendukung GPS yang memungkinkan *developer* untuk mengakses lokasi pengguna. Contoh aplikasi yang memanfaatkan GPS adalah Aplikasi Peta (*Map*) yang menunjukkan lokasi pengguna dan memberikan petunjuk untuk menuju suatu lokasi.

c. *Accelerometer*

Android mendukung *Accelerometer*, yaitu perangkat yang digunakan untuk mengukur percepatan. *Accelerometer* dapat memberitahukan apabila suatu Perangkat Android bergerak, atau terguncang, atau berbalik arah posisinya.

d. SD Card

Android memiliki fitur yang memungkinkan pengguna atau aplikasi untuk mengakses (menyimpan atau membuka) file pada SD Card. SD Card merupakan media penyimpanan medium yang digunakan Perangkat Android dan beberapa perangkat *mobile* lain non Android sebagai media penyimpanan.

1.4. Fitur Perangkat Lunak Android

Android memiliki banyak fitur perangkat lunak yang dapat digunakan oleh *developer* dalam mengembangkan aplikasi. Beberapa fitur populer yang akan dijabarkan disini.

a. *Internet*

Kemampuan akses internet pada Android memberikan banyak keunggulan. Berbagai informasi secara real-time dapat diperoleh dengan mudah dengan internet. Contoh, sebagai pengguna, dapat menggunakan internet untuk melihat jadwal pemutaran film bioskop, cuaca suatu area, jadwal penerbangan dan lainnya. Sebagai *developer*, dapat menggunakan internet untuk akses secara real-time kepada data, *update* data. *Developer* juga dapat menggunakan internet untuk menyimpan berbagai asset untuk

kemudian digunakan suatu aplikasi, seperti dilakukan Pandora dan YouTube. Dengan internet dapat dibangun model aplikasi yang disebut *client-server computing*. Contoh lain, aplikasi peta, mengakses data peta dan GPS dari web server.

b. *Audio dan Video Support*

Sistem operasi Android memungkinkan developer menyertakan audio dan video dalam aplikasi dengan mudah. Berbagai standar format audio dan video didukung.

c. *Contact*

Android memungkinkan akses ke *contacts* yang tersimpan dapat Perangkat Android. Developer dapat menggunakan fitur ini untuk menampilkan *contacts* dalam cara baru yang berbeda. Hal lain yang dapat dilakukan adalah membangun aplikasi yang menggabungkan antara *contacts* dengan GPS, yang memberikan notifikasi kepada pengguna jika pengguna berada di dekat alamat satu *contact* yang ada.

d. *Security*

Android memungkinkan aplikasi untuk melakukan banyak hal. Akan tetapi Android juga menyiapkan mekanisme keamanan berupa *permission* berkaitan dengan beberapa tugas. Contoh: Download *image* dan menyimpannya di SD Card, maka harus disetujui terlebih dahulu *permission* untuk mengakses SD Card.

e. *Google APIs*

Sistem operasi Android memungkinkan dengan tidak terbatas membuat panggilan telepon, mengorganisasi *contacts* atau meng-*install* aplikasi. Developer dapat mengintegrasikan peta (*map*) ke dalam suatu aplikasi dengan menggunakan Maps API yang mengandung Map Widgets. Berbagai fitur dapat ditambahkan dengan Maps API, antara lain: (1) Menampilkan suatu lokasi di peta, (2) Mendapatkan panduan navigasi, (3) Komunikasi data antara aplikasi dengan *clouds*.

1.5. Menjadi Developer Android

Semua perangkat lunak untuk membangun aplikasi Android bisa didapatkan dengan GRATIS. Hal tersebut salah satu keunggulan membangun aplikasi Android.

Menjadi developer Android bukan merupakan pekerjaan yang sulit, yang utama yang perlu ditanyakan terhadap diri adalah:

- 1) Apakah anda ingin membangun Aplikasi Android ?
- 2) Apakah anda suka perangkat lunak pengembang yang gratis ?
- 3) Apakah anda memiliki komputer atau laptop untuk digunakan ?

1.6. Framework Android

Android sepenuhnya open source, sehingga semua developer yang membutuhkan dapat mengakses untuk menggunakan dan memodifikasi kode program Android. Source code android dapat di download melalui <http://source.android.com>.

Android dibangun diatas open source linux kernel 2.6. Kernel linexu tersebut dipilih karena menyediakan fitur utama untuk membangun sistem operasi Android, diantaranya:

- 4) Security Model: Linux kernel menangani keamanan antara aplikasi dan sistem
- 5) Memory Management: Linux kernel menangani manajemen memori untuk developer
- 6) Process Management: Linux kernel mengatur proses dengan baik, mengalokasikan resource untuk proses sesuai dengan kebutuhan
- 7) Network Management: Linux kernel juga menangani jaringan komunikasi
- 8) Driver Model: Perusahaan perangkat keras dapat mengembangkan drivers perangkat mereka secara mandiri untuk berjalan di atas linux



Diatas kernel linux tersebut, framework Android dibangun dengan berbagai fitur. Fitur-fitur tersebut diadopsi dari berbagai project opensource. Beberapa fitur Android framework diantaranya:

- 9) Android Run Time: Mengandung inti library java dan dalvik virtual machine
- 10) Open GL (Graphics Library): Merupakan API (Application Program Interface) yang digunakan untuk menghasil graphics 2D dan 3D, bersifat cross-language dan cross-platform
- 11) Webkit: merupakan mesin web browser opensource yang menyediakan fungsionalitas untuk menampilkan web konten
- 12) SQLite: merupakan opensource mesin database yang di rancang untuk sistem embedded
- 13) Media frameworks: library yang digunakan untuk menjalankan dan merekam audio serta video
- 14) Secure Socket Layer (SSL): library ini bertanggung jawab terhadap keamanan internet



Untuk pemanfaatan fitur-fitur dalam Android framework diatas, disediakan berbagai library dalam Application Framework yang dapat digunakan oleh developer, diantaranya:

- 15) Activity Manager: Mengelola siklus hidup activity
- 16) Telephony Manager: Menyediakan akses ke telephone service
- 17) View System: Menangani view dan layout yang membangun User Interface (UI)
- 18) Location Manager: menemukan lokasi secara geografi



1.7. Open Handled Alliance (OHA)

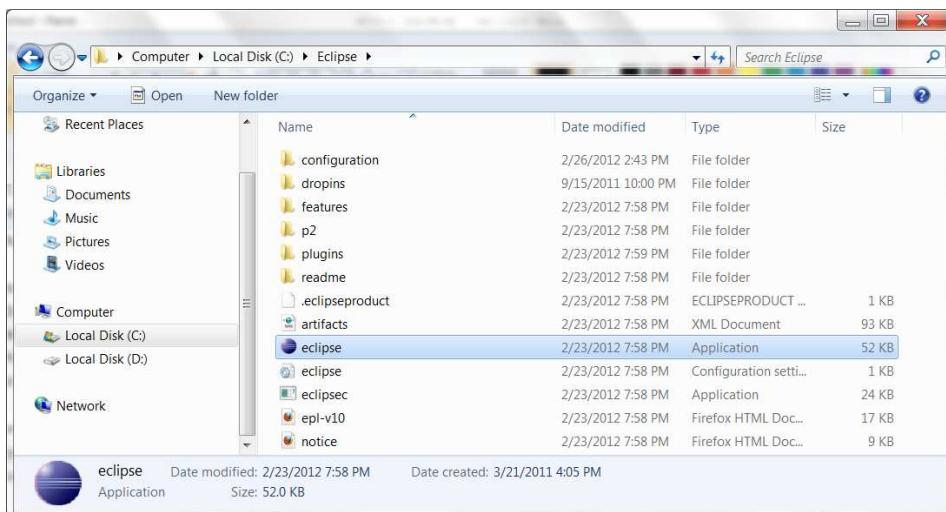
Open Handled Alliance (OHA) berdiri November 2007 dengan 34 anggota yang dimotori oleh google. Saat ini OHA telah memiliki 71 anggota, diantaranya: T-Mobile, Sprint, LG, Motorola, HTC, NVidia, and Texas Instruments

1.8. Perangkat Lunak Pengembangan

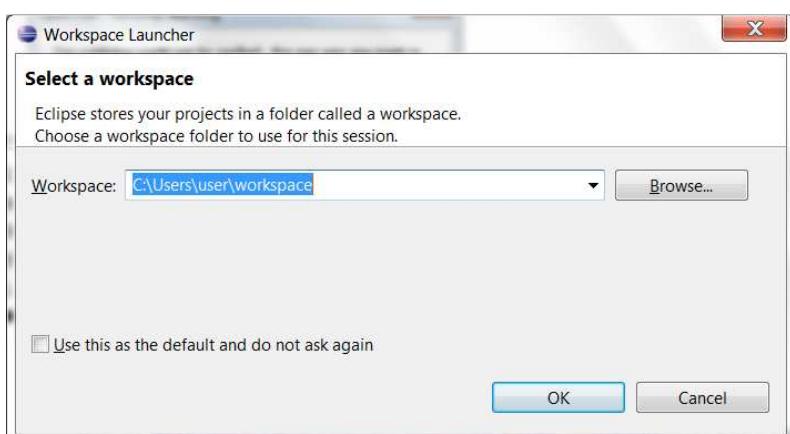
- Java JDK → <http://java.sun.com/javase/downloads/index.jsp>
Android SDK → <http://developer.android.com/sdk/index.html>
Eclipse IDE → www.eclipse.org/downloads
Android ADT → <https://dl-ssl.google.com/android/eclipse/>

1.9. Menginstall Eclipse

1. Double-Click pada shortcut Eclipse.exe untuk menjalankan Eclipse



2. Jika dibutuhkan konfigurasi workspace



Jika anda bekerja dengan beberapa *project* dianjurkan untuk menggunakan *workspace* yang berbeda untuk masing-masing *project*.

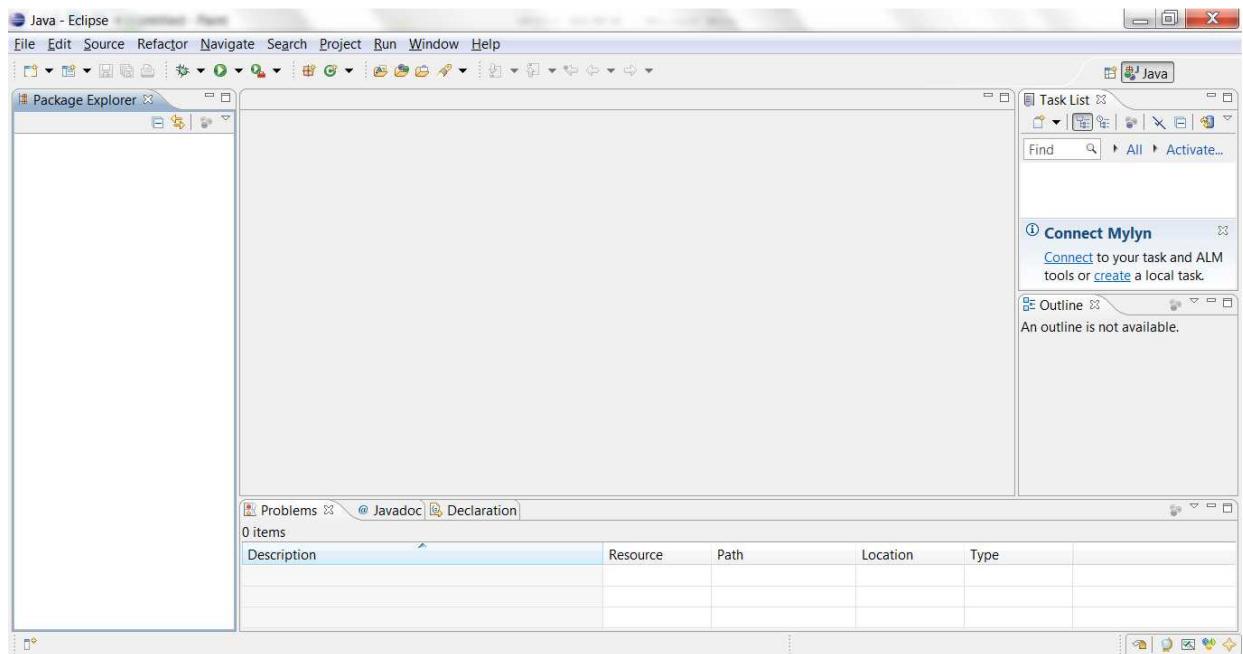
3. Tampilan awal Eclipse



4. Click pada icon untuk menuju workbench.



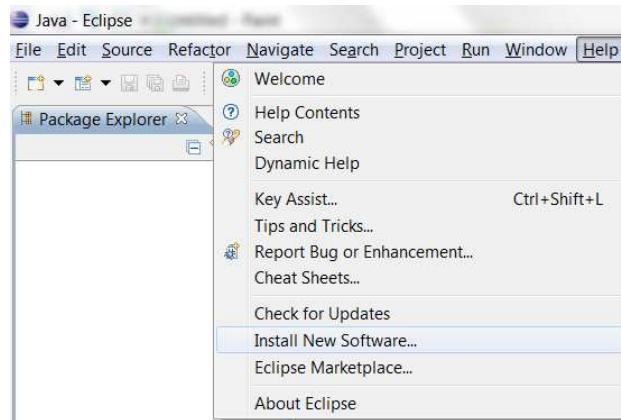
5. Workbench



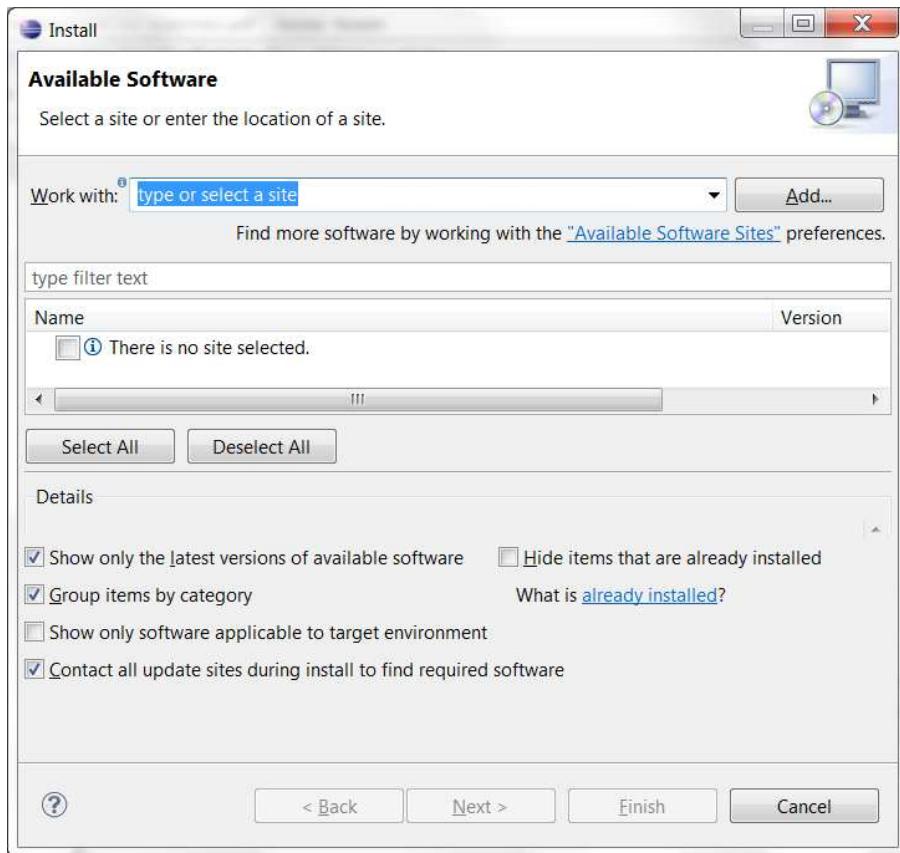
1.10. Mengkonfigurasi Eclipse dengan ADT

Untuk dapat menggunakan Eclipse dalam membangun Aplikasi Android. Terlebih dahulu harus dilakukan konfigurasi *Android Development Tools* (ADT) dalam Eclipse.

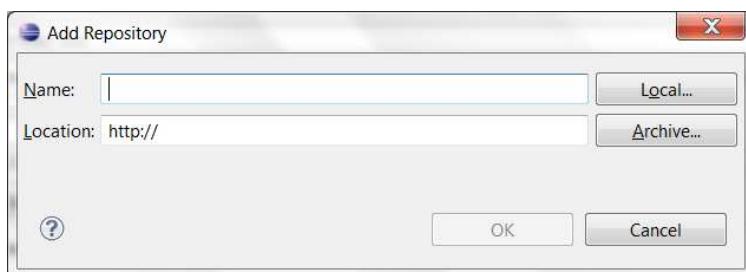
1. Jalankan Eclipse (jika belum berjalan)
2. Pilih pada menubar → *Help* → *Install New Software*



3. Muncul Install window pops-up, yang mengijinkan untuk menginstall plug-in baru ke dalam Eclipse.

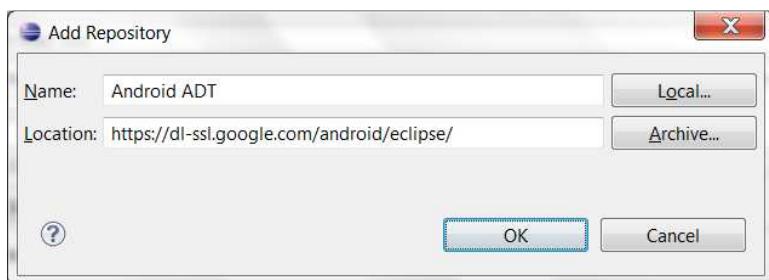


4. Klik Button Add, untuk menambahkan

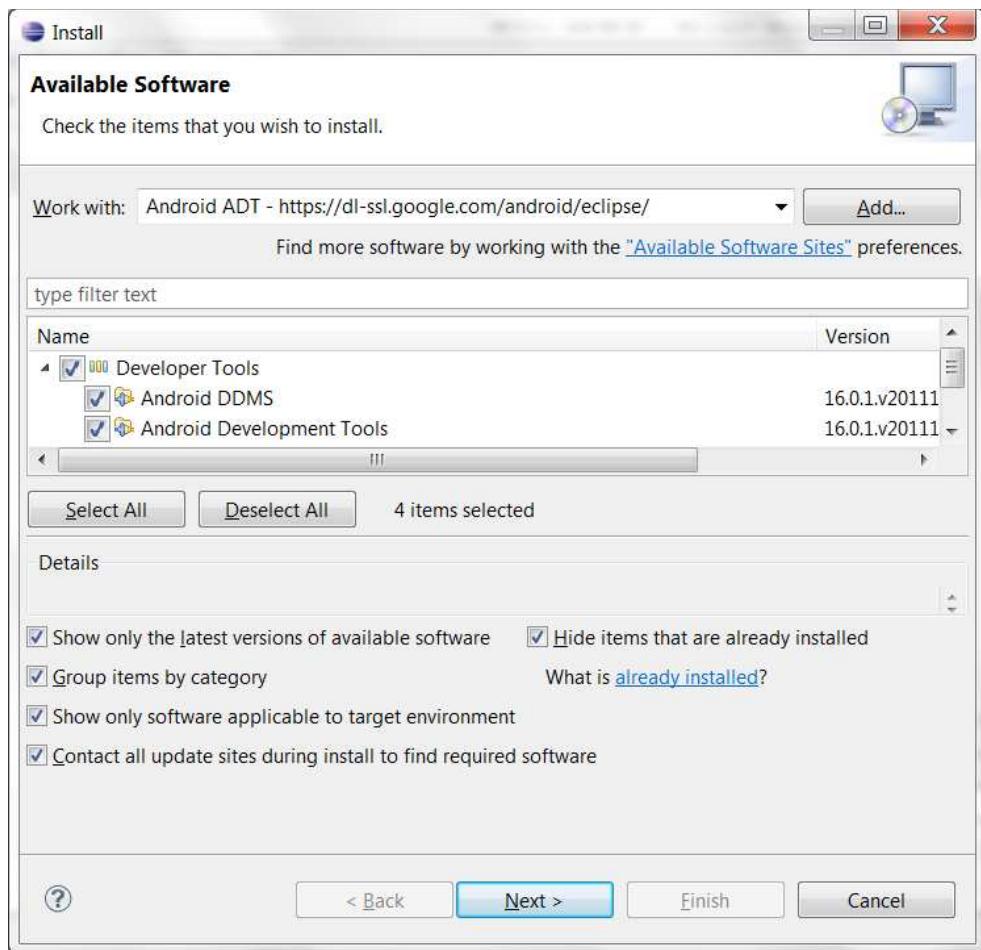


5. Tuliskan nama pada Name Field, misal: **Android ADT**

6. Pada Location Field, tuliskan: <https://dl-ssl.google.com/android/eclipse/>

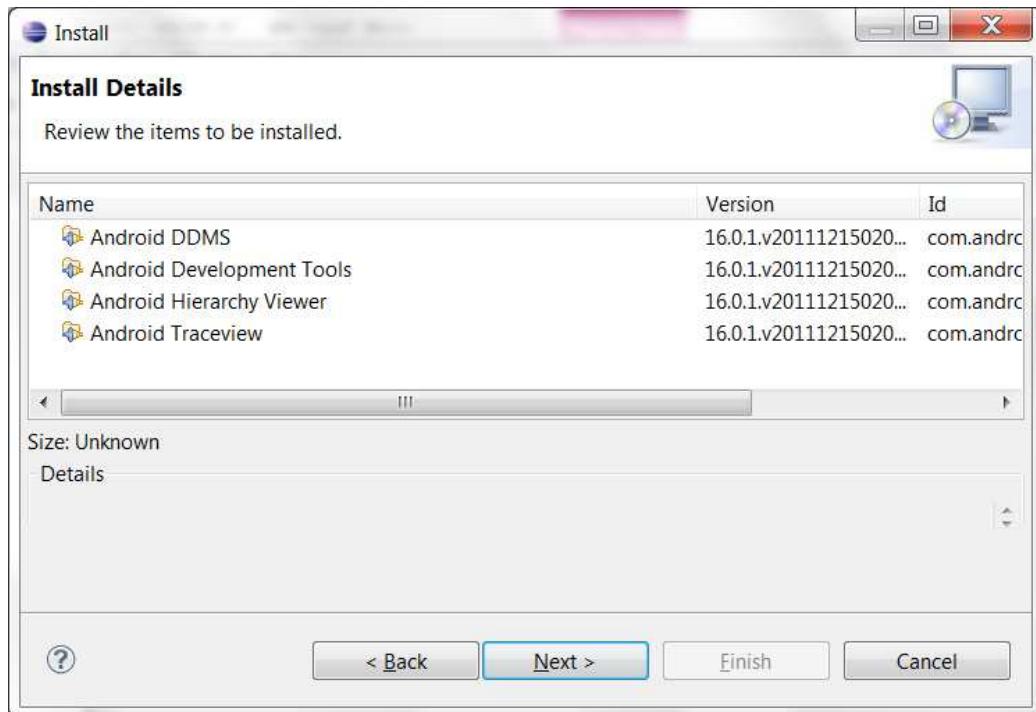


7. Tampil:

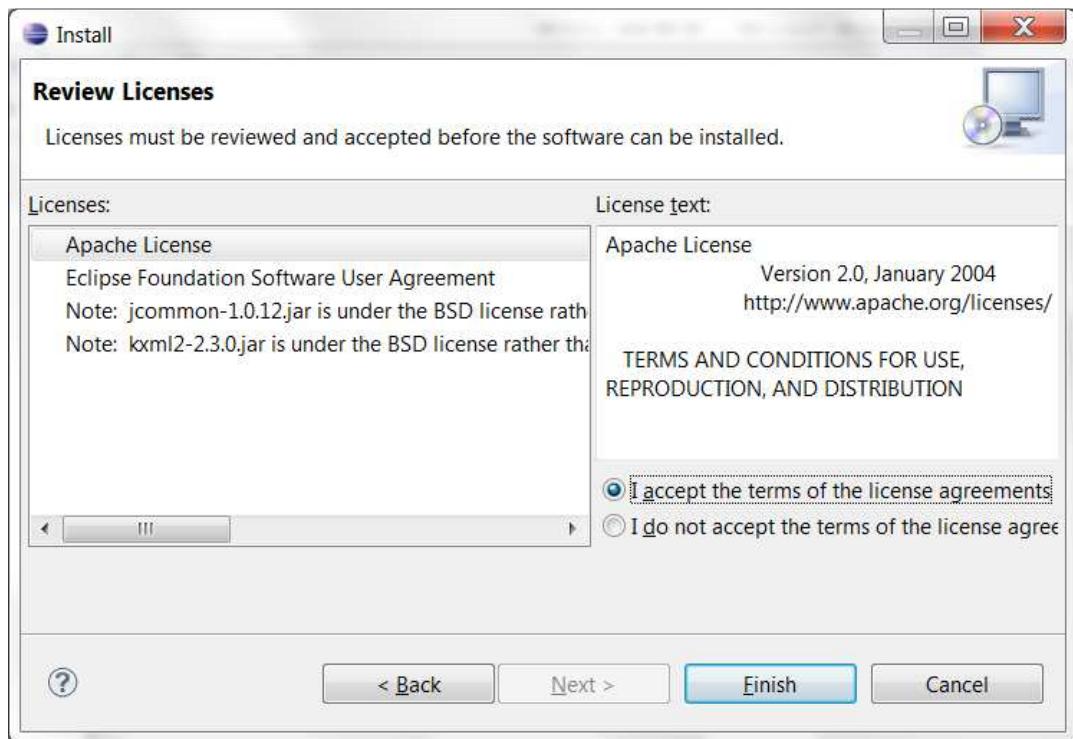


8. Select All, klik next> untuk install

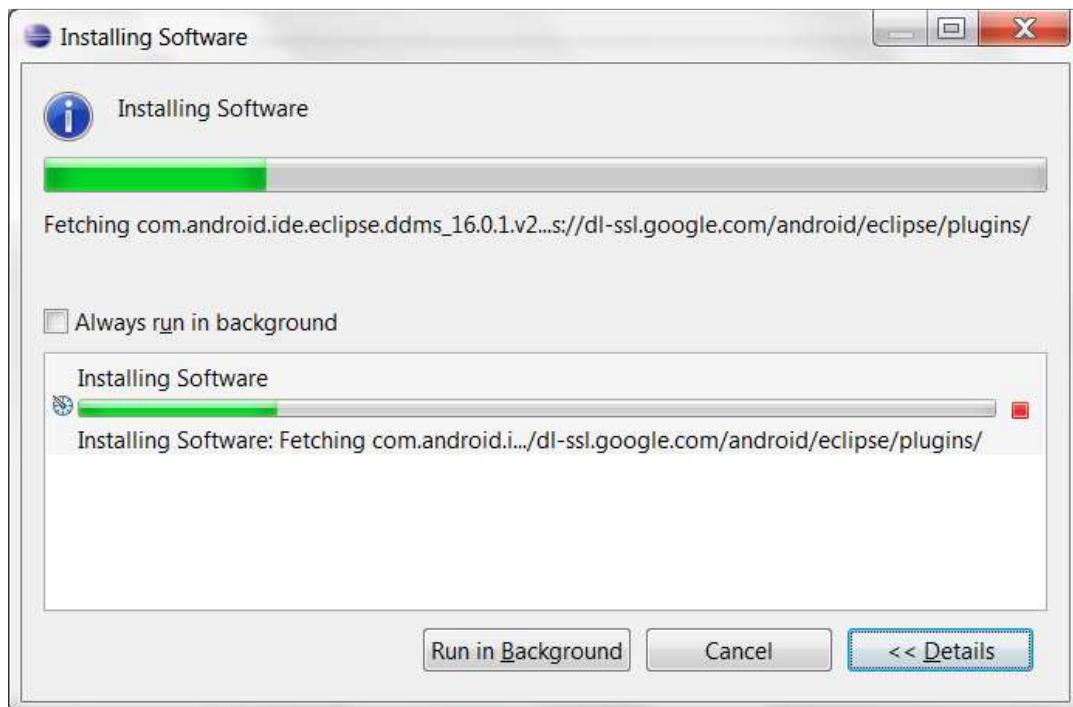
9. Review, klik next>



10. License Agreement, Klik Finish

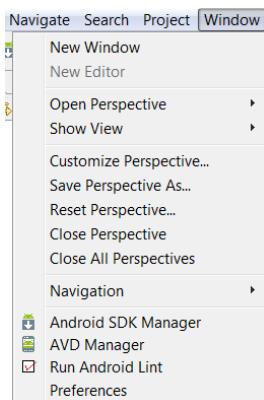


11. Install dalam akan berjalan, plugin ADT akan terinstall

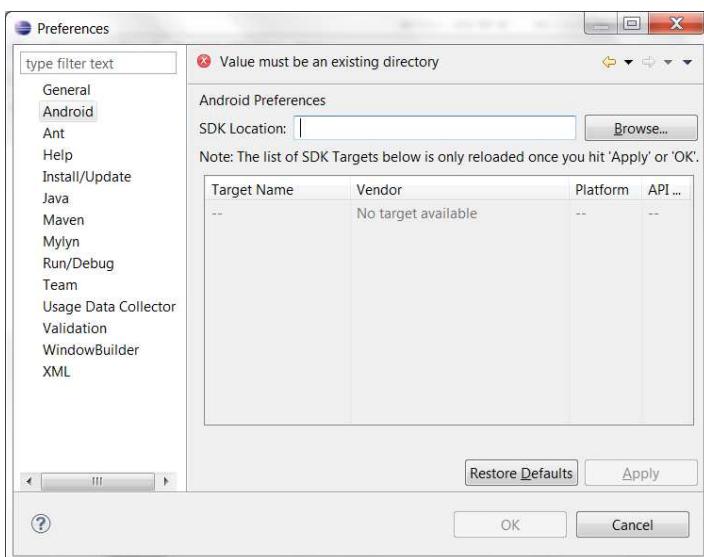


1.11. Mengkonfigurasi lokasi SDK

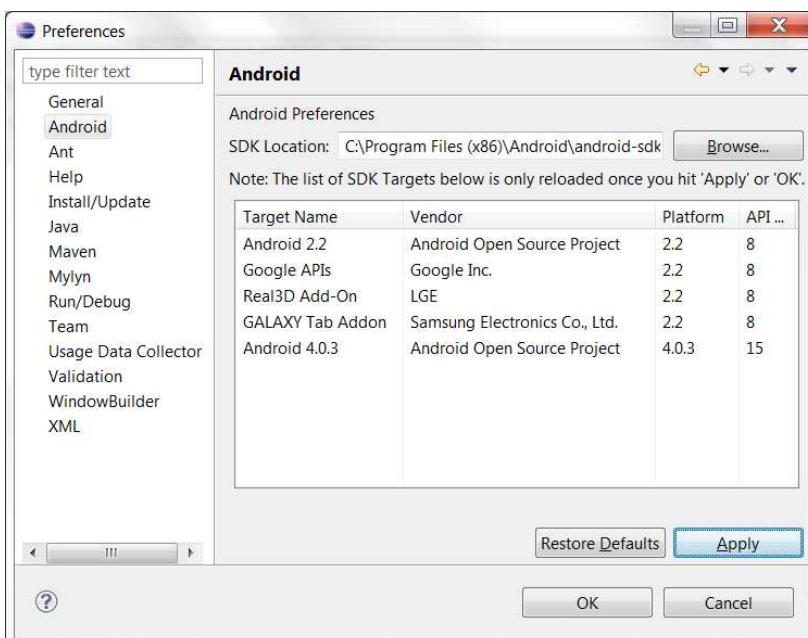
1. Pada menubar, pilih window →preferences



2. Pilih Android pada tab di kiri



3. Atur lokasi SDK, Kemudian OK



1.12. Pustaka

FelkerD.; Dobbs J, 2011, *Android Application Development For Dummies*, Wiley Publishing Inc., USA

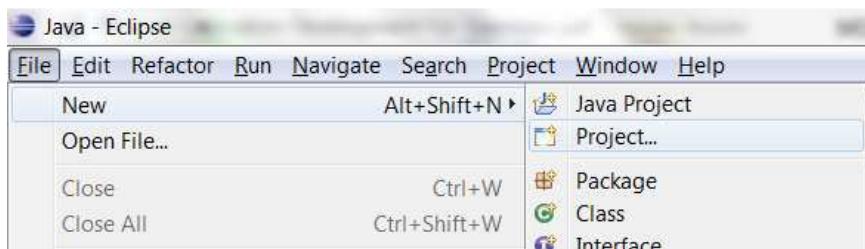
MATERI 2

MEMBANGUN APLIKASI ANDROID

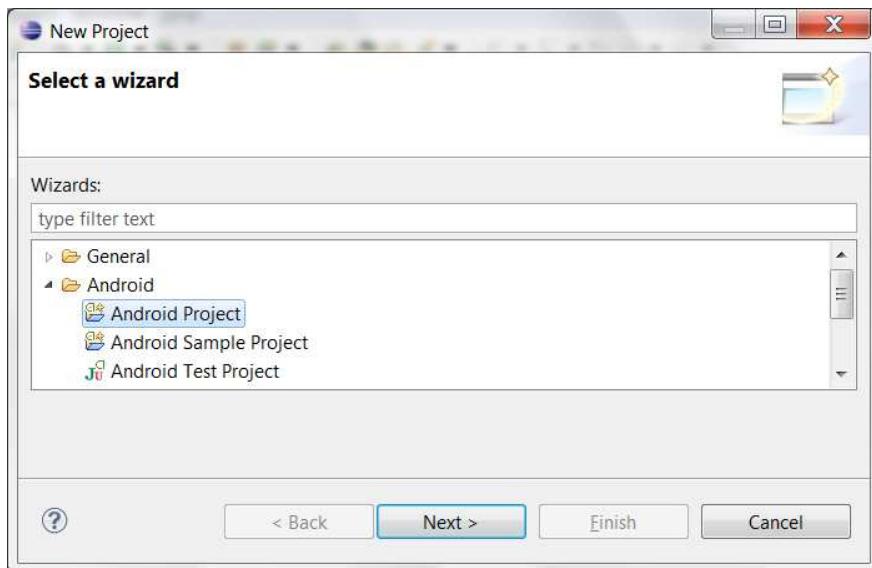
- Membangun Project Baru di Eclipse
 - Mengkonfigurasi Emulator
 - Konfigurasi dan Menjalankan Aplikasi
 - Memahami Anatomi dari Project
-

2.1. Membangun Project Baru di Eclipse

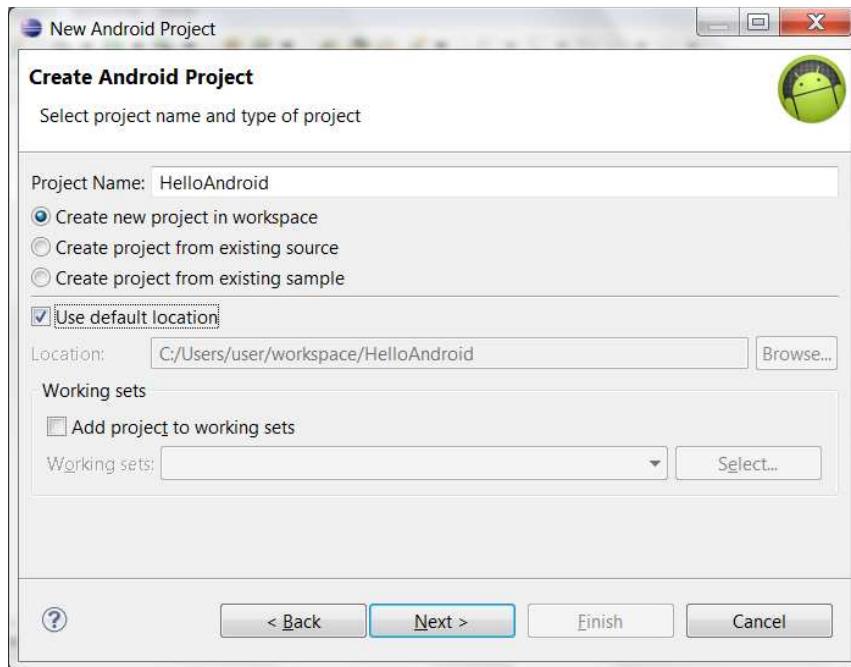
1. Dalam Eclipse, Pilih File → New → Project



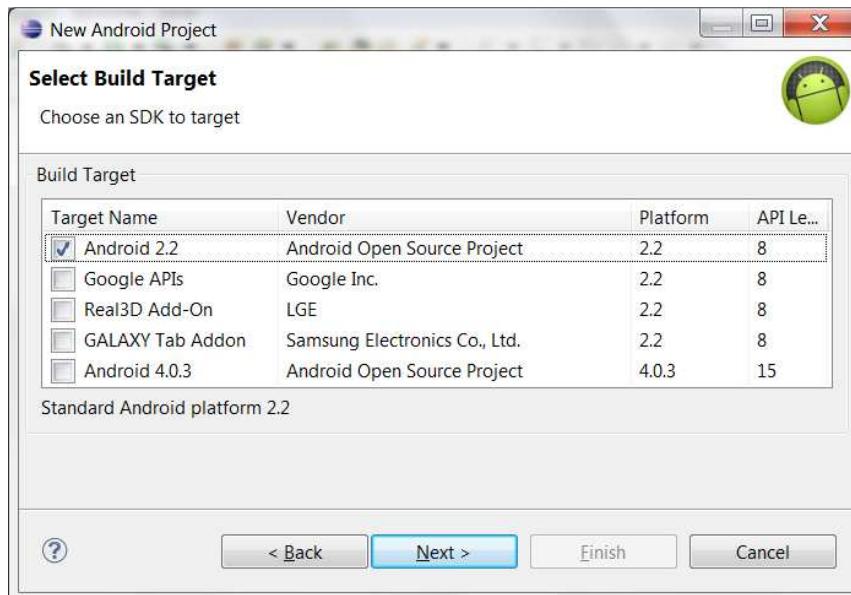
2. Pilih Android → Android Project → Next



3. Pada Project Name, tulis: **HelloAndroid**



4. Untuk Build Target → Pilih Android 2.2



5. Isikan untuk Application Info:

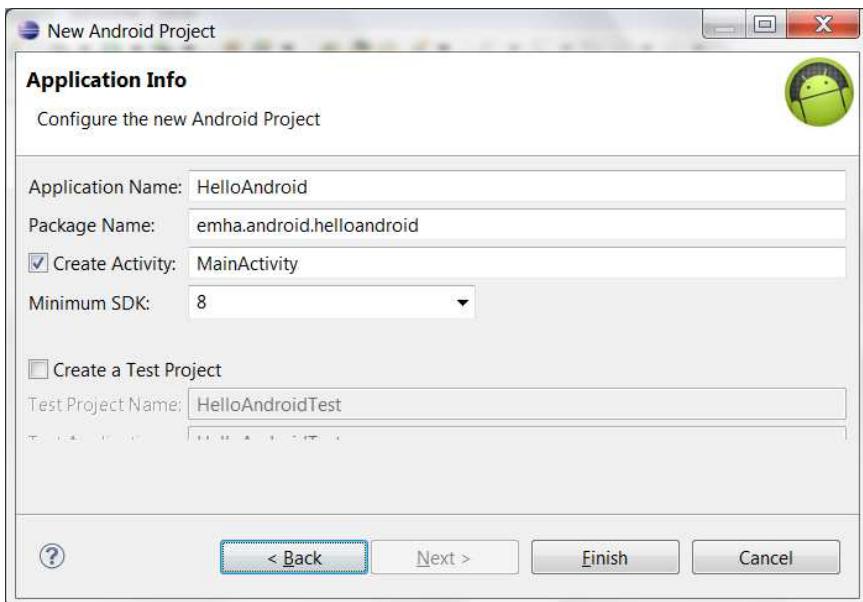
Application Name : **HelloAndroid**

Package Name : **emha.android.helloandroid**

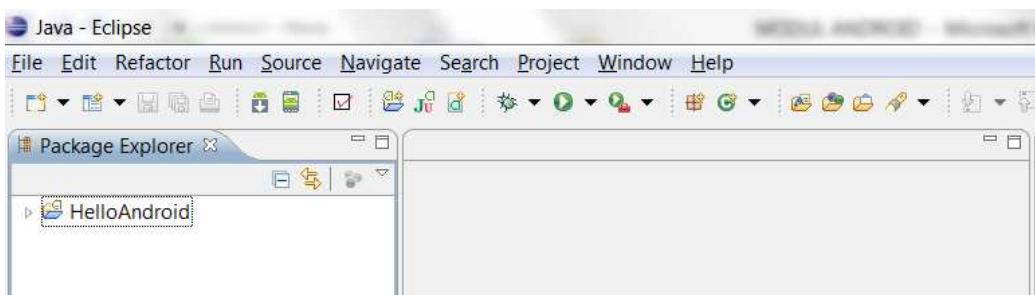
Create Activity : **MainActivity**

Minimum SDK : **8**

Klik Finish



6. Didapatkan tampilan pada Package Explorer sbb:



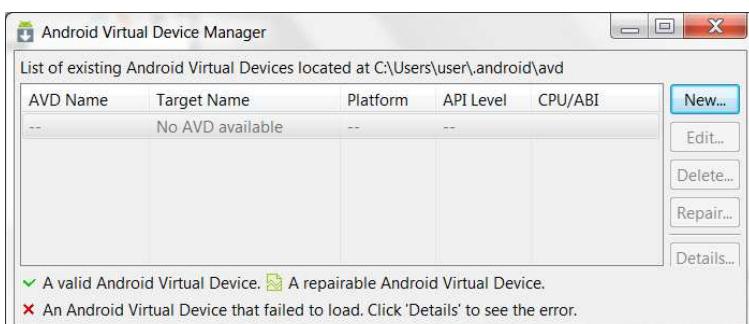
2.2. Mengkonfigurasi Emulator

Untuk melihat apakah aplikasi yang dibangun pada kondisi berjalan baik atau tidak, dapat dicoba dijalankan dengan emulator. Developer harus tahu bagaimana men-set up emulator dengan beberapa konfigurasi yang berbeda. Pertama harus membuat Android Virtual Device (AVD) yang disebut juga emulator. AVD merupakan Perangkat Virtual Android yang memiliki fitur seperti Perangkat Android sesungguhnya.

1. Buka AVD Manager, pada toolbar klik ikon (Opens Android Virtual Device Manager)



2. Muncul kotak dialog sbb:

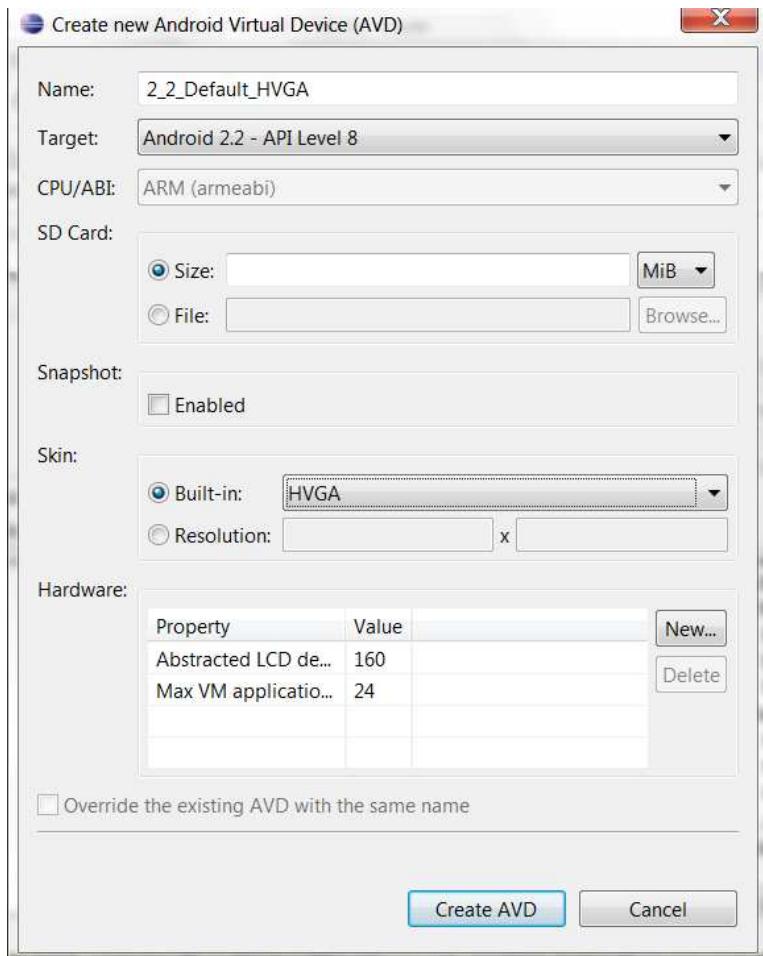


3. Klik New

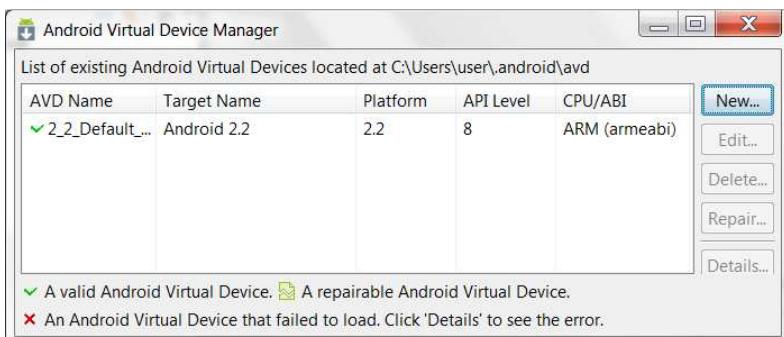
Kemudian isikan :

Name : 2_2_Default_HVGA
Target : Android 2.2 – API Level 8
SD Card : <sementara kosongkan>
Skin : HVGA
Hardware : <sementara abaikan>

Klik Button → Create AVG



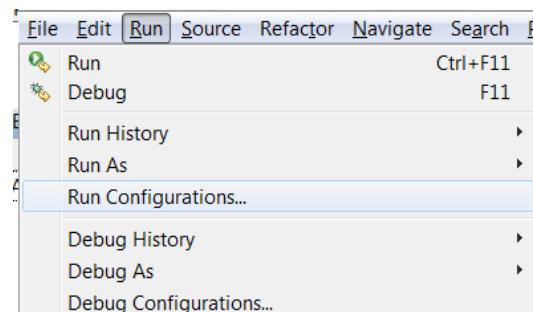
Didapatkan hasil sebagai berikut :



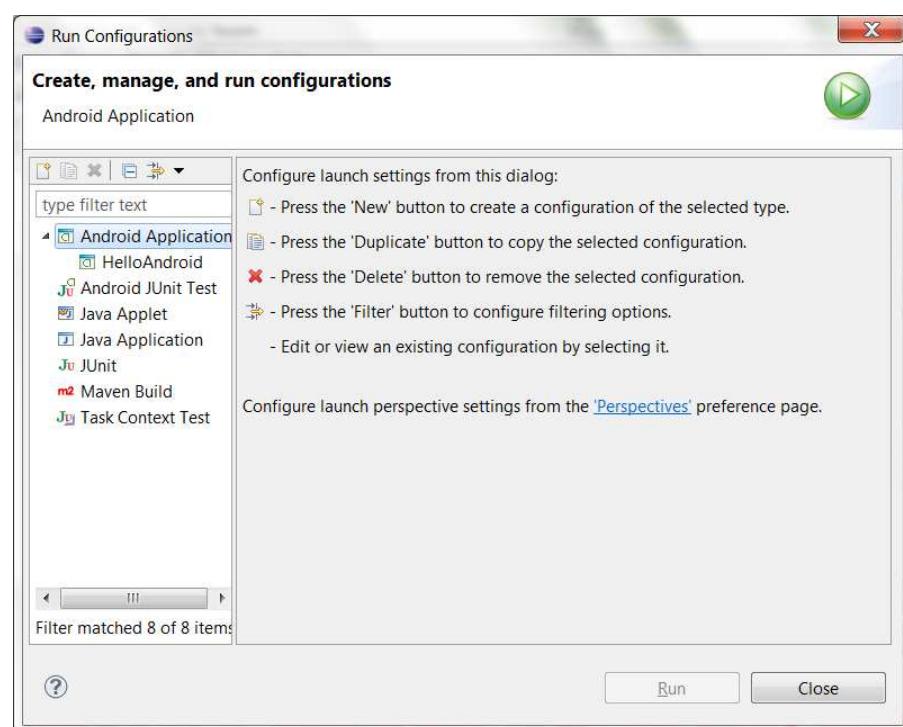
2.3. Menjalankan Project Aplikasi Android dengan Emulator

1. Sebelum project aplikasi dijalankan, atur terlebih dahulu **run configuration**

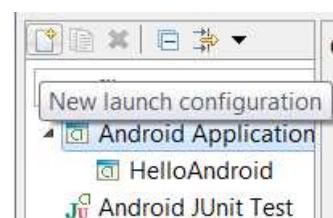
Pada MenuBar → Run → Run Configuration



2. Pilih Android Application



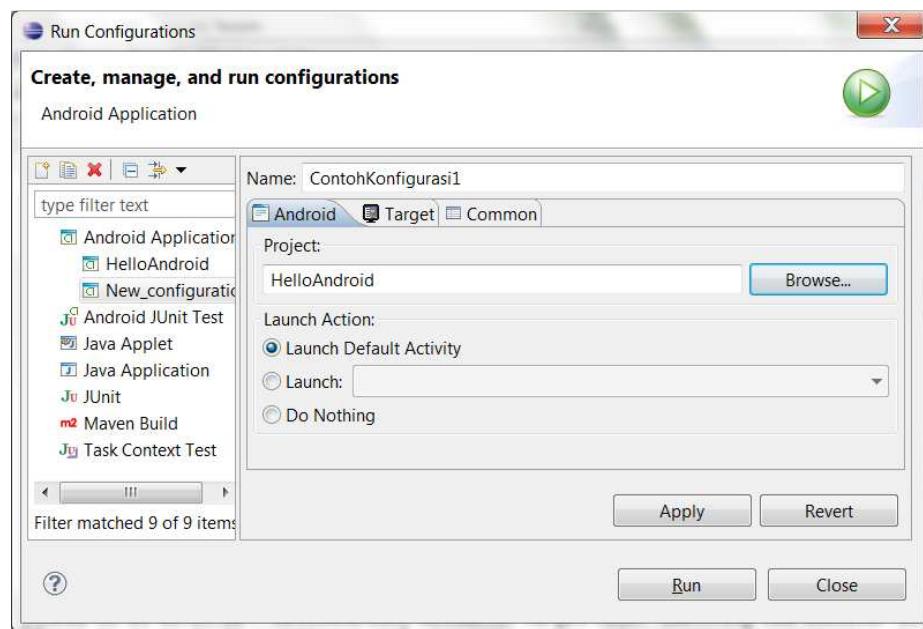
3. Kemudian klik ikon new



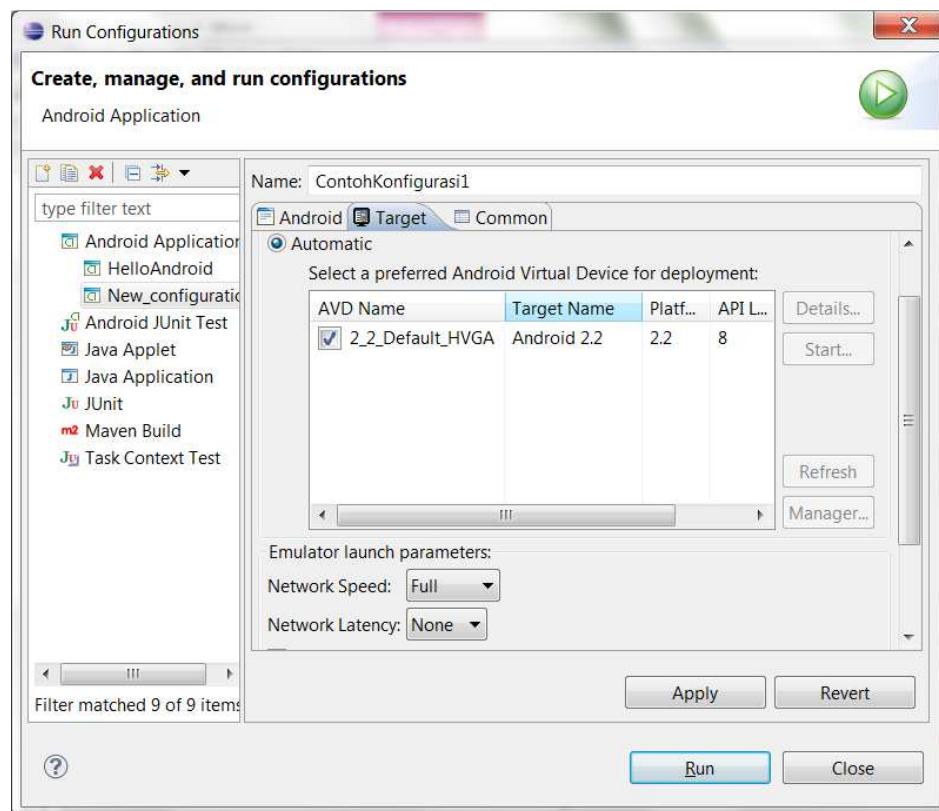
4. Masukan Nama Konfigurasi dan Pilih Project yang akan dijalankan dengan konfigurasi ini

Name : ContohKonfigurasi1

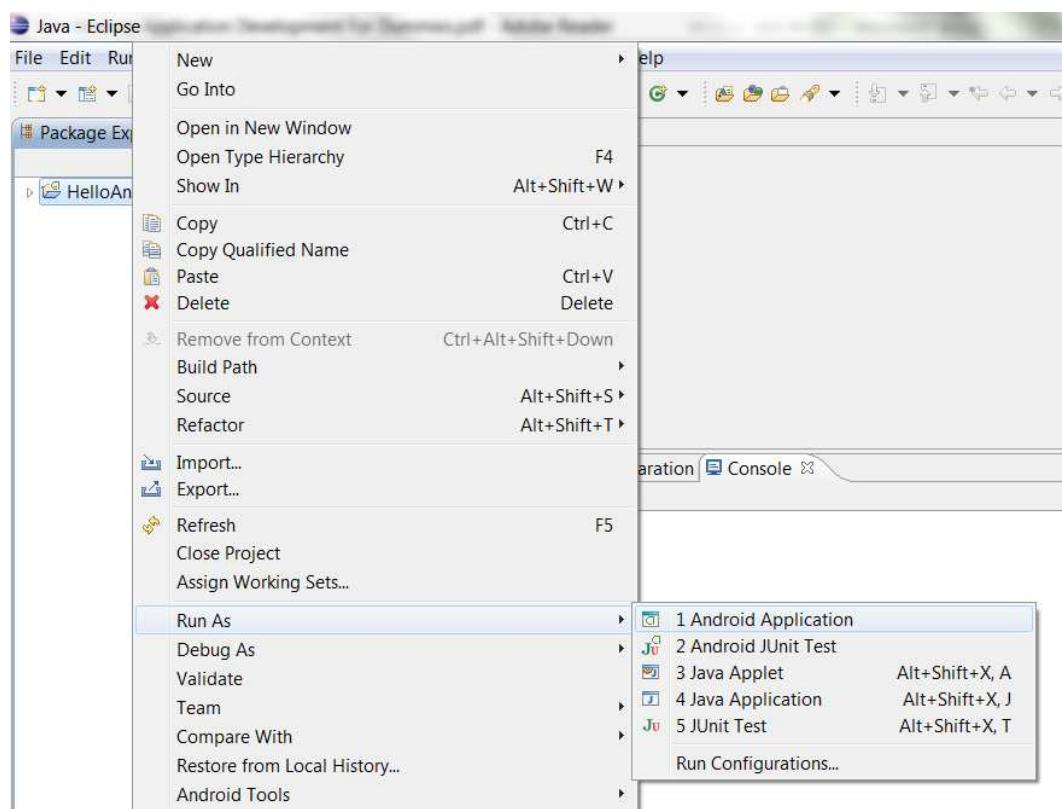
Project : HelloAndroid



Pada Tab Target : Pilih Automatic kemudian AVD → 2_2_Default_HVGA

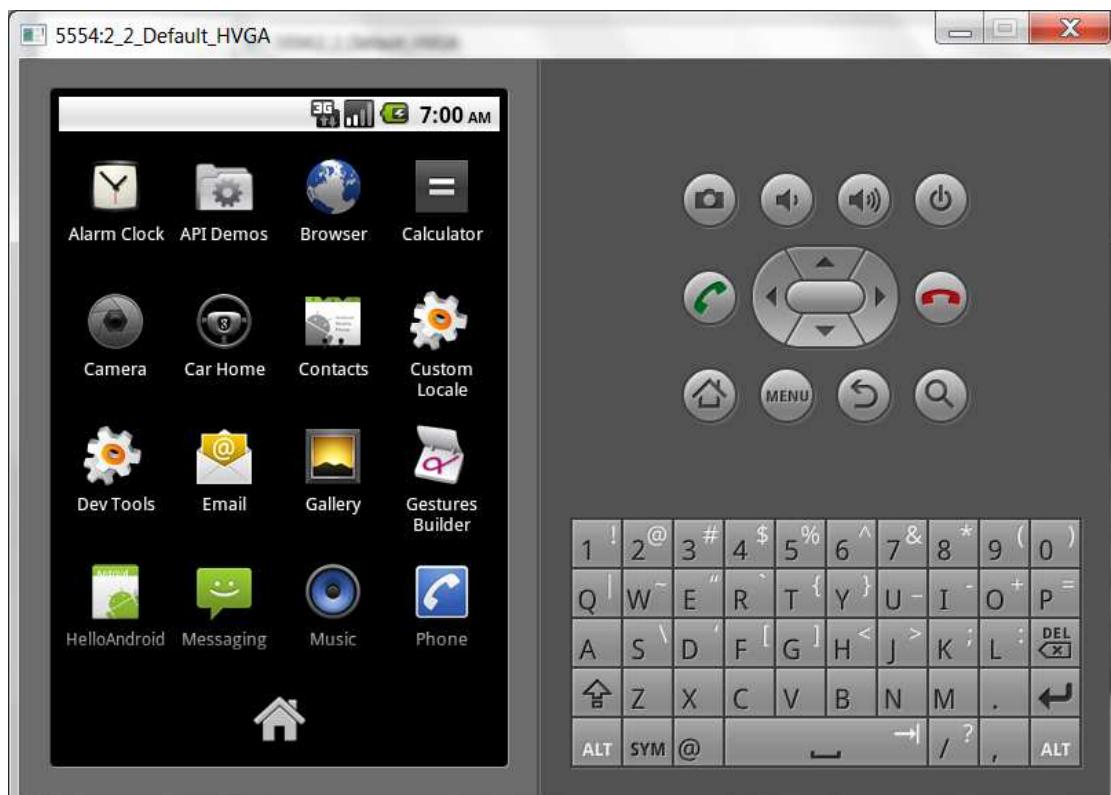


5. Jalankan



6. Tampilan AVD

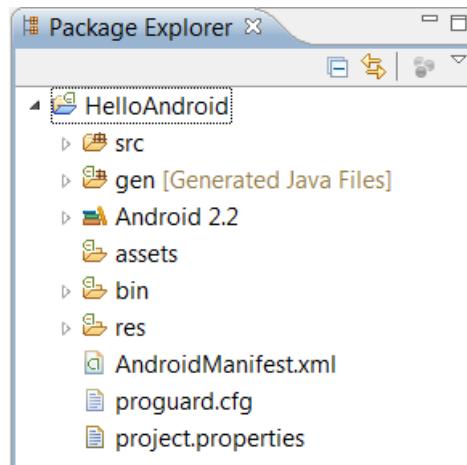






2.4. Struktur Project Android

Pada Project Explorer terlihat detail struktur dari suatu Project Android



Terdapat beberapa folder dibawah Nama Project, yaitu:

19) src

merupakan folder untuk source

20) gen

merupakan folder misterius. Folder ini berikan file yang di generate oleh ADT

21) Android Version (misal: Android 2.2)

22) assets

merupakan folder untuk menempatkan berbagai file asset (data) yang dimiliki project (contoh: file data dari SQLite). Data di folder ini dapat diakses melalui AssetManager atau metode getAssets().

23) res

merupakan folder untuk menempatkan berbagai resource yang digunakan oleh project, misal: file xml, icon atau picture.

Selain folder-folder diatas, terdapat beberapa folder lain misal folder: *bin*, *libs*, dan *referenced libraries*.

Folder *bin* tidak tampak dikarenakan di *hidden*. Sementara folder *libs* dan *referenced libraries* tidak akan muncul sampai developer menambahkan *third-party library* dan direferensi dalam project.

File *AndroidManifest.xml* membantu developer mengidentifikasi komponen yang di *build* dan dijalankan oleh aplikasi.

Project.properties membantu developer mengidentifikasi *default properties* dari project android.

2.5. Pustaka

FelkerD.; Dobbs J, 2011, *Android Application Development For Dummies*, Wiley Publishing Inc., USA

MATERI 3

MEMBANGUN GRAPHICAL USER INTERFACE

- Membangun Aplikasi Pengalih Mode *Silent*
 - Merancang *Layout*
 - Membangun Antarmuka Pengguna
 - Menambahkan *Image* dan *Button*
-

1. Buatlah sebuah project android baru dengan aturan sebagai berikut:

Project Name : PengalihModeSilent

Application Name : PengalihModeSilent

Contents : default

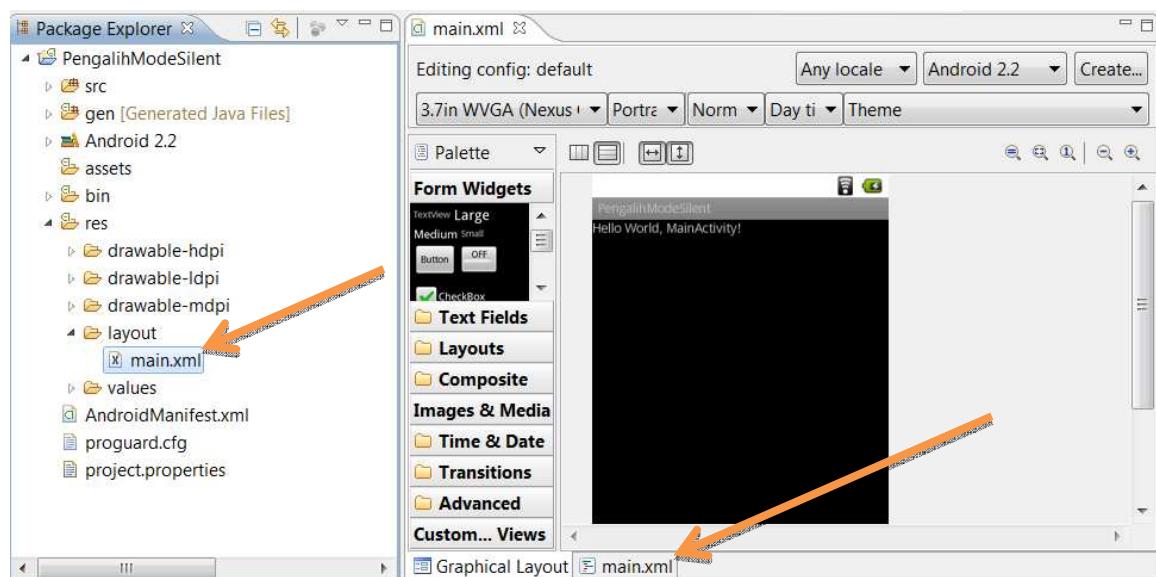
Build Target : Android 2.2

Package Name : emha.android.pengalihmodesilent

Create Activity : MainActivity

Min SDK Version : 8

2. Buka file main.xml di folder res → layout



Default-nya yang akan muncul adalah Visual Designer dari main.xml, klik pada tab main.xml dibagian bawah.

```
*main.xml
<?xml version="1.0" encoding="utf-8"?>

```

Default Deklarasi XML

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

Baris pertama menunjukkan deklarasi file xml. Memberikan informasi tipe file tersebut terhadap editor.

Tipe Layout

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >
```

Baris selanjutnya diatas menunjukkan tipe Layout yang digunakan, dalam hal ini adalah LinearLayout (akan dibahas lebih lanjut nanti).

View

```
<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="@string/hello" />
```

Baris selanjut menunjukkan blok untuk view, yaitu blok dasar pembangun *user interface*.

Dalam hal ini adalah sebuah TextView.

3. Dari main.xml, hapus bagian view (TextView) yang ada. Sehingga main.xml secara keseluruhan akan menjadi sbb:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >

</LinearLayout>
```

4. Tambahkan (tarik) image yang akan digunakan kedalam folder res → drawable-mdpi



phone_on.png → untuk mode telepon regular

phone_silent.png → untuk mode telepon silent

5. Tambahkan image ke dalam Layout

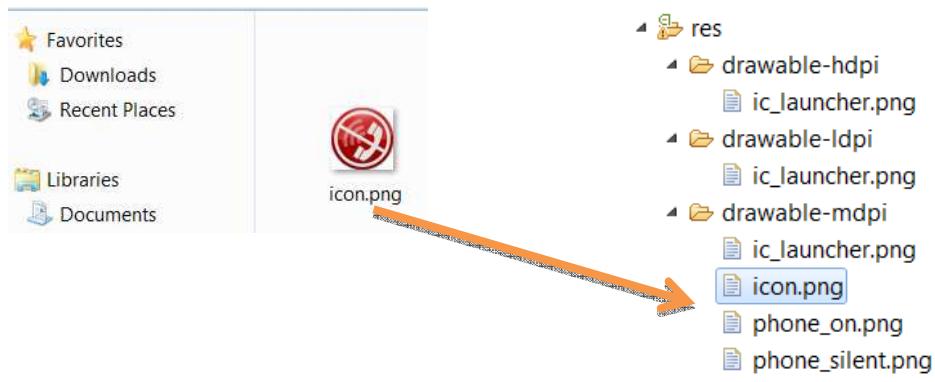
Dilakukan dengan mengetikkan kode tambahan di main.xml sehingga menjadi sbb:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >

    <ImageView
        android:id="@+id/phone_icon"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:src="@drawable/phone_on" />

</LinearLayout>
```

6. Tambahkan (tarik) icon untuk aplikasi ke project, res → drawable-mdpi



7. Tambahkan button ke dalam main.xml, menjadi sbb:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >

    <ImageView
        android:id="@+id/phone_icon"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:src="@drawable/phone_on" />

    <Button
        android:id="@+id/toggleButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:text="Pengalih Mode Silent" />

</LinearLayout>
```



8. Coba tampilkan Visual Designer



9. Atur Run Configuration

10. Jalankan Aplikasi PengalihModeSilent



11. Buka file MainActivity.java di folder src → emha.android.pengalihmodesilent

The screenshot shows the Eclipse IDE interface. On the left, the Package Explorer view displays the project structure under 'PengalihModeSilent'. It includes a 'src' folder containing 'emha.android.pengalihmodesilent' which has 'MainActivity.java'. Other visible folders are 'gen', 'Android 2.2', 'assets', 'bin', and 'res'. The 'res' folder contains 'drawable-hdpi' and 'ic_launcher.png'. On the right, the code editor shows the Java code for 'MainActivity.java':

```
package emha.android.pengalihmodesilent;

import android.app.Activity;

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

12. Kita akan mencoba membuat event handler pertama, yang menangani jika button di klik

Tambahkan kode program sbb :

The screenshot shows the Eclipse IDE code editor with the 'MainActivity.java' file open. The code is identical to the one in the previous screenshot, but there is a red arrow pointing to the closing brace of the 'onClick' method. This indicates a syntax error.

```
package emha.android.pengalihmodesilent;

import android.app.Activity;

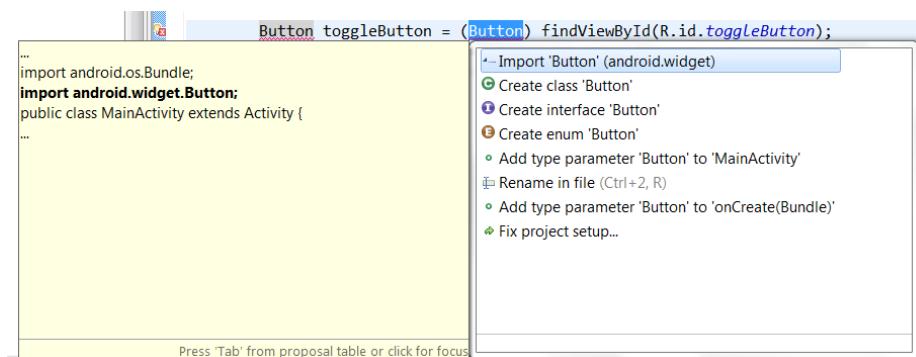
public class MainActivity extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

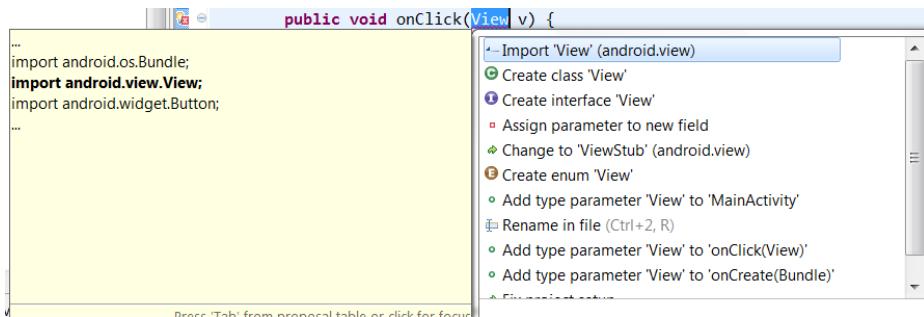
        Button toggleButton = (Button) findViewById(R.id.toggleButton);
        toggleButton.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
            });
    }
}
```

13. Tanda cross merah di kiri pinggir menunjukan terdapat error pada baris kode bersangkutan.

14. Hal ini terjadi dikarenakan object Button dan View belum dikenali.

Pada tanda cross merah, klik kemudian klik import untuk menambahkan library





15. Pada bagian atas kode, akan terdapat beberapa tambahan import sbb:

```
import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class MainActivity extends Activity {
```

16. Untuk selanjutnya agar manajemen kode program menjadi lebih baik, maka kita ubah, dengan menempatkan kode program dengan fungsi tertentu dalam satu paket method

```
package emha.android.pengalihmodesilent;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

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

        setButtonClickListener();
    }

    private void setButtonClickListener() {
        Button toggleButton = (Button) findViewById(R.id.toggleButton);
        toggleButton.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                // TODO Auto-generated method stub
            }
        });
    }
}
```

(orange arrows point to the 'setButtonClickListener()' method definition and its call in the onCreate() method)

17. Untuk selanjutnya kita akan coba bekerja dengan audio service dan pengaturan mode.

Lengkapi kode program menjadi sbb:

```
package emha.android.pengalihmodesilent;

import android.app.Activity;

public class MainActivity extends Activity {

    private AudioManager mAudioManager;
    private boolean statusModeSilent;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        setButtonClickListener();
        mAudioManager = (AudioManager) getSystemService(AUDIO_SERVICE);
        checkIfPhoneIsSilent();
    }

    private void checkIfPhoneIsSilent() {
        int ringerMode = mAudioManager.getRingerMode();
        if (ringerMode == AudioManager.RINGER_MODE_SILENT) {
            statusModeSilent = true;
        } else {
            statusModeSilent = false;
        }
    }

    private void setButtonClickListener() {
        Button toggleButton = (Button) findViewById(R.id.toggleButton);
        toggleButton.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                if (statusModeSilent) {
                    // Ubah kembali ke mode normal
                    mAudioManager.setRingerMode(AudioManager.RINGER_MODE_NORMAL);
                    statusModeSilent = false;
                } else {
                    // Ubah ke mode silent
                    mAudioManager.setRingerMode(AudioManager.RINGER_MODE_SILENT);
                    statusModeSilent = true;
                }
                // Now toggle the UI again
                pindahUI();
            }
        });
    }
}
```

```

private void pindahUI() {
    ImageView imageView = (ImageView) findViewById(R.id.phone_icon);
    Drawable imgTampil;
    if (statusModeSilent) {
        imgTampil = getResources().getDrawable(R.drawable.phone_silent);
    } else {
        imgTampil = getResources().getDrawable(R.drawable.phone_on);
    }
    imageView.setImageDrawable(imgTampil);
}

protected void onResume() {
    super.onResume();
    checkIfPhoneIsSilent();
    pindahUI();
}
}

```

18. Jalankan aplikasi



Klik Button



Pustaka

FelkerD.; Dobbs J, 2011, *Android Application Development For Dummies*, Wiley Publishing Inc., USA

MATERI 4

MEMBANGUN GRAPHICAL USER INTERFACE

- **Bekerja dengan TextView**
 - **Bekerja dengan Button**
 - **Bekerja dengan Image**
 - **Bekerja dengan CheckBox**
 - **Bekerja dengan RadioButton**
-

1. Buatlah sebuah project android baru dengan aturan sebagai berikut:

Project Name : Materi4_NIMANDA

Application Name : Materi4_NIMANDA

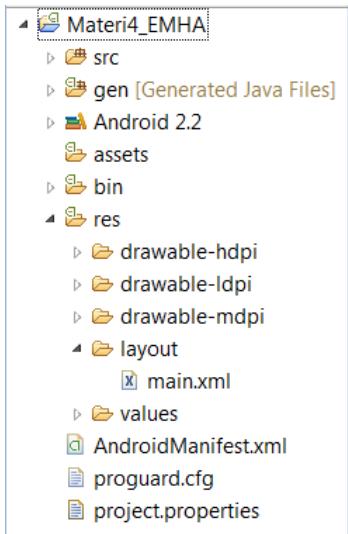
Contents : default

Build Target : Android 2.2

Package Name : emha.android.materi4

Create Activity : MainActivity

Min SDK Version : 8



2. Buka main.xml di folder layout

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >

    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="@string/hello" />

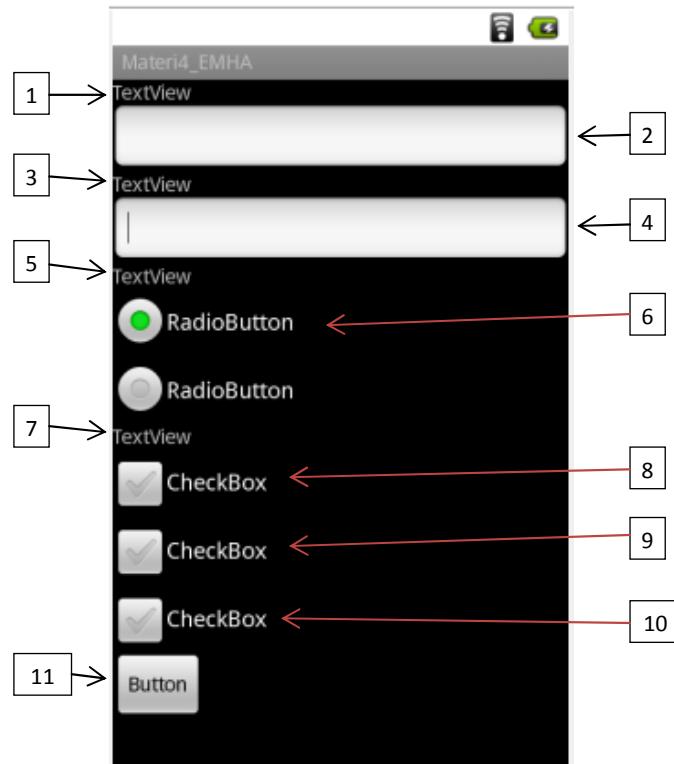
</LinearLayout>
```

3. Hapus TextView yang sudah ada sehingga main.xml menjadi sbb:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >

</LinearLayout>
```

4. Melalui Graphical Layout, tambahkan view-view sebagai berikut:



5. Atur property semua view sbb:

No	Tipe View	Properti	Nilai
1	TextView	id	@+id/textNama
		text	Nama Lengkap
		typeFace	monospace
		textSize	24dp
		textStyle	Bold
2	EditText (Plain Text)	id	@+id/editNama
		typeFace	monospace
		textSize	24dp
3	TextView	id	@+id/textEmail
		text	Email
		typeFace	monospace

		textSize	24dp
		textStyle	Bold
4	EditText (E-Mail)	id	@+id/editEmail
		typeFace	Monospace
		textSize	24dp
5	TextView	id	@+id/textJenisKelamin
		text	Jenis Kelamin
		typeFace	monospace
		textSize	24dp
		textStyle	Bold
6	RadioGroup		
	a. RadioButton	id	radioLaki
		text	Laki-Laki
	b. RadioButton	Id	radioPerempuan
		text	Perempuan
7	TextView	id	textHobi
		text	Hobi
		typeFace	monospace
		textSize	24dp
		textStyle	Bold
8	CheckBox	Id	checkBaca
		text	Membaca
9	CheckBox	id	checkMemancing
		text	Memancing
10	CheckBox	id	checkBersepeda
		text	Bersepeda
11	Button	id	btnOK
		text	OK

6. menu.xml akan berubah menjadi sbb:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical">

    <TextView
        android:id="@+id/textNama"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Nama Lengkap">
```

```
        android:typeface="monospace"
        android:textSize="24dp"
        android:textStyle="bold"/> >

<EditText
    android:id="@+id/editNama"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textSize="24dp"
    android:typeface="monospace"/> >

<TextView
    android:id="@+id/textEmail"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Email"
    android:typeface="monospace"
    android:textSize="24dp"
    android:textStyle="bold"/> >

<EditText
    android:id="@+id/editEmail"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textSize="24dp"
    android:typeface="monospace"
    android:inputType="textEmailAddress">
    <requestFocus/>
</EditText>

<TextView
    android:id="@+id/textJenisKelamin"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Jenis Kelamin"
    android:typeface="monospace"
    android:textSize="24dp"
    android:textStyle="bold"/> >

<RadioGroup
    android:id="@+id/radioGroup1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content">
    <RadioButton
        android:id="@+id/radioLaki"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checked="true"
        android:text="Laki-Laki"/>

    <RadioButton
        android:id="@+id/radioPerempuan"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Perempuan"/>
</RadioGroup>
```

```

<TextView
    android:id="@+id/textHobi"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Hobi" android:textSize="24dp"/>

<CheckBox
    android:id="@+id/checkBaca"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Membaca"/>

<CheckBox
    android:id="@+id/checkMemancing"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Memancing"/>

<CheckBox
    android:id="@+id/checkBersepeda"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Bersepeda"/>

<Button
    android:id="@+id	btnOK"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="OK"/>

</LinearLayout>

```

7. Coba jalankan aplikasi, akan didapatkan tampilan sbb:



Akan tampak bahwa, ukuran layar secara vertical tidak cukup untuk menampilkan seluruh obyek view yang ada. Solusinya adalah dengan menambahkan Scroll untuk menggulung layar kearah vertical.

8. Tambahkan ScrollView, dan atur struktur dalam main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >

    <ScrollView
        android:id="@+id/scrollView1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" >

        <LinearLayout
            android:id="@+id/linearLayout1"
            android:layout_width="match_parent"
            android:layout_height="match_parent" >
        </LinearLayout>

    </ScrollView>

    <TextView
        android:id="@+id/textNama"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Nama Lengkap"
        android:typeface="monospace"
        android:textSize="24dp" />

```

Ubah menjadi :

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/scrollView1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <LinearLayout
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:orientation="vertical" >

        <TextView
            android:id="@+id/textNama"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Nama Lengkap"
            android:typeface="monospace"
            android:textSize="24dp"
            android:textStyle="bold" />

        ...
    </LinearLayout>

</ScrollView>
```

9. Tambahkan kode program ke dalam btnOK, pada mainActivity

Jika di klik OK, maka akan memunculkan identitas ke Form

```
package emha.android.materi4;

import android.app.Activity;
import android.os.Bundle;
import android.text.style.LineHeightSpan.WithDensity;
import android.view.View;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.EditText;
import android.widget.RadioButton;

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

        final EditText editNama = (EditText) findViewById(R.id.editNama);

        final EditText editEmail = (EditText)
        findViewById(R.id.editEmail);

        final RadioButton radioLaki = (RadioButton)
        findViewById(R.id.radioLaki);

        final RadioButton radioPerempuan = (RadioButton)
        findViewById(R.id.radioPerempuan);

        final CheckBox checkBaca = (CheckBox)
        findViewById(R.id.checkBaca);

        final CheckBox checkMemancing = (CheckBox)
        findViewById(R.id.checkMemancing);

        final CheckBox checkBersepeda = (CheckBox)
        findViewById(R.id.checkBersepeda);

        Button btnOK = (Button) findViewById(R.id.btnOK);

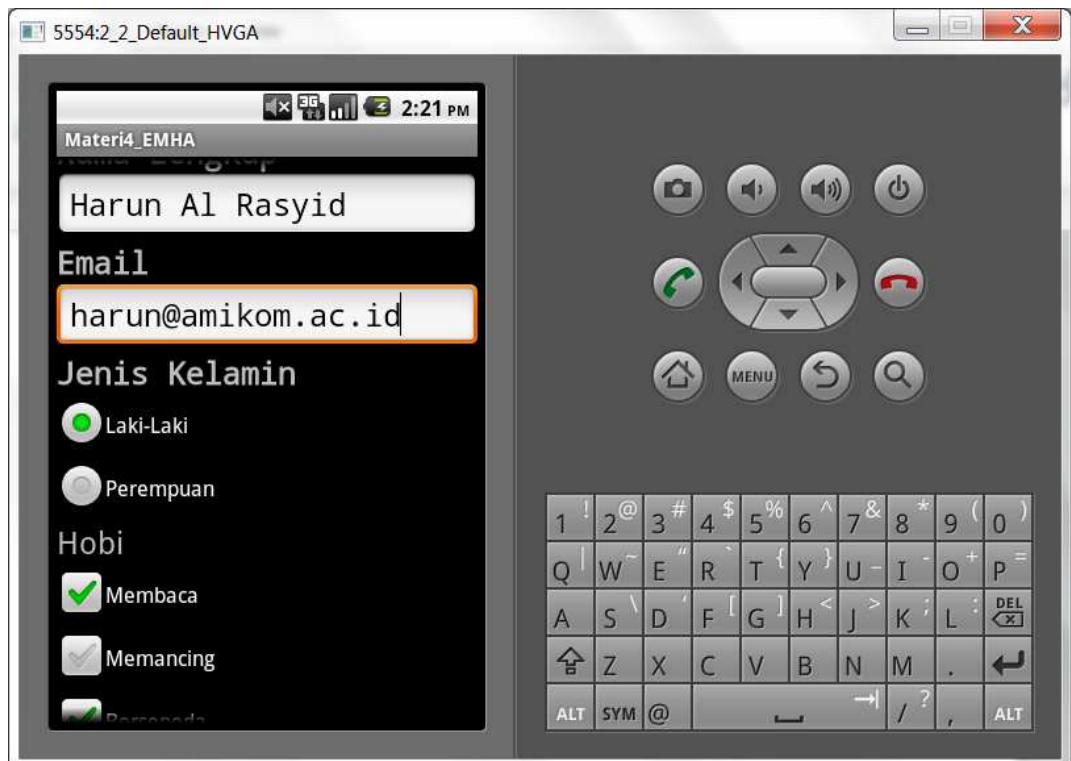
        btnOK.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View v) {
                // TODO Auto-generated method stub
                editNama.setText("Harun Al Rasyid");
                editEmail.setText("harun@amikom.ac.id");
                radioLaki.setChecked(true);
                checkBaca.setChecked(true);
                checkMemancing.setChecked(false);
                checkBersepeda.setChecked(true);
            }
        });
    }
}
```

10. Jalankan aplikasi



11. Klik button OK



MATERI 5

MEMAHAMI RESOURCES DAN APP WIDGET

- **Bekerja dengan Resources**
 - **Bekerja dengan App Widgets**
-

Android memiliki beberapa resources, antara lain:

1) *Layouts*

Layout merupakan bagian yang mendefinisikan bagaimana keadaan view dalam aplikasi. File layout defaultnya berada di direktori res/layouts. ADT secara default akan membuat file main.xml pada direktori tersebut.

LinearLayout	Layout yang menyusun bagian-bagiannya dalam satu baris
RelativeLayout	Layout yang posisi dari bagian-bagiannya dapat di deskripsikan dalam relasi satu dgn yg lainnya atau dgn induknya.
FrameLayout	Layout ini dirancang untuk memblokir area pada screen untuk menampilkan satu komponen. Developer dapat menambahkan beberapa anak bagian, tetapi semua akan dipatok ke kiri atas screen. Anak bagian dari layout akan digambarkan dalam stack, yang terakhir akan berada di atas stack.
TableLayout	Layout yang menyusun anak bagiannya dalam baris dan kolom

2) *Strings*

3) *Images*

4) *Dimensions*

Merupakan unit ukuran numerik:

- dp (*density-independent pixels*)
- sp (*scale-independent pixels*)
- pt (*points*)
- px (*pixels*)
- mm (*millimeters*)
- in (*inches*)

5) Styles

Styles di android sangat mirip dengan Cascading Style Sheets (CSS) di pengembangan web. Style merupakan kumpulan dari property yang dapat diterapkan pada View secara mandiri, Activity, atau keseluruhan aplikasi (melalui file manifest). Style mendukung *inheritance*. Contoh property style antara lain: *font size*, *font color*, dan *screen background*.

6) Themes

Theme merupakan Style yang diterapkan pada keseluruhan Activity atau Aplikasi. Jika Style diterapkan sebagai Theme dalam suatu Aplikasi, maka semua View dan Activity dalam Aplikasi akan Meng-Inherits setting dari Style.

7) Values

Value dapat mengandung berbagai tipe untuk aplikasi, diantaranya:

- **Bool**: merupakan nilai Boolean. Disimpan dalam file xml, biasanya di lokasi res/values/<filename>.xml;
Contoh: res/values/bools.xml
- **Integer**: merupakan nilai Integer. Disimpan dalam file xml, biasanya di lokasi res/values/<filename>.xml;
Contoh: res/values/integers.xml
- **Integer Array**: merupakan Array Integer. Disimpan dalam file xml, biasanya di lokasi res/values/<integers>.xml; Yang mana integers merupakan <nama file>
- **Typed Arrays**: merupakan tipe yang digunakan untuk membuat Array dari berbagai resource yang tidak sejenis. Biasanya disimpan di res/values/<filename>.xml;
Misal res/values/types.xml

8) Menus

Menu dapat didefinisikan melalui kode maupun XML. Cara awal untuk mendefinisikan menu adalah melalui XML. Biasanya menu diletakkan di direktori /menus. Setiap menu memiliki file .xml sendiri.

9) Colors

File colorsbiasanya diletakkan di values/colors.xml. File ini memungkinkan developer untuk mendefinisikan nama color, misal: login_screen_font_colors, yang berarti color dari font yang digunakan pada halaman login.

LATIHAN

1. Buka kembali project **PengalihModeSilent** yang pernah dibuat



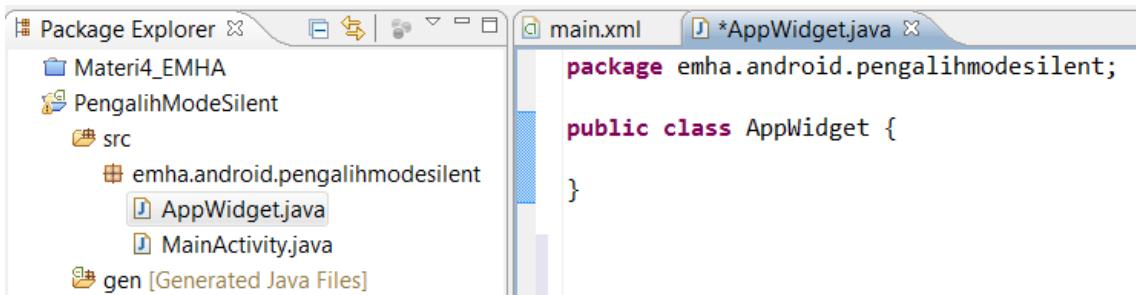
2. Pada sesi latihan ini akan coba di buat Home Screen Widget untuk Aplikasi **PengalihMode Silent**. Beberapa class yang akan digunakan:

- Intent
- BroadcastReceiver
- AppWidgetProvider
- IntentService
- AppWidgetProviderInfo

Home Screen Widget dalam android merupakan miniature aplikasi yang dapat dilekatkan pada aplikasi lain seperti pada Home Screen. Hal ini juga disebut App Widgets. App Widgets dapat menerima input dari pengguna melalui event click dan dapat meng-update diri mereka sendiri melalui jadwal yg regular. App Widget diimplementasikan pada *home screen* dengan *long-pressing*.

Untuk membuat Aplikasi PengalihModeSilent lebih bermanfaat, maka akan digunakan Home Screen Widget, sehingga pengguna bisa menambahkan aplikasi ini ke home screen perangkat mereka.

3. Tambahkan sebuah class baru pada project anda (**Aplikasi PengalihModeSilent**), pada package **emha.android.pengalihmodesilent**, dengan nama: AppWidget.java



4. Lengkapi kode program menjadi seperti berikut ini:

```
package emha.android.pengalihmodesilent;

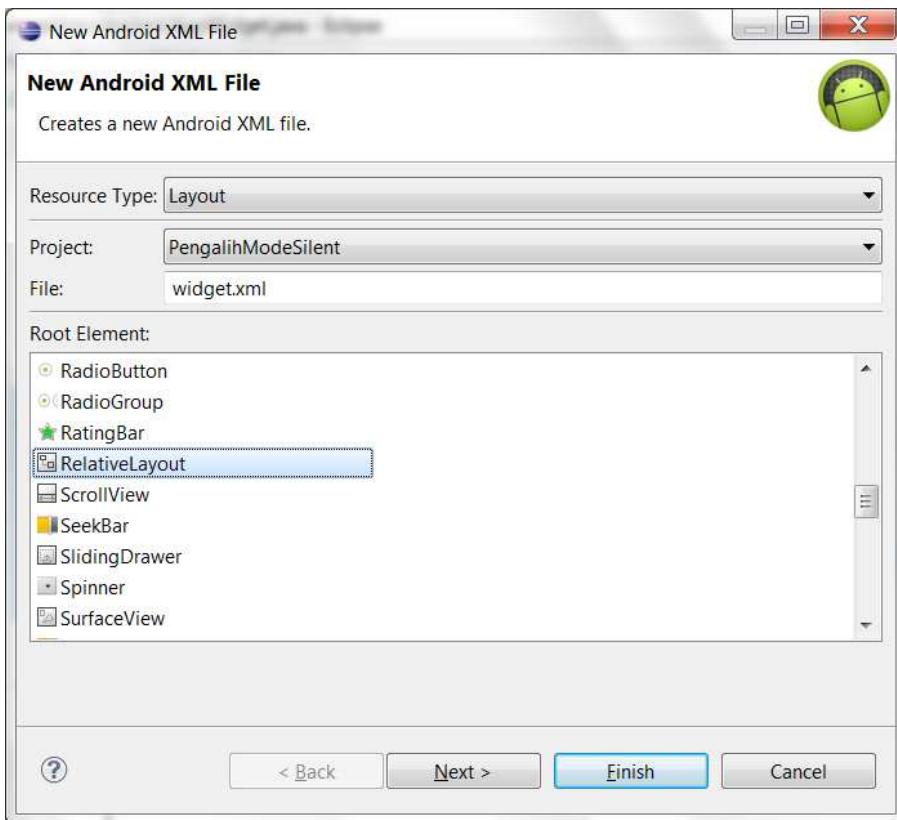
import android.appwidget.AppWidgetManager;
import android.appwidget.AppWidgetProvider;
import android.content.Context;
import android.content.Intent;

public class AppWidget extends AppWidgetProvider {

    @Override
    public void onReceive(Context context, Intent intent) {
        if (intent.getAction() == null) {
            //Lakukan sesuatu
        } else {
            super.onReceive(context, intent);
        }
    }

    @Override
    public void onUpdate(Context context, AppWidgetManager
            appWidgetManager, int[] appWidgetIds) {
        //Lakukan Sesuatu
    }
}
```

5. Pada layout, buat sebuah **layout baru** dengan nama **widget.xml**



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

```

6. Tambahkan sebuah ImageView, dan atur properties sehingga didapat kondisi widget.xml sebagai berikut:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent" >

    <ImageView
        android:id="@+id/kondisiPhone"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/icon"
        android:clickable="true"
        android:layout_centerInParent="true"/>

</RelativeLayout>
```



7. Lengkapi kode program pada src/ → widget.java, menjadi sbb:

```
1 package emha.android.pengalihmodesilent;
2
3 import android.app.Activity;
4 import android.app.IntentService;
5 import android.app.PendingIntent;
6 import android.appwidget.AppWidgetManager;
7 import android.appwidget.AppWidgetProvider;
8 import android.content.ComponentName;
9 import android.content.Context;
10 import android.content.Intent;
11 import android.media.AudioManager;
12 import android.widget.RemoteViews;
13
14 public class AppWidget extends AppWidgetProvider {
15
16     @Override
17     public void onReceive(Context context, Intent intent) {
18         if (intent.getAction() == null) {
19             //Lakukan sesuatu
20             context.startService(new Intent(context, ToggleService.class));
21         } else {
22             super.onReceive(context, intent);
23         }
24     }
25
26     @Override
27     public void onUpdate(Context context, AppWidgetManager
28                         appWidgetManager, int[] appWidgetIds) {
29         //Lakukan Sesuatu
30         context.startService(new Intent(context, ToggleService.class));
31
32     }
33
34 }
```

```

35 public static class ToggleService extends IntentService {
36     public ToggleService() {
37         super("AppWidget$ToggleService");
38     }
39
40     @Override
41     protected void onHandleIntent(Intent intent) {
42         ComponentName me=new ComponentName(this, AppWidget.class);
43         AppWidgetManager mgr=AppWidgetManager.getInstance(this);
44         mgr.updateAppWidget(me, buildUpdate(this));
45     }
46
47     private RemoteViews buildUpdate(Context context) {
48         RemoteViews updateViews = new
49 RemoteViews(context.getPackageName(), R.layout.widget);
50         AudioManager audioManager =
51 (AudioManager)context.getSystemService(Activity.AUDIO_SERVICE);
52         if(audioManager.getRingerMode() ==
53 AudioManager.RINGER_MODE_SILENT) {
54             updateViews.setImageResource(R.id.kondisiPhone,
55 R.drawable.phone_on);
56             audioManager.setRingerMode(AudioManager.RINGER_MODE_NORMAL);
57         } else {
58             updateViews.setImageResource(R.id.kondisiPhone,
59 R.drawable.phone_silent);
59 audioManager.setRingerMode(AudioManager.RINGER_MODE_SILENT);
60         }
61         Intent i = new Intent(this, AppWidget.class);
62         PendingIntent pi = PendingIntent.getBroadcast(context, 0, i,0);
63         updateViews.setOnClickListener(R.id.kondisiPhone,pi);
64         return updateViews;
65     }
66 }
67 }
68 }
69

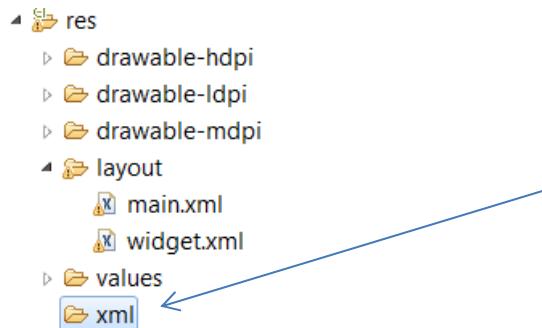
```

Baris	Keterangan
14	Menjelaskan bahwa class AppWidget ini merupakan class AppWidgetProvider karena meng-inherits class AppWidgetProvider
17	class AppWidget meng-override metode onReceive() yang memungkinkan untuk mendeteksi intent baru yang diterima dari RemoteView. intent dapat muncul diawali dengan aksi dari user, seperti button click
18	intent merupakan object yang mengandung berbagai data, diantaranya adalah action. Baris ini melakukan pengecekan apakah intent mengandung action
20	Baris kode ini memulai instance baru ToogleService. context pada baris ini merujuk pada Context Object dari android yang merupakan antarmuka terhadap informasi global dari aplikasi. Intent baru dibuat untuk memberitahukan android apa yg harus dilakukan.
22	Mendelegasikan tugas ke class induk, dikarenakan tidak perlu melakukan apapun karena intent tidak sesuai harapan (mengandung action)

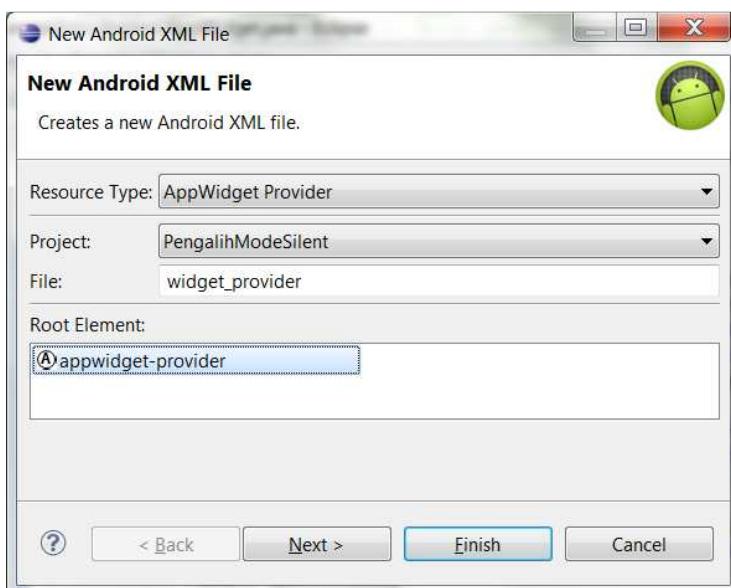
27, 28	onUpdate() metode ini dipanggil oleh android framework berdasarkan waktu, diatur di metadata.
30	Sama seperti baris 20
35	Ini merupakan implementasi dari IntentService. IntentService ini menangani proses sama seperti MainActivity yaitu merubah mode-phone dari normal ke silent dan sebaliknya. Ini merupakan implementasi dari background service pada android.
37	Metode ini memanggil superclass dengan nama AppWidget\$ToggleService. Pemanggilan metode ini dilakukan untuk menangani proses debugging dari thread. Jika baris ini dihapus maka, compiler akan memberikan pesan error yang menyatakan harus secara eksplisit memanggil super constructor. Jika app widget diberi nama lain selain AppWidget, maka baris ini harus disesuaikan.
41	Metode HandleIntent() bertanggung jawab untuk menangani intent yang diberikan ke service ini. Pada kasus ini adalah intent yang di buat pada baris 20 dan atau 30, selanjutnya object intent merupakan kurir yang mengintruksikan ToggleService untuk memulai proses.
42	Object ComponentName dibuat. Object ini digunakan bersama AppWidgetManager sebagai provider dari content baru yang akan di kirim ke app widget melalui RemoteView
43	Sebuah instance dari AppWidgetManager dibuat dengan pemanggilan metode static AppWidgetManager.getInstance(). AppWidgetManager merupakan class yang bertanggung jawab untuk meng-update state dari app widget dan menyediakan berbagai informasi tentang aplikasi app widget. Instance ini akan digunakan untuk meng-update state dari app widget.
44	App widget mengambil update dengan memanggil metode updateAppWidget() pada baris ini. Pemanggilan ini membutuhkan 2 hal : ComponentName yang melakukan update dan object RemoteView yang digunakan untuk mengupdate app widget. ComponentName di bentuk di baris 42.
47	Pendefinisian metode buildUpdate(). Metode ini mengembalikan object RemoteView yang digunakan di baris 44. Logika apa yang seharusnya terjadi berada di metode ini.
48, 49	Membangun object RemoteView dengan mengembalikan layout nya, R.layout.widget
62	Membuat object intent yang akan berjalan ketika class AppWidget dimulai
63	App Widget tidak dapat berkomunikasi secara langsung dengan intent, sehingga memerlukan PendingIntent. App Widget merupakan aplikasi dengan komunikasi cross process. Maka dibangun PendingIntent untuk memerintahkan AppWidget terhadap aksi berikutnya.

8. Mengatur App Widget Metadata

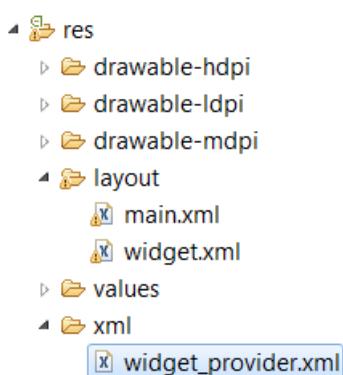
- Pada direktori res/
- Buat satu direktori baru, New → Folder, beri nama: xml



- Dalam folder xml, buat sebuah new android xml file, dengan nama: **widget_provider.xml**
- Tipe file tersebut adalah AppWidgetProvider**



- Klik Finish



```
<?xml version="1.0" encoding="utf-8"?>
<appwidget-provider xmlns:android="http://schemas.android.com/apk/res/android" >

</appwidget-provider>
```

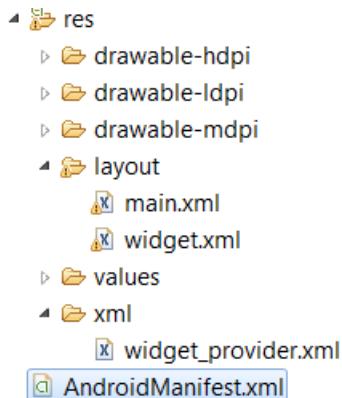
9. Lengkapi widget_provider.xml menjadi sbb:

```
<?xml version="1.0" encoding="utf-8"?>
<appwidget-provider xmlns:android="http://schemas.android.com/apk/res/android"
    android:minWidth="79px"
    android:minHeight="79px"
    android:updatePeriodMillis="1800000"
    android:initialLayout="@layout/widget" >

</appwidget-provider>
```

10. Langkah selanjutnya, mendaftarkan komponen baru ke manifest

a. Buka : AndroidManifest.xml



Akan tampak source xml sbb:

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="emha.android.pengalihmodesilent"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk android:minSdkVersion="8" />

    <application
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name" >
        <activity
            android:name=".MainActivity"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>

```

- b. Tambahkan elemen xml sehingga menjadi sbb (yg diblok orange adalah kode yg baru):

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="emha.android.pengalihmodesilent"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk android:minSdkVersion="8" />

    <application
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name" >
        <activity
            android:name=".MainActivity"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

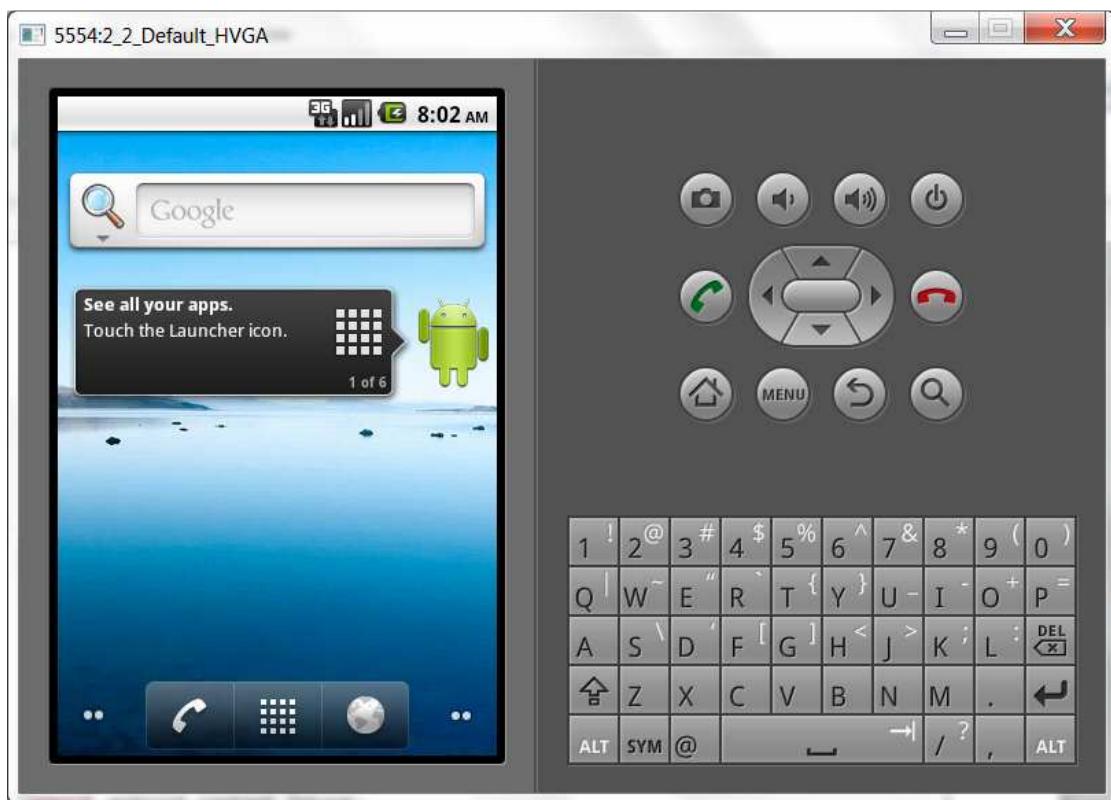
        <receiver
            android:name=".AppWidget"
            android:icon="@drawable/icon"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.appwidget.action.APPWIDGET_UPDATE" />
            </intent-filter>
            <meta-data
                android:name="android.appwidget.provider"
                android:resource="@xml/widget_provider" />
        </receiver>

        <service android:name=".AppWidget$ToggleService" />
    </application>
</manifest>

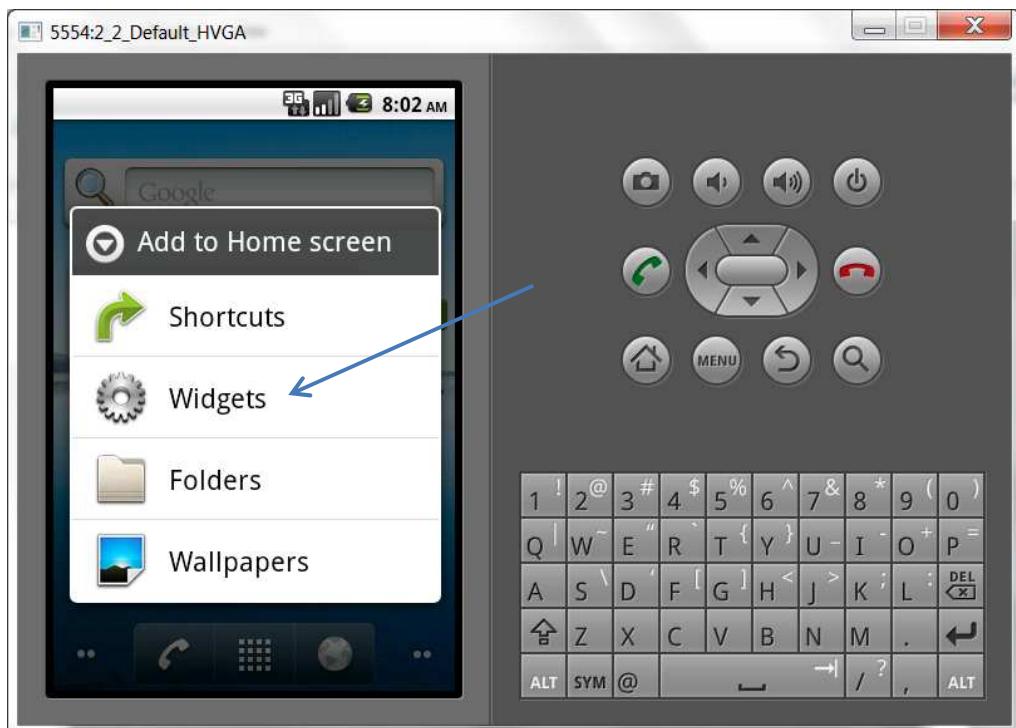
```

11. Selanjutnya, Meletakkan Widget di atas Home Screen

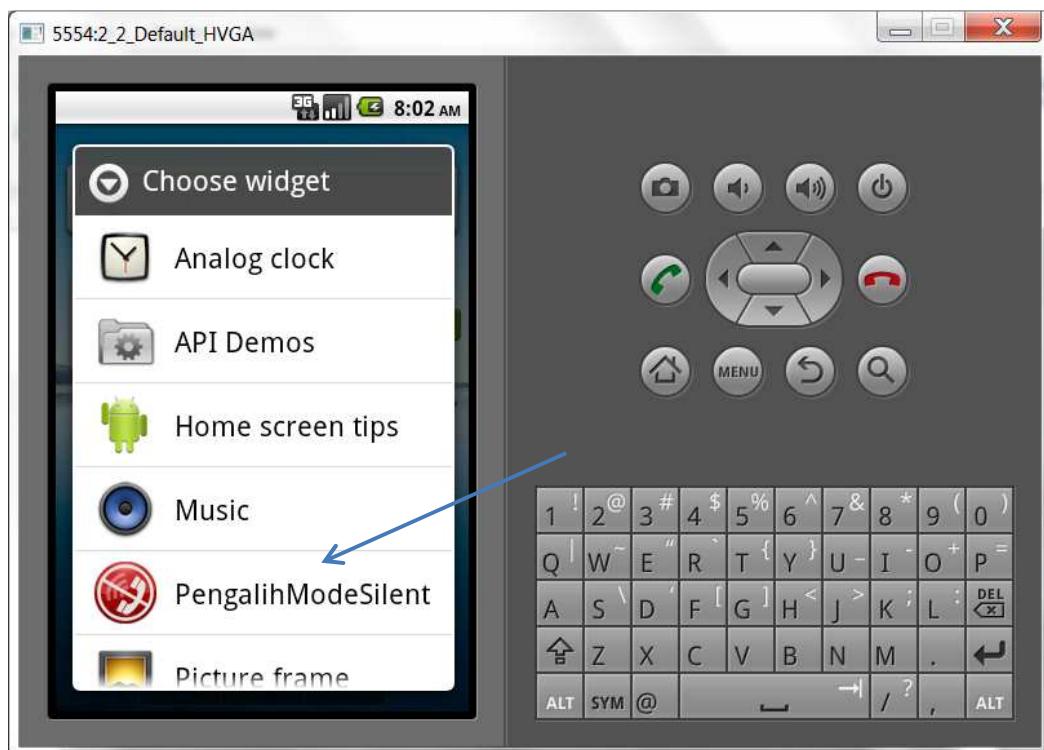
- a. Klik lama di home screen



- b. Akan muncul pop up, pilih **widgets**

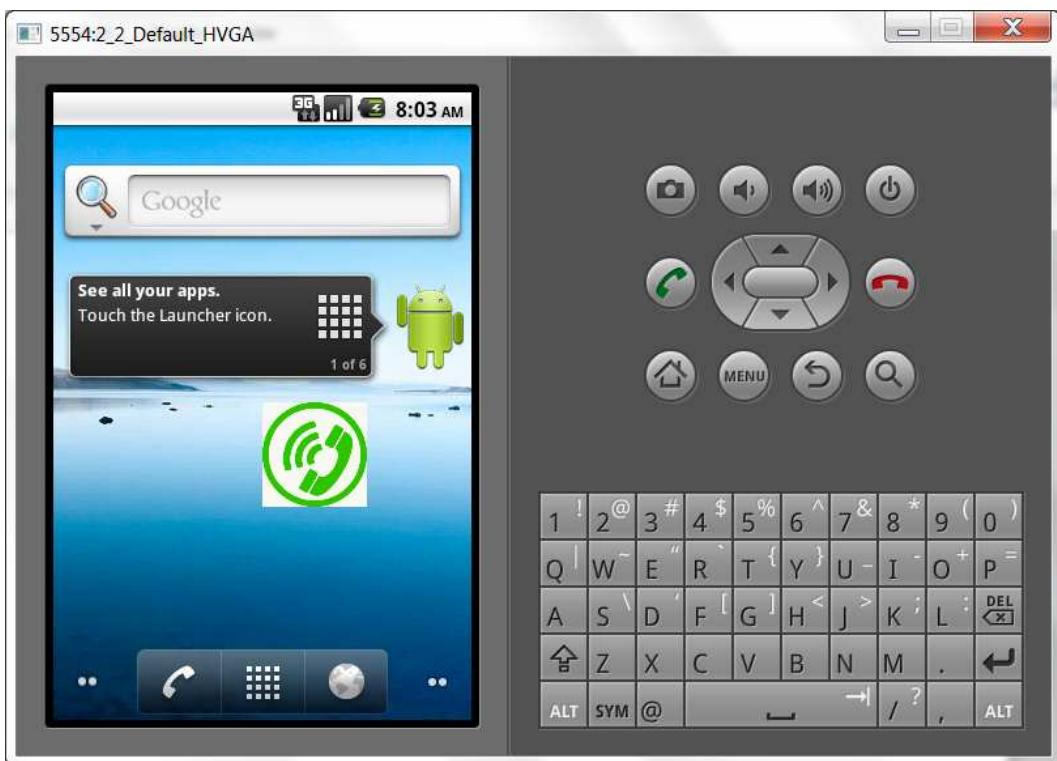


c. Kemudian Pilih PengalihModeSilent



12. Coba jalankan aplikasi, klik icon widgets PengalihModeSilent di home screen



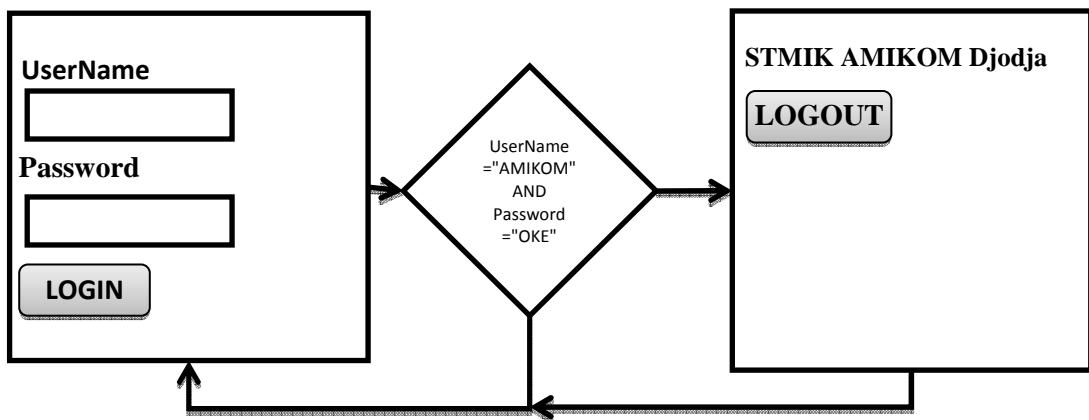


MATERI 6

BEKERJA DENGAN BEBERAPA ACTIVITY DAN LAYOUT

- Bekerja dengan beberapa Activity
- Bekerja dengan beberapa Layout

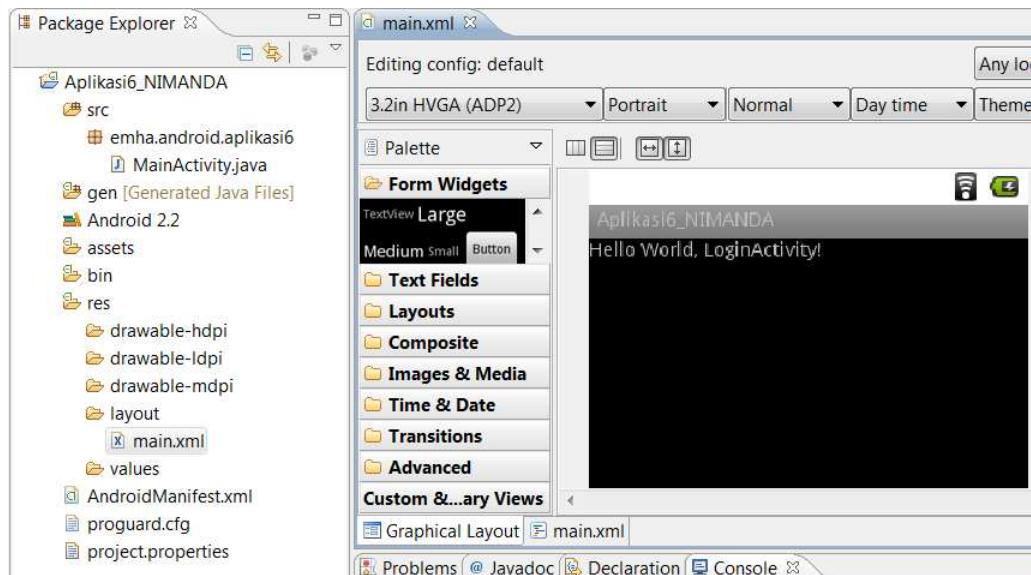
1. Pada latihan Materi ini akan dibangun aplikasi yang memiliki fungsi kerja sebagai berikut:



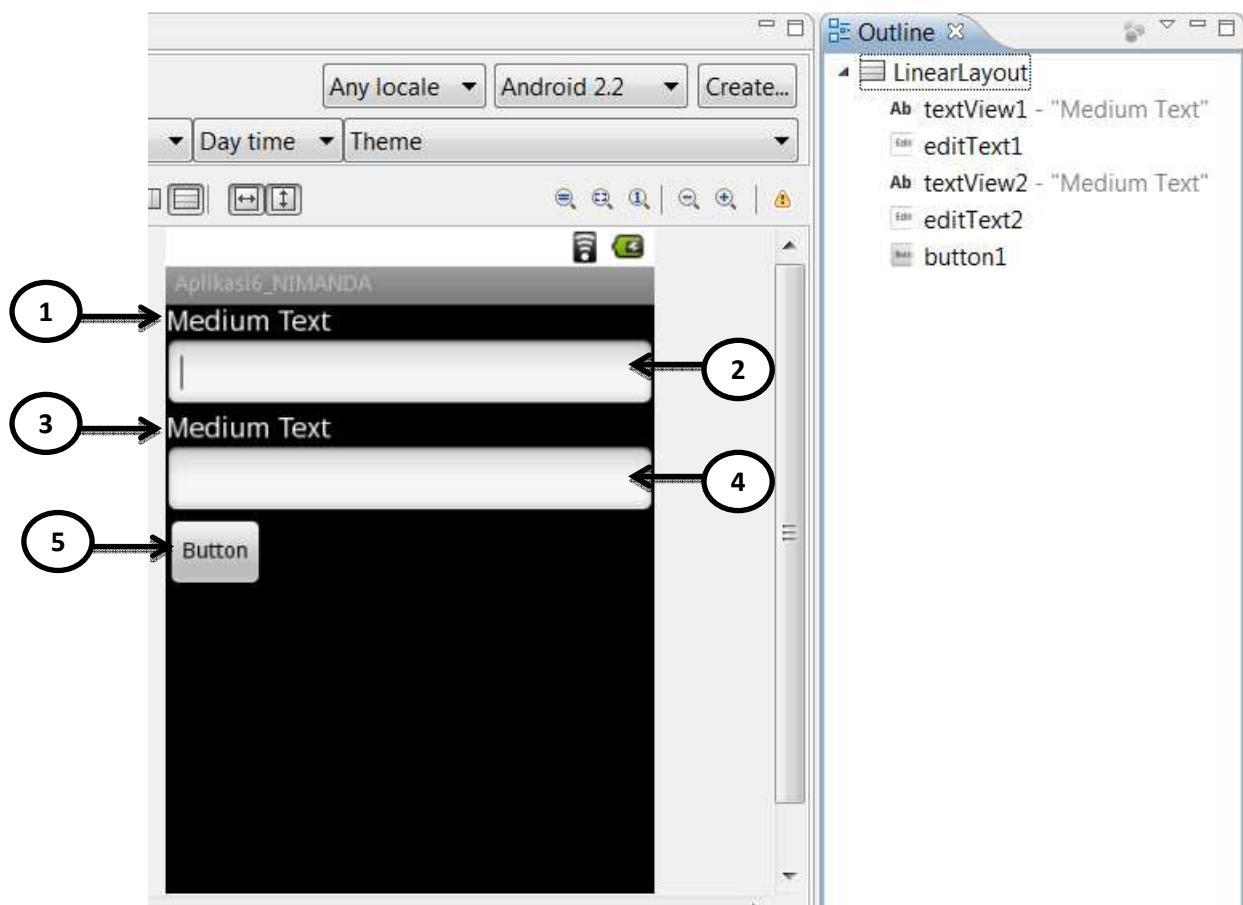
2. Buat Project Android baru dengan aturan sbb:

Project Name : Aplikasi6_NIMANDA
Build Target : Android 2.2
Application Name : Aplikasi6_NIMANDA
Package Name : emha.android.aplikasi6
Create Activity : MainActivity
Minimum SDK : 8

3. Bangun Tampilan Satu untuk fungsi Form Login di Layout main.xml



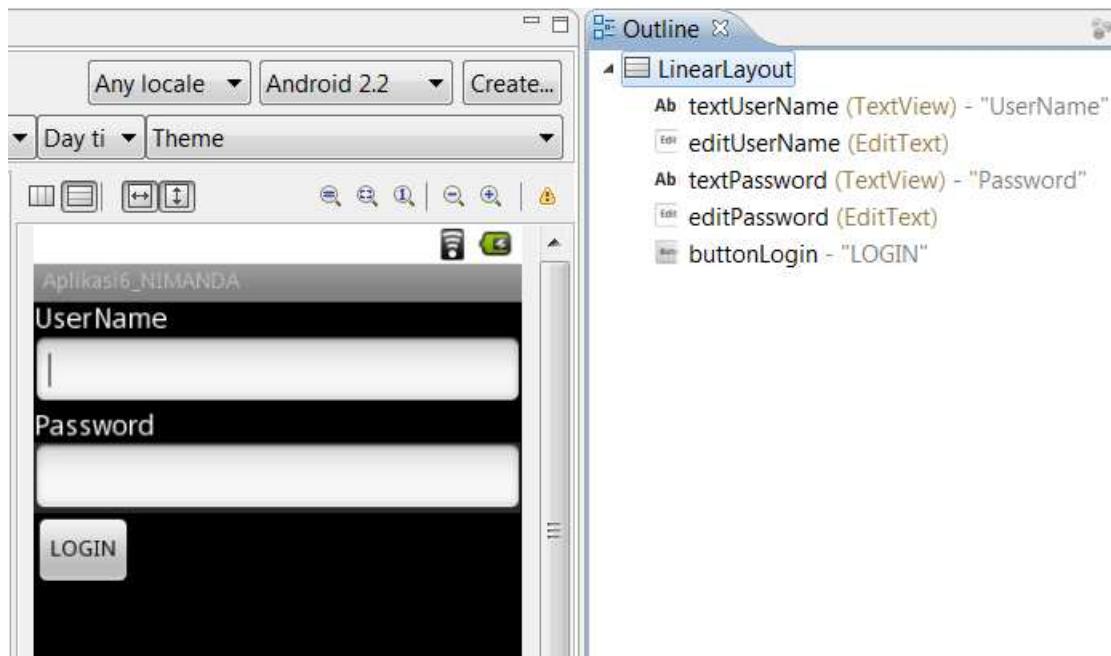
4. Ubah Layout main.xml menjadi sbb :



No	View Type	Property	Value
1	TextView (Medium)	Id	textUserName
		Text	UserName
2	EditText (PlainText)	Id	editUserName
3	TextView (Medium)	Id	textPassword
			Password
4	EditText (Password)	Id	editPassword
5	Button	Id	buttonLogin
		Text	LOGIN

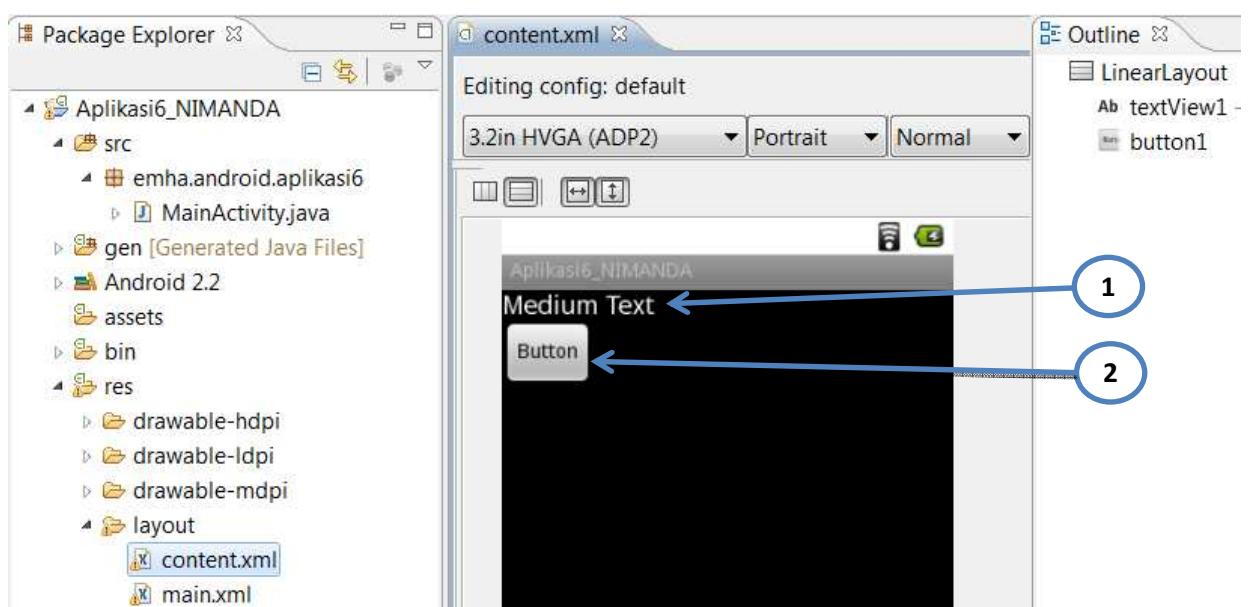
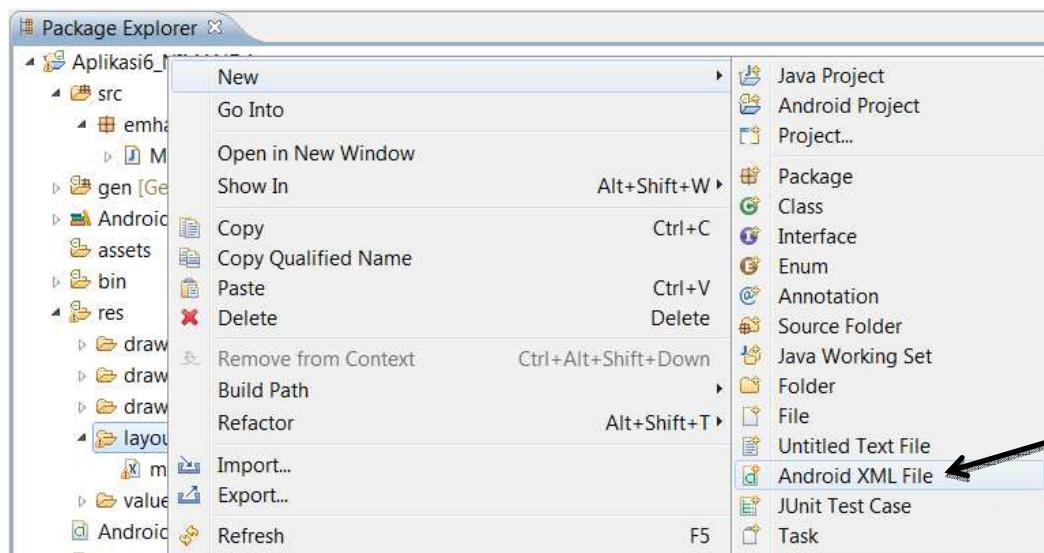
```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical">
    <TextView
        android:id="@+id/textUserName"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="Medium Text"
        android:textAppearance="?android:attr/textAppearanceMedium" />
    <EditText
        android:id="@+id/editUserName"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:hint="User Name"
        android:password="true" />
    <TextView
        android:id="@+id/textPassword"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="Medium Text"
        android:textAppearance="?android:attr/textAppearanceMedium" />
    <EditText
        android:id="@+id/editPassword"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:hint="Password"
        android:password="true" />
    <Button
        android:id="@+id/buttonLogin"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="LOGIN" />
</LinearLayout>
```

```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="UserName"
    android:textAppearance="?android:attr/textAppearanceMedium" />
<EditText
    android:id="@+id/editUserName"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
</EditText>
<TextView
    android:id="@+id/textPassword"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Password"
    android:textAppearance="?android:attr/textAppearanceMedium" />
<EditText
    android:id="@+id/editPassword"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:inputType="textPassword" />
<Button
    android:id="@+id/buttonLogin"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="LOGIN" />
</LinearLayout>
```



5. Tambahkan untuk Tampilan Dua, sebuah Layout dengan nama : content.xml

Buat file xml baru : content.xml



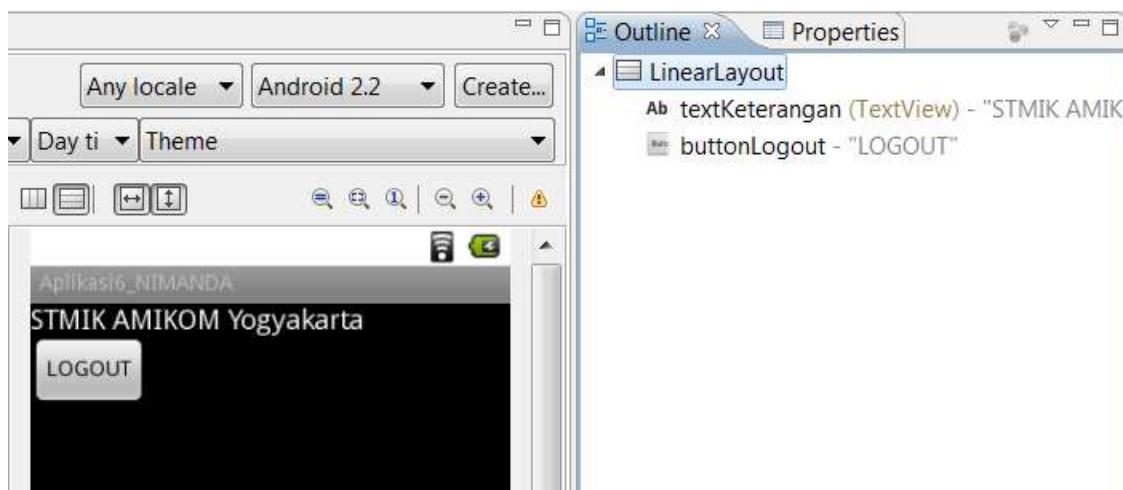
No	View Type	Property	Value
1	TextView (Medium)	Id	textKeterangan
		Text	STMIK AMIKOM Yogyakarta
2	Button	Id	buttonLogout
		Text	LOGOUT

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android
"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
```

```

<TextView
    android:id="@+id/textKeterangan"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="STMIK AMIKOM Yogyakarta"
    android:textAppearance="?android:attr/textAppearanceMedium" />
<Button
    android:id="@+id/buttonLogout"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="LOGOUT" />
</LinearLayout>

```



6. Buat kode program di **MainActivity.java**,

Jika BENAR UserName = "AMIKOM" dan Password = "OKE" maka tampilan pindah ke Tampilan DUA, jika SALAH Text Username dan Password di kosongkan kembali.

```

package emha.android.aplikasi6;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends Activity
implements View.OnClickListener {

    /** Called when the activity is first created. */
    EditText editUserName;
    EditText editPassword;
    Button buttonLogin;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}

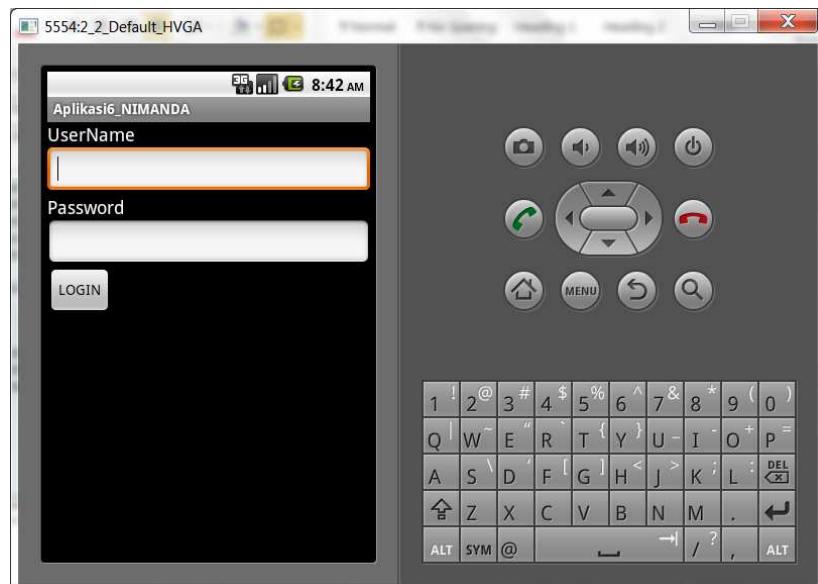
```

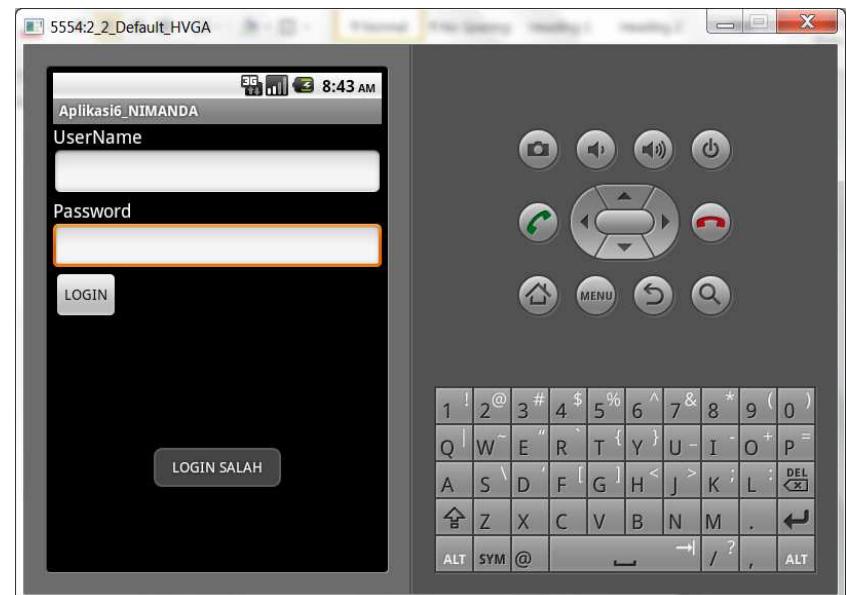
```

editUserName = (EditText) findViewById(R.id.editUserName);
editPassword = (EditText) findViewById(R.id.editPassword);
buttonLogin = (Button) findViewById(R.id.buttonLogin);
    buttonLogin.setOnClickListener(this);
}
@Override
public void onClick(View v) {
    // TODO Auto-generated method stub
    String user = editUserName.getText().toString();
    String pwd = editPassword.getText().toString();

    if (user.equals("AMIKOM") && pwd.equals("OKE")) {
        setContentView(R.layout.content);
    } else {
        Toast.makeText(this, "LOGIN SALAH", 18000);
        editUserName.setText("");
        editPassword.setText("");
    }
}
}

```





7. Tambahkan sebuah Activity baru, dengan nama : **ContentActivity.java**

Lengkapi kodennya menjadi sbb:

```
package emha.android.aplikasi6;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class ContentActivity extends Activity implements
View.OnClickListener {
    Button buttonLogout;
    Intent i;
```

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    // TODO Auto-generated method stub
    super.onCreate(savedInstanceState);
    setContentView(R.layout.content);
    buttonLogout = (Button) findViewById(R.id.buttonLogout);
    buttonLogout.setOnClickListener(this);

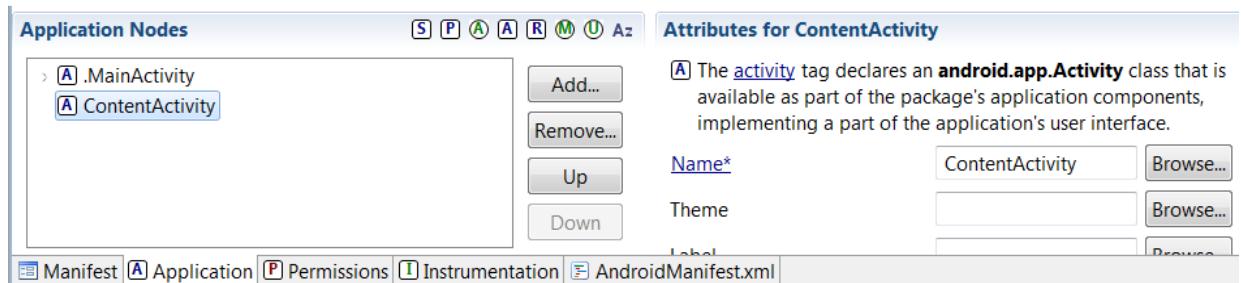
    i = new Intent(this,MainActivity.class);
}

@Override
public void onClick(View v) {
    // TODO Auto-generated method stub
    startActivity(i);
}

}

```

8. Daftarkan ContentActivity ke dalam Manifest



9. Ubah kode program MainActivity menjadi sbb:

```

package emha.android.aplikasi6;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends Activity
implements View.OnClickListener {
    /** Called when the activity is first created. */
    EditText editUserName;
    EditText editPassword;
    Button buttonLogin;

    Intent i;

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        editUserName = (EditText) findViewById(R.id.editUserName);
        editPassword = (EditText) findViewById(R.id.editPassword);
    }
}

```

```

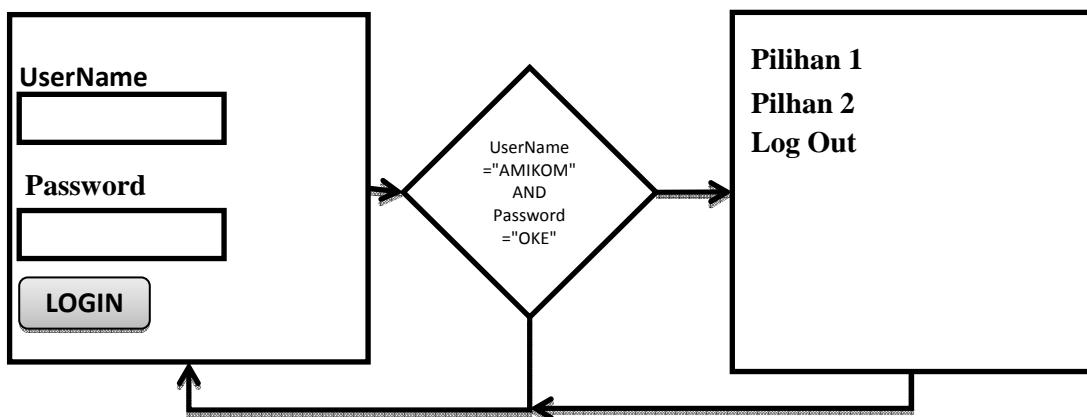
buttonLogin = (Button) findViewById(R.id.buttonLogin);
buttonLogin.setOnClickListener(this);

    i = new Intent(this, ContentActivity.class);
}
public void onClick(View v) {
    // TODO Auto-generated method stub
    String user = editUserName.getText().toString();
    String pwd = editPassword.getText().toString();
    if (user.equals("AMIKOM") && pwd.equals("OKE")) {
        startActivity(i);
    } else {
        Toast.makeText(this, "LOGIN SALAH", 16000).show();
        editUserName.setText("");
        editPassword.setText("");
    }
}
}

```

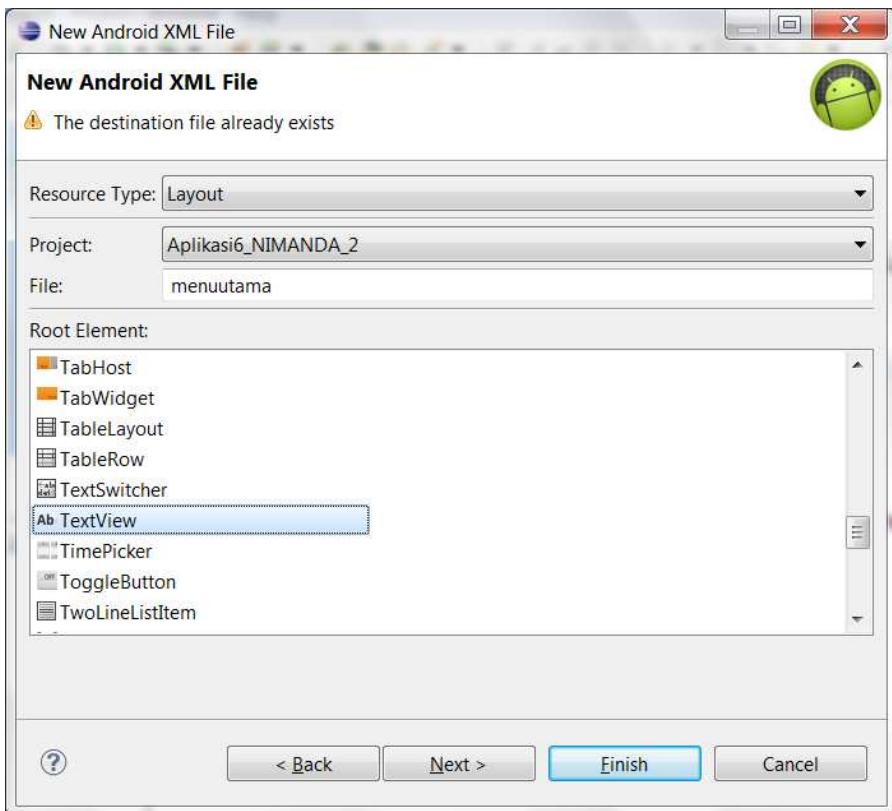
10. Jalankan

11. Menambahkan List sebagai menu utama



12. Pada Layout, buat AndroidXML baru,

Resource Type	: Layout
Filename	: menuutama
Root Element	: TextView



```
<?xml version="1.0" encoding="utf-8"?>
<TextView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

</TextView>
```

13. Buat Activity baru dengan nama MenuUtamaActivity.java

Lengkapi kode programnya menjadi sbb:

```
package emha.android.aplikasi6;

import android.app.ListActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import android.widget.TextView;
import android.widget.Toast;

public class MenuUtamaActivity extends ListActivity implements
OnItemClickListener {

    String [] pilihan = new String[]
    {"Pilihan 1", "Pilihan 2", "Log Out"};
```

```
Intent intentLogOut;

protected void onCreate(Bundle savedInstanceState) {
    // TODO Auto-generated method stub
    super.onCreate(savedInstanceState);

    ArrayAdapter<String> adapter =
new ArrayAdapter<String>(this, R.layout.menuutama, pilihan);

setListAdapter(adapter);

    ListView lv = getListView();
    lv.setOnItemClickListener(this);
    intentLogOut = new Intent(this, MainActivity.class);
}

public void onItemClick(AdapterView<?> arg0, View arg1,
int arg2, long arg3) {

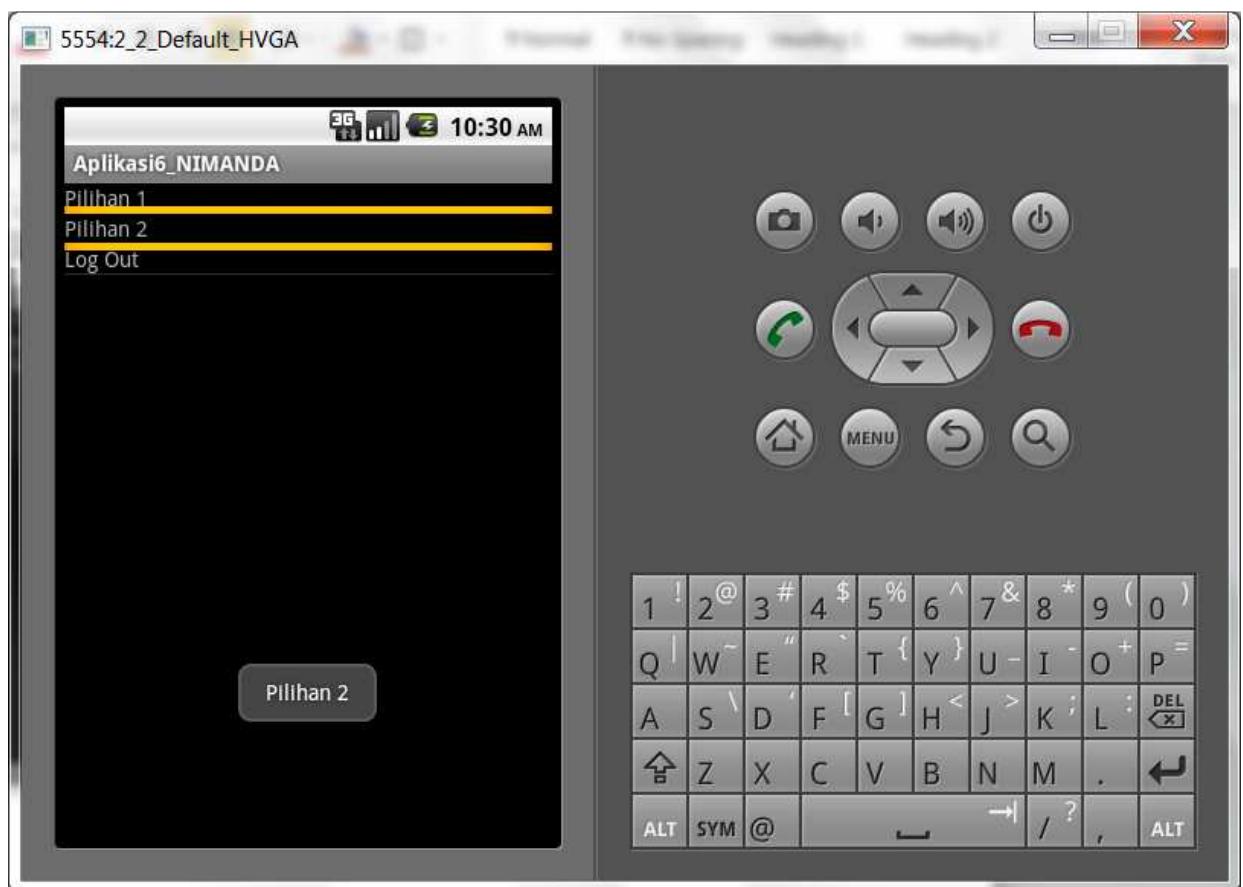
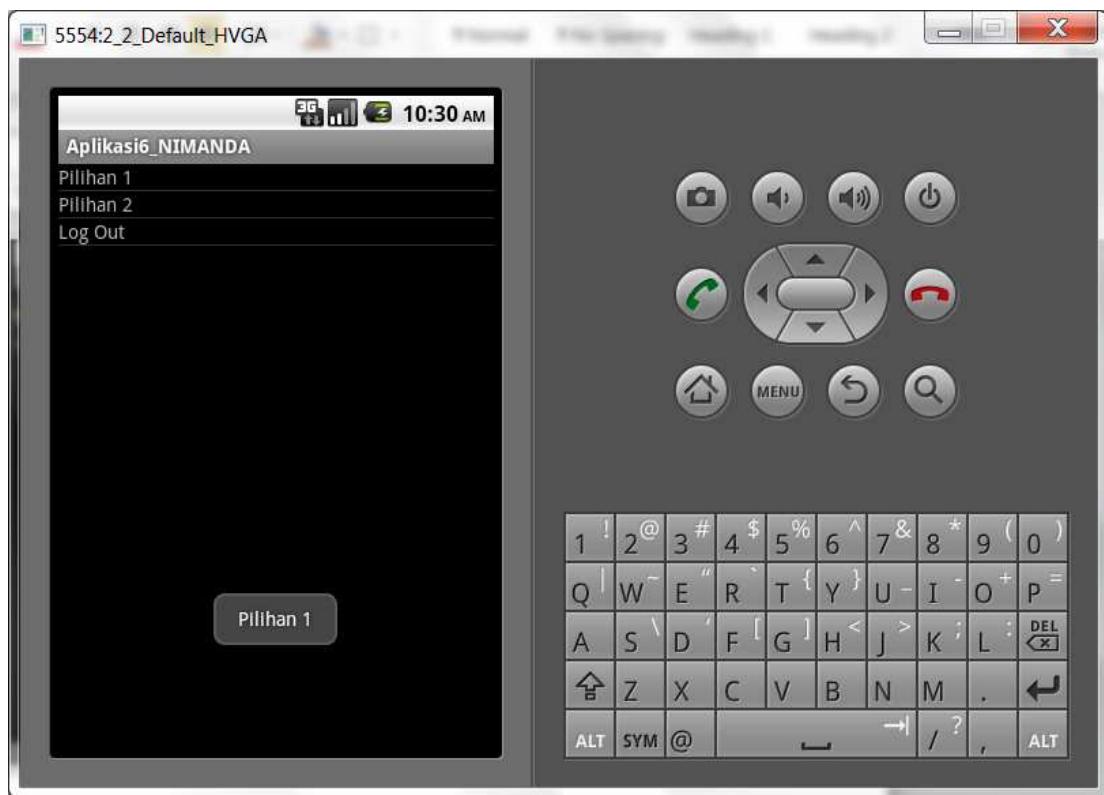
    String dipilih =
((TextView) arg1.getText().toString().trim());
    Toast.makeText(this, dipilih, 16000).show();

    if (dipilih.equals(pilihan[0])) {

    } else if (dipilih.equals(pilihan[1])) {

    } else if (dipilih.equals(pilihan[2])) {
        startActivity(intentLogOut);
    }

}
}
```

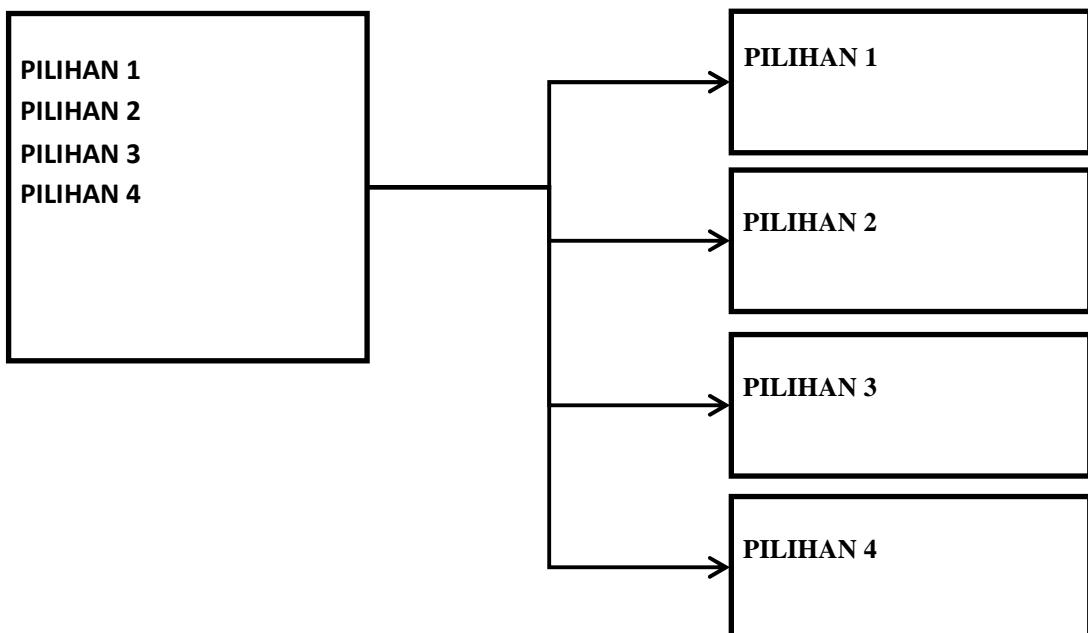


MATERI 7

BEKERJA DENGAN LIST MENU

- **Bekerja dengan ListActivity**
- **Bekerja dengan ArrayAdapter**
- **Bekerja dengan ListItemClick**

1. Pada latihan Materi ini akan dibangun aplikasi yang memiliki fungsi kerja sebagai berikut:



2. Buat Project Android baru dengan aturan sbb:

Project Name : Aplikasi7_NIMANDA
Build Target : Android 2.2
Application Name : Aplikasi7_NIMANDA
Package Name : emha.android.aplikasi7
Create Activity : MainActivity
Minimum SDK : 8

3. Tambahkan 1 buah Activity dengan nama MenuAwalActivity

The screenshot shows the Eclipse IDE interface. On the left, the Package Explorer view displays the project structure under 'Aplikasi7_NIMANDA' with 'src' and 'emha.android.aplikasi7' packages, containing 'MainActivity.java' and 'MenuAwalActivity.java'. On the right, the code editor shows the Java code for 'MenuAwalActivity':

```
package emha.android.aplikasi7;  
  
public class MenuAwalActivity {  
}
```

4. Ubah kode program MenuAwalActivity untuk membentuk List Menu menjadi sebagai berikut:

```
⊕ import android.app.ListActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.ListView;

public class MenuAwalActivity extends ListActivity {

    String pilihan[] = {"PILIHAN 1", "PILIHAN 2", "PILIHAN 3", "PILIHAN 4"};
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        // TODO Auto-generated method stub
        super.onCreate(savedInstanceState);
        setListAdapter(new ArrayAdapter<String>(MenuAwalActivity.this,
            android.R.layout.simple_list_item_1, pilihan));
    }

    @Override
    protected void onListItemClick(ListView l, View v, int position, long id) {
        // TODO Auto-generated method stub
        super.onListItemClick(l, v, position, id);
    }
}
```

5. Ubah AndroidManifest.xml menjadi seperti berikut:

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="emha.android.aplikasi7"
    android:versionCode="1"
    android:versionName="1.0" >
    <uses-sdk android:minSdkVersion="8" />
    <application
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name" >
        <activity
            android:name=".MenuAwalActivity"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity
            android:name=".MainActivity"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="emha.android.aplikasi7.MainActivity" />
                <category android:name="android.intent.category.DEFAULT" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

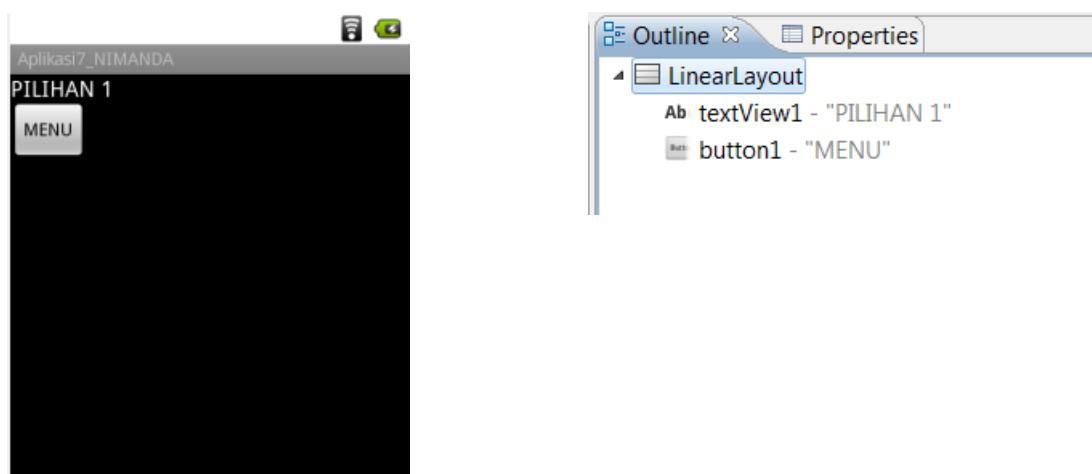
6. Coba jalankan aplikasi



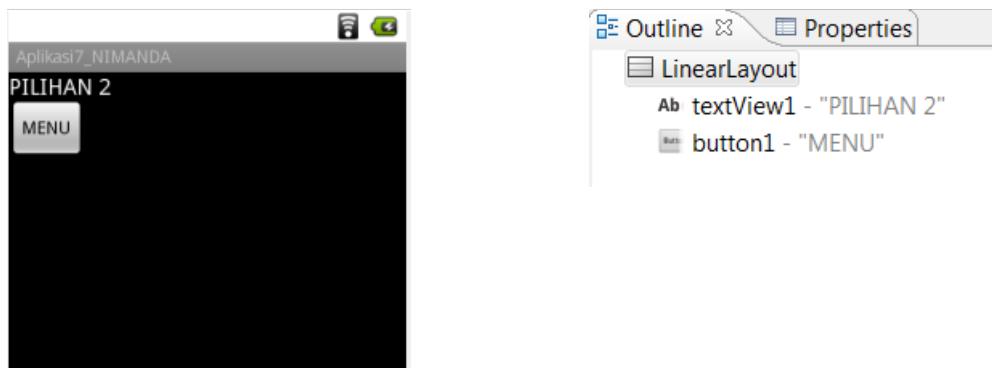
7. Tambahkan 4 buah Layout baru :

↳ layout
↳ main.xml
↳ pilihan1.xml
↳ pilihan2.xml
↳ pilihan3.xml
↳ pilihan4.xml

a. pilihan1.xml



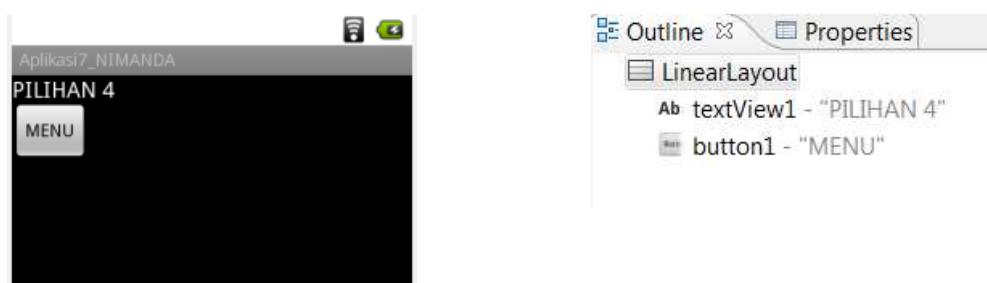
b. pilihan2.xml



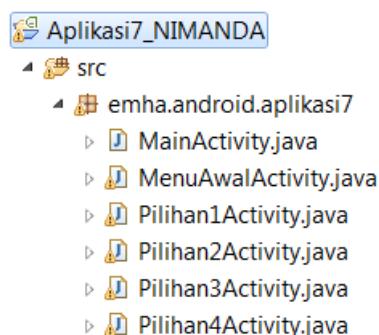
c. pilihan3.xml



d. pilihan4.xml



8. Tambahkan pula 4 buah Activity baru sebagai berikut :



a. Pilihan1Activity

```
public class Pilihan1Activity extends Activity{

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        // TODO Auto-generated method stub
        super.onCreate(savedInstanceState);
        setContentView(R.layout.pilihan1);

        Button btn = (Button) findViewById(R.id.button1);
        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                try {
                    Class menu = Class.forName("emha.android.aplikasi7.MenuAwalActivity");
                    Intent i = new Intent(Pilihan1Activity.this, menu);
                    startActivity(i);
                } catch (ClassNotFoundException e) {
                    // TODO Auto-generated catch block
                    e.printStackTrace();
                }
            }
        });
    }
}
```

b. Pilihan2Activity

```
public class Pilihan2Activity extends Activity{

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        // TODO Auto-generated method stub
        super.onCreate(savedInstanceState);
        setContentView(R.layout.pilihan1);

        Button btn = (Button) findViewById(R.id.button1);
        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                try {
                    Class menu = Class.forName("emha.android.aplikasi7.MenuAwalActivity");
                    Intent i = new Intent(Pilihan2Activity.this, menu);
                    startActivity(i);
                } catch (ClassNotFoundException e) {
                    // TODO Auto-generated catch block
                    e.printStackTrace();
                }
            }
        });
    }
}
```

c. Pilihan3Activity

```
public class Pilihan3Activity extends Activity{

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        // TODO Auto-generated method stub
        super.onCreate(savedInstanceState);
        setContentView(R.layout.pilihan1);

        Button btn = (Button) findViewById(R.id.button1);
        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                try {
                    Class menu = Class.forName("emha.android.aplikasi7.MenuAwalActivity");
                    Intent i = new Intent(Pilihan3Activity.this, menu);
                    startActivity(i);
                } catch (ClassNotFoundException e) {
                    // TODO Auto-generated catch block
                    e.printStackTrace();
                }
            }
        });
    }
}
```

d. Pilihan4Activity

```
public class Pilihan4Activity extends Activity{

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        // TODO Auto-generated method stub
        super.onCreate(savedInstanceState);
        setContentView(R.layout.pilihan1);

        Button btn = (Button) findViewById(R.id.button1);
        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                try {
                    Class menu = Class.forName("emha.android.aplikasi7.MenuAwalActivity");
                    Intent i = new Intent(Pilihan4Activity.this, menu);
                    startActivity(i);
                } catch (ClassNotFoundException e) {
                    // TODO Auto-generated catch block
                    e.printStackTrace();
                }
            }
        });
    }
}
```

9. Ubah MenuAwalActivity.java menjadi sebagai berikut:

```
public class MenuAwalActivity extends ListActivity {  
    String pilihan[] = {"PILIHAN 1", "PILIHAN 2", "PILIHAN 3", "PILIHAN 4"};  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        // TODO Auto-generated method stub  
        super.onCreate(savedInstanceState);  
        setListAdapter(new ArrayAdapter<String>(MenuAwalActivity.this,  
            android.R.layout.simple_list_item_1, pilihan));  
    }  
    @Override  
    protected void onListItemClick(ListView l, View v, int position, long id) {  
        // TODO Auto-generated method stub  
        super.onListItemClick(l, v, position, id);  
        try {  
            String x = String.valueOf(position+1);  
            Class classdijalankan = Class.forName("emha.android.aplikasi7.Pilihan"+ x +"Activity");  
            Intent i = new Intent(MenuAwalActivity.this, classdijalankan);  
            startActivity(i);  
        } catch (ClassNotFoundException e) {  
            // TODO Auto-generated catch block  
            e.printStackTrace();  
        }  
    }  
}
```

10. Ubah AndroidManifest.xml menjadi sbb:

```
<?xml version="1.0" encoding="utf-8"?>  
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
    package="emha.android.aplikasi7"  
    android:versionCode="1"  
    android:versionName="1.0">  
  
<uses-sdk android:minSdkVersion="8"/>  
  
<application  
    android:icon="@drawable/ic_launcher"  
    android:label="@string/app_name">  
    <activity  
        android:name=".MenuAwalActivity"  
        android:label="@string/app_name">  
        <intent-filter>  
        <action android:name="android.intent.action.MAIN"/>  
        <category android:name="android.intent.category.LAUNCHER"/>  
        </intent-filter>  
    </activity>  
    <activity  
        android:name=".MainActivity"  
        android:label="@string/app_name">  
        <intent-filter>  
        <action android:name="emha.android.aplikasi7.MainActivity"/>  
        <category android:name="android.intent.category.DEFAULT"/>  
        </intent-filter>  
    </activity>  
    <activity  
        android:name=".Pilihan1Activity"  
        android:label="@string/app_name">
```

```
<intent-filter>
<actionandroid:name="emha.android.aplikasi7.Pilihan1Activity"/>
<categoryandroid:name="android.intent.category.DEFAULT"/>
</intent-filter>
</activity>
<activity
    android:name=".Pilihan2Activity"
    android:label="@string/app_name">
    <intent-filter>
        <actionandroid:name="emha.android.aplikasi7.Pilihan2Activity"/>
        <categoryandroid:name="android.intent.category.DEFAULT"/>
    </intent-filter>
</activity>
<activity
    android:name=".Pilihan3Activity"
    android:label="@string/app_name">
    <intent-filter>
        <actionandroid:name="emha.android.aplikasi7.Pilihan3Activity"/>
        <categoryandroid:name="android.intent.category.DEFAULT"/>
    </intent-filter>
</activity>
<activity
    android:name=".Pilihan4Activity"
    android:label="@string/app_name">
    <intent-filter>
        <actionandroid:name="emha.android.aplikasi7.Pilihan4Activity"/>
        <categoryandroid:name="android.intent.category.DEFAULT"/>
    </intent-filter>
</activity>
</application>
</manifest>
```

MATERI 8

BEKERJA DENGAN SQLITE

- Memahami Cara Membuat, dan Mengelolala Database
 - Bekerja dengan SQL Query
-

SQLite merupakan embedded database yang sangat popular. Banyak perusahaan seperti adobe, apple, google, sun dan Symbian serta Mozilla, php dan python menggunakan SQLite. Android memasukkan SQLite ke dalam runtime Android, sehingga setiap aplikasi Android dapat memanfaatkan database SQLite.

SQLite menggunakan dialek SQL untuk query (SELECT), manipulasi data (INSERT, dll), dan definisi data (CREATE TABLE, dll).

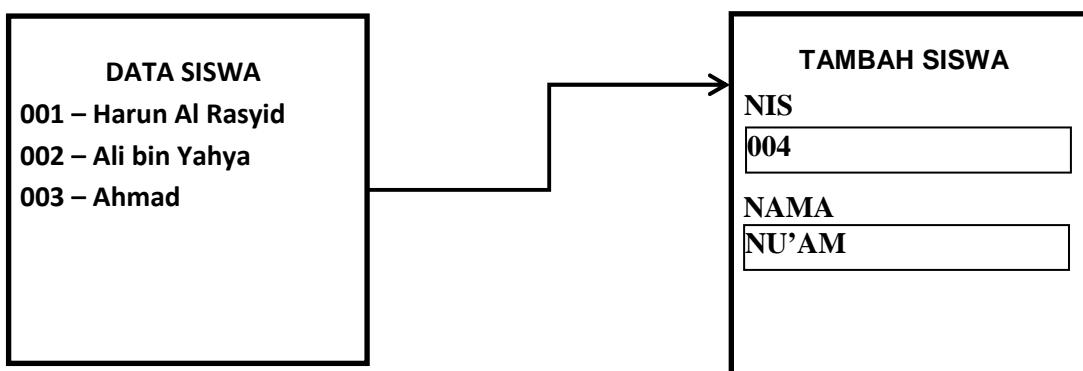
Perbedaan utama SQLite dengan database SQL lainnya adalah dalam penulisan data. Ketika pembuatan table, dapat di tentukan tipe data untuk satu kolom. Akan tetapi SQLite akan menggunakan itu sebagai petunjuk, admin dapat apapun dalam dalam suatu kolom, misal: data string di kolom integer dan sebaliknya. Selain itu juga terdapat beberapa fitur standar SQL yang tidak di support, antara lain: Foreign Key, Nested Transactions, Right Outer Join, Full Outer Join dan Alter Table. Selebihnya semua Fitur standar SQL didukung.

- Membuat Database

Untuk membuat dan membuka database, salah satu cara menggunakan SQLiteOpenHelper. SQLiteOpenHelper memiliki 3 komponen, yaitu:

- Contructor,
- Method onCreate(),
- Method onUpgrade(),

1. Pada Latihan 8 Materi ini akan dibangun aplikasi yang memiliki fungsi kerja sebagai berikut:



2. Buat Project Android baru dengan aturan sbb:

Project Name : Aplikasi8_NIMANDA
Build Target : Android 2.2
Application Name : Aplikasi8_NIMANDA
Package Name : emha.android.aplikasi8
Create Activity : MainActivity
Minimum SDK : 8

3. Dalam project Aplikasi8_NIMANDA, tambahkan satu buah class baru dengan nama class :

Siswa

Lengkapi kode class Siswa menjadi sbb:

```
package emha.android.aplikasi8;

public class Siswa {
    private String nis;
    private String nama;

    public Siswa() {
    }
    public Siswa(String nis, String nama) {
        this.nis = nis;
        this.nama = nama;
    }
    public String getNis() {
        return nis;
    }
    public void setNis(String nis) {
        this.nis = nis;
    }
    public String getNama() {
        return nama;
    }
    public void setNama(String nama) {
        this.nama = nama;
    }
}
```

4. Dalam project Aplikasi8_NIMANDA, tambahkan satu buah class baru dengan nama class :

DatabaseHandler

Lengkapi kode class DatabaseHandler menjadi sbb:

```
package emha.android.aplikasi8;

import java.util.ArrayList;
import java.util.List;
import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
```

```

public class DatabaseHandler extends SQLiteOpenHelper {
    private static final int DATABASE_VERSION = 1;
    // Nama Database
    private static final String DATABASE_NAME = "Sekolah";
    // Nama Table
    private static final String TABLE_SISWA = "Siswa";
    // NamaKolom Table Siswa
    private static final String KEY_NIS = "nis";
    private static final String KEY_NAMA = "nama";

    public DatabaseHandler(Context context) {
        // TODO Auto-generated constructor stub
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        // TODO Auto-generated method stub
        String query_table_siswa = "CREATE TABLE " + TABLE_SISWA + "("
            + KEY_NIS + " TEXT PRIMARY KEY," + KEY_NAMA + " TEXT)";
        db.execSQL(query_table_siswa);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        // TODO Auto-generated method stub
        // Drop older table if existed
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_SISWA);
        // Create tables again
        onCreate(db);
    }

    // MenambahSiswaBaru
    public void addSiswa(Siswa siswa) {
        SQLiteDatabase db = this.getWritableDatabase();

        ContentValues values = new ContentValues();
        values.put(KEY_NIS, siswa.getNis());
        values.put(KEY_NAMA, siswa.getNama());

        // Inserting Row
        db.insert(TABLE_SISWA, null, values);
        db.close(); //
    }

    // MembacaSiswa
    public Siswa getSiswa(String nis) {
        SQLiteDatabase db = this.getReadableDatabase();

        Cursor cursor = db.query(TABLE_SISWA, new String[] {
            KEY_NIS, KEY_NAMA}, KEY_NIS + "=?", 
            new String[] {nis}, null, null, null);

        if (cursor != null)
            cursor.moveToFirst();

        Siswa siswa = new Siswa(cursor.getString(0), cursor.getString(1));
        return siswa;
    }
}

```

```

// MembacaSemuaSiswa
public List<Siswa> getSemuaSiswa() {
    List<Siswa> siswaList = new ArrayList<Siswa>();

    String query_select_siswa = "SELECT * FROM " + TABLE_SISWA;

    SQLiteDatabase db = this.getWritableDatabase();
    Cursor cursor = db.rawQuery(query_select_siswa, null);

    if (cursor.moveToFirst()) {
        do {
            Siswa siswa = newSiswa(cursor.getString(0),
cursor.getString(1));
            siswaList.add(siswa);
        } while (cursor.moveToNext());
    }
    return siswaList;
}

public void deleteSiswa(Siswa siswa) {
    SQLiteDatabase db = this.getWritableDatabase();
    db.delete(TABLE_SISWA, KEY_NIS + " = ?",
    new String[] { siswa.getNis()});
    db.close();
}
}

```

5. Lengkapi kode program di **MainActivity**, menjadi sbb:

```

package emha.android.aplikasi8;

import java.util.List;
import android.app.Activity;
import android.os.Bundle;
import android.util.Log;

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

        DatabaseHandler db = newDatabaseHandler(this);

        // Tambah Siswa
        Log.d("Tambah Siswa: ", "MenambahData Siswa..");
        db.addSiswa(new Siswa("001", "Harun Al Rasyid"));
        db.addSiswa(new Siswa("002", "Ali bin Yahya"));
        db.addSiswa(new Siswa("003", "Ahmad"));

        // Membaca Semua Siswa
        Log.d("Baca Siswa: ", "Membaca Semua Data Siswa..");
        List<Siswa> siswa = db.getSemuaSiswa();

        for (Siswa s : siswa) {
            String log = "NIS: " + s.getNis() + ", Nama: " + s.getNama();
            Log.d("Name: ", log);
        }
    }
}

```

6. Coba jalankan aplikasi, lihat hasil pembacaan data pada log melalui LogCat

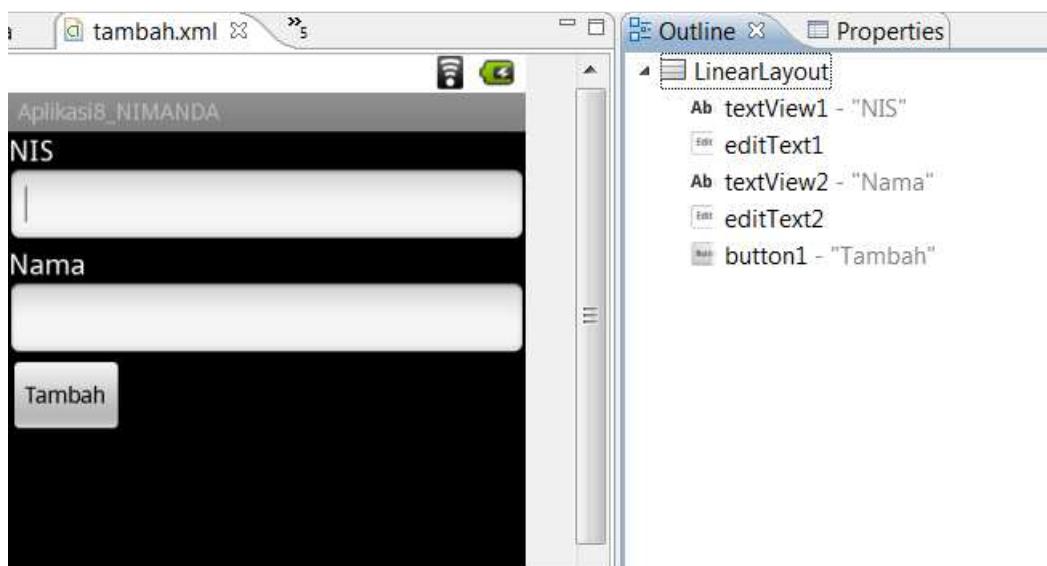
(Window → Show View → Other → Android → LogCat)

The screenshot shows the LogCat window with the following log entries:

PID	Application	Tag	Text
309	emha.android.a...	Reading:	Reading all contacts..
309	emha.android.a...	Name:	NIS: 001 ,Name: Harun Al Rasyid
309	emha.android.a...	Name:	NIS: 002 ,Name: Ali bin Yahya
309	emha.android.a...	Name:	NIS: 003 ,Name: Ahmad

7. Buat sebuah Android Layout baru, dengan nama **tambah.xml**

Tambahkan view menjadi sebagai berikut:



8. Tambahkan sebuah Activity dengan nama **TambahActivity**,

Tambahkan kode program menjadi sebagai berikut:

```
package emha.android.aplikasi8;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

public class TambahActivity extends Activity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        // TODO Auto-generated method stub
        super.onCreate(savedInstanceState);
        setContentView(R.layout.tambah);

        final DatabaseHandler db = new DatabaseHandler(this);
```

```

final EditText editNis = (EditText) findViewById(R.id.editText1);
final EditText editNama = (EditText) findViewById(R.id.editText2);

Button btnTambah = (Button) findViewById(R.id.button1);
btnTambah.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {
        // TODO Auto-generated method stub
        String nis = editNis.getText().toString();
        String nama = editNama.getText().toString();
        db.addSiswa(new Siswa(nis, nama));
        editNis.setText("");
        editNama.setText("");

        try {
            Class c = Class.forName("emha.android.aplikasi8.MainActivity");
            Intent i = new Intent(TambahActivity.this, c);
            startActivity(i);
        } catch (ClassNotFoundException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }
});
}
}

```

9. Tambahkan Activity TambahActivity ke dalam AndroidManifest

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="emha.android.aplikasi8"
    android:versionCode="1"
    android:versionName="1.0">
    <uses-sdk android:minSdkVersion="8"/>
    <application
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name">
        <activity
            android:name=".MainActivity"
            android:label="@string/app_name">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity
            android:name=".TambahActivity"
            android:label="@string/app_name">
            <intent-filter>
                <action android:name="emha.android.aplikasi8.TambahActivity" />
                <category android:name="android.intent.category.DEFAULT" />
            </intent-filter>
        </activity>
    </application>
</manifest>

```

10. Ubah Main Activity, menjadi sbb:

```

package emha.android.aplikasi8;

import java.util.List;

import android.app.ListActivity;
import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.ContextMenu;
import android.view.ContextMenu.ContextMenuItemInfo;
import android.view.MenuItem;
import android.view.View;
import android.widget.ArrayAdapter;

public class MainActivity extends ListActivity {

    String dataSiswa[] = null;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        //setContentView(R.layout.main);

        DatabaseHandler db = new DatabaseHandler(this);

        // TambahSiswa
        Log.d("Insert: ", "Inserting ..");
        db.addSiswa(new Siswa("001", "Harun Al Rasyid"));
        db.addSiswa(new Siswa("002", "Ali bin Yahya"));
        db.addSiswa(new Siswa("003", "Ahmad"));

        Log.d("Reading: ", "Reading all ..");
        List<Siswa> siswa = db.getSemuaSiswa();

        dataSiswa = new String[siswa.size()];

        int i=0;
        for (Siswa s : siswa) {
            String log = "NIS: " + s.getNis() + ", Nama: " + s.getNama();
            Log.d("Nama: ", log);
            dataSiswa[i] = s.getNis() + " - " + s.getNama();
            i++;
        }

        setListAdapter(new ArrayAdapter<Object>(this,
                android.R.layout.simple_list_item_1, dataSiswa));
        registerForContextMenu(getListView());
    }

    @Override
    public void onCreateContextMenu(ContextMenu menu, View v,
                                   ContextMenuInfo menuInfo) {
        // TODO Auto-generated method stub
        super.onCreateContextMenu(menu, v, menuInfo);

        menu.setHeaderTitle("Action");
        menu.add("Tambah");
    }

    @Override
    public boolean onContextItemSelected(MenuItem item) {

```

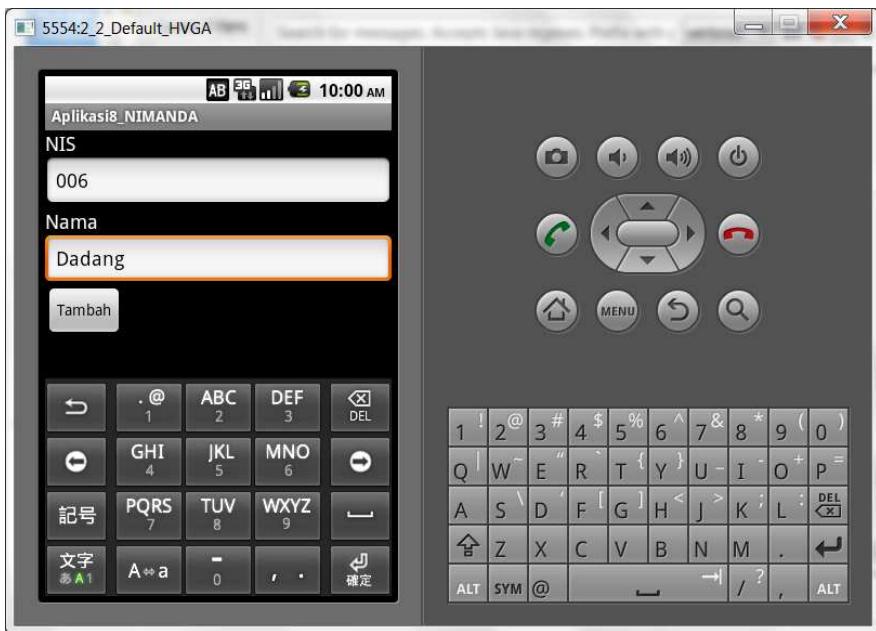
```

// TODO Auto-generated method stub
try {
    Class c =
Class.forName("emha.android.aplikasi8.TambahActivity");
    Intent i = newIntent(MainActivity.this, c);
    startActivity(i);
} catch (ClassNotFoundException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
returntrue;
}
}

```

11. Jalankan





LATIHAN

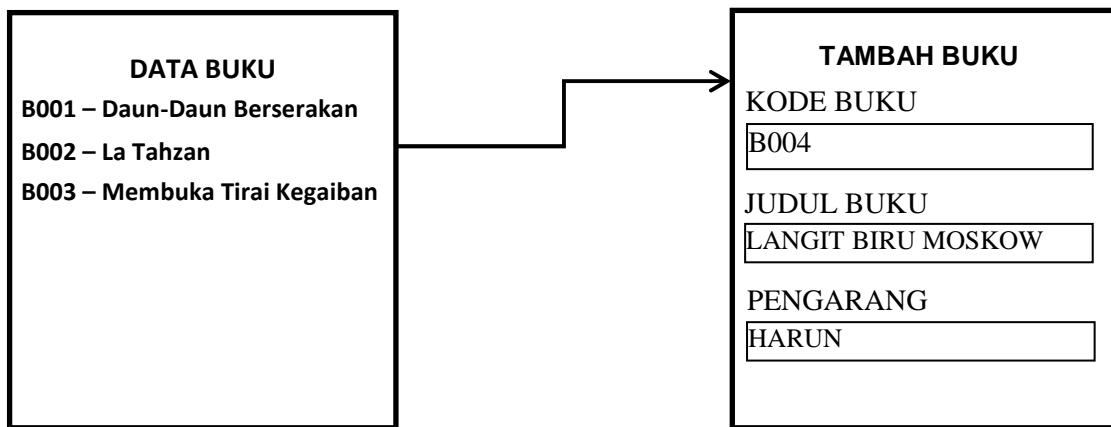
Pada Latihan Materi ini akan dibangun aplikasi yang memiliki fungsi untuk mengelola (menampilkan daftar, menambah) data buku di perpustakaan

Rancangan Database

NAMA DATABASE : PERPUSTAKAAN

NAMA TABEL : BUKU

kodebuku	judulbuku	pengarang
B001	Daun-Daun Berserakan	Fuad Bashori
B002	La Tahzan	Aidh AL Qarni
B003	Membuka Tirai Kegaiban	Jalaluddin Rakhmat



MATERI 9

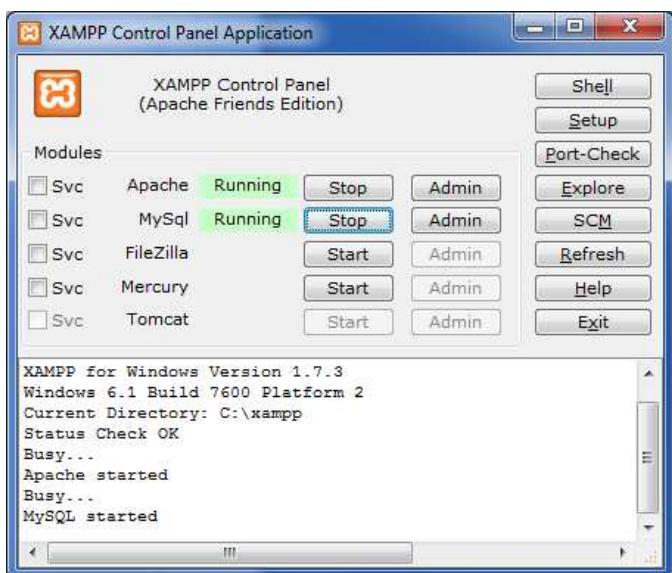
ANDROID, PHP DAN MYSQL

- Memahami Cara Kerja Android dengan PHP dan MySQL

1. Buat Project Baru sebagai berikut :

Project Name : Aplikasi9_NIMANDA
Build Target : Android 2.2
Application Name : Aplikasi8_NIMANDA
Package Name : emha.android.aplikasi9
Create Activity : MainActivity
Minimum SDK : 8

2. Jalankan XAMPP



Jalankan services Apache, dan MySQL

3. Dengan menggunakan PHPMYADMIN, buat database dan tabel baru sbb:

NAMA DATABASE : PERPUSTAKAAN

NAMA TABEL : BUKU

kodebuku	judulbuku	Pengarang
B001	Daun-Daun Berserakan	Fuad Bashori
B002	La Tahzan	Aidh AL Qarni
B003	Membuka Tirai Kegaiban	Jalaluddin Rakhmat

Server: localhost > Database: PERPUSTAKAAN > Tabel: BUKU

Field	kodebuku	judulbuku	pengarang
Jenis	CHAR	VARCHAR	VARCHAR
Ukuran Panjang/Nilai ¹	10	30	30
Default ²	None	None	None
Penyortiran			
Atribut			
Kosong	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Indeks	PRIMARY	---	---
AUTO_INCREMENT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Komentar			
Tipe MIME			
Transformasi Browser			
Pilihan transformasi ³			

Server: localhost > Database: PERPUSTAKAAN > Tabel: BUKU

✓ Table 'PERPUSTAKAAN'.'BUKU' has been created.

```
CREATE TABLE `PERPUSTAKAAN`.`BUKU` (
  `kodebuku` CHAR( 10 ) NOT NULL ,
  `judulbuku` VARCHAR( 30 ) NOT NULL ,
  `pengarang` VARCHAR( 30 ) NOT NULL ,
  PRIMARY KEY ( `kodebuku` )
) ENGINE = MYISAM ;
```

	Field	Jenis	Penyortiran	Atribut	Kosong	Default	Ekstra	Aksi
<input type="checkbox"/>	kodebuku	char(10)	latin1_swedish_ci		Tidak	None		
<input type="checkbox"/>	judulbuku	varchar(30)	latin1_swedish_ci		Tidak	None		
<input type="checkbox"/>	pengarang	varchar(30)	latin1_swedish_ci		Tidak	None		

↑ Pilih semua / Balik pilihan yang ditandai:

Server: localhost > Database: PERPUSTAKAAN > Tabel: BUKU

SQL Cari Sisipkan Ekspor Import Operasi Mengosongkan Hapus

Field	Jenis	Fungsi	Kosong	Nilai
kodebuku	char(10)			B001
judulbuku	varchar(30)			Daun-Daun Berserakan
pengarang	varchar(30)			Fuad Bashori

Abaikan

Field	Jenis	Fungsi	Kosong	Nilai
kodebuku	char(10)			B002
judulbuku	varchar(30)			Fuad Bashori
pengarang	varchar(30)			Aidh AL Qarni

Field	Jenis	Fungsi	Kosong	Nilai
kodebuku	char(10)			B003
judulbuku	varchar(30)			Membuka Tirai Kegaiban
pengarang	varchar(30)			Jalaluddin Rakhmat

+ Options

	kodebuku	judulbuku	pengarang
<input type="checkbox"/>	B001	Daun-Daun Berserakan	Fuad Bashori
<input type="checkbox"/>	B002	Fuad Bashori	Aidh AL Qarni
<input type="checkbox"/>	B003	Membuka Tirai Kegaiban	Jalaluddin Rakhmat

↑ Pilih semua / Balik pilihan yang ditandai:

4. Buat file PHP dengan nama **tampil.php**

```
<?php

mysql_connect("localhost", "root", "") ;
mysql_select_db("PERPUSTAKAAN") ;

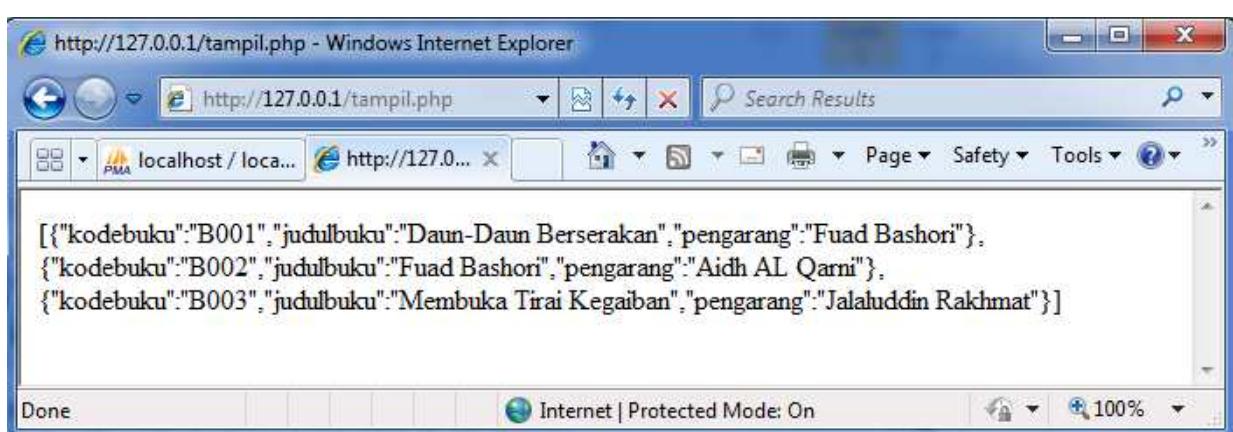
$q = mysql_query("SELECT * FROM buku") ;

while($r = mysql_fetch_assoc($q))
    $output[] = $r;

print(json_encode($output)) ;

mysql_close();

?>
```



5. Lengkapi kode program di MainActivity menjadi sbb:

```
1. package emha.android.aplikasi9;
2.
3. import java.io.BufferedReader;
4. import java.io.InputStream;
5. import java.io.InputStreamReader;
6. import org.apache.http.HttpEntity;
7. import org.apache.http.HttpResponse;
8. import org.apache.http.client.HttpClient;
9. import org.apache.http.client.methods.HttpPost;
10. import org.apache.http.impl.client.DefaultHttpClient;
11. import org.json.JSONArray;
12. import org.json.JSONException;
13. import org.json.JSONObject;
```

```

14. import android.app.ListActivity;
15. import android.net.ParseException;
16. import android.os.Bundle;
17. import android.util.Log;
18. import android.view.View;
19. import android.widget.AdapterView;
20. import android.widget.AdapterView.OnItemClickListener;
21. import android.widget.ArrayAdapter;
22. import android.widget.ListView;
23. import android.widget.Toast;
24.
25. public class MainActivity extends ListActivity {
26.     String kodebuku;
27.     String[] judulbuku = null;
28.
29.     /** Called when the activity is first created. */
30.     @Override
31.     public void onCreate(Bundle savedInstanceState) {
32.         super.onCreate(savedInstanceState);
33.
34.         String hasil = null;
35.         InputStream is = null;
36.         StringBuilder sb = null;
37.
38.         // http post
39.         try {
40.             HttpClient httpclient = new DefaultHttpClient();
41.             HttpPost httppost = new
42.                 HttpPost("http://10.0.2.2:80/tampil.php");
43.             HttpResponse response = httpclient.execute(httppost);
44.             HttpEntity entity = response.getEntity();
45.             is = entity.getContent();
46.         } catch (Exception e) {
47.             Log.e("log_tag", "Error in http connection" + e.toString());
48.         }
49.
50.         // convert response to string
51.         try {
52.             BufferedReader reader = new BufferedReader(new
53.                 InputStreamReader(is, "iso-8859-1"), 8);
54.             sb = new StringBuilder();
55.             sb.append(reader.readLine() + "\n");
56.             String line = "0";
57.             while ((line = reader.readLine()) != null) {
58.                 sb.append(line + "\n");
59.             }
60.             is.close();
61.             hasil = sb.toString();
62.         } catch (Exception e) {
63.             Log.e("log_tag", "Error converting result " + e.toString());
64.
65.             // parsing data
66.             JSONArray jArray;
67.             try {
68.                 jArray = new JSONArray(hasil);
69.                 JSONObject json_data = null;
70.                 judulbuku = new String[jArray.length()];
71.                 for (int i = 0; i < jArray.length(); i++) {
72.                     json_data = jArray.getJSONObject(i);
73.                     kodebuku = json_data.getString("kodebuku");
74.                     judulbuku[i] = json_data.getString("judulbuku");
75.                 }
76.             } catch (JSONException e1) {
77.                 Toast.makeText(getApplicationContext(), "No Buku Found",
78.                     Toast.LENGTH_LONG).show();
79.             } catch (ParseException e1) {
80.                 e1.printStackTrace();
81.             }
82.         }
83.     }
84. }

```

```

81.         //tampilkan
82.         setListAdapter(new ArrayAdapter<String>(this,
83.                         android.R.layout.simple_list_item_1, judulbuku));
84.         ListView lv;
85.         lv = getListView();
86.         lv.setTextFilterEnabled(true);
87.         lv.setOnItemClickListener(new OnItemClickListener() {
88.
89.             public void onItemClick(AdapterView<?> arg0, View arg1, int arg2,
90.                                     long arg3) {
91.                 // TODO Auto-generated method stub
92.                 Toast.makeText(getApplicationContext(),
93.                     judulbuku[arg2] + " wasClicked",
94.                     Toast.LENGTH_SHORT).show();
95.             }
96.         });

```



LATIHAN

- Buat Tabel baru di database MySQL Perpustakaan:

NAMA TABEL : BUKU

KodePetugas	Nama
P01	Surya
P02	Hari
P03	Yuni
P04	Dea

- Tampilan data Petugas tsb di Android