

# MODUL 14



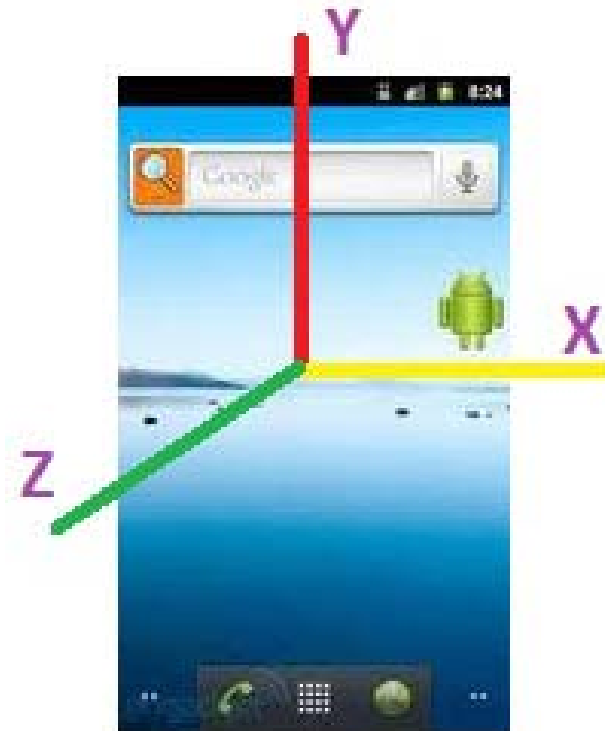
## FITUR ANDROID INTERKONEKSI CLIENT SERVER

DISADUR DARI MODUL TRAINING SEAMEO SEAMOLEC  
MODIFIED BY : MUHAMMAD ZEN S. HADI  
ZENHADI@EEPIS-ITS.EDU

# Accelerometer dan Compass



- Mendeteksi kemana *device* android menghadap
- Azimuth – Inklinasi.



# Parameter Pendeteksian Arah



➤ Menggunakan :

a) Objek `SensorManager`

b) Method `getSystemService()`

c) *Register* listener dari `SensorManager`

d) Method `onSensorChanged()`

# Global Positioning System



Pengaksesan lokasi pada android



# com.google.android.maps

Berikut adalah *class-class* penting yang ada pada *package* com.google.android.maps :

<b><i>Class</i></b>	<b>Fungsi</b>
<b>MapActivity</b>	Menampilkan Google MAPs, menangani <i>low-level networking</i>
<b>MapView</b>	Menampilkan map. Harus dimuat oleh MapActivity
<b>Map Controller</b>	Menggerakkan map di layar
<b>Overlay</b>	Drawable objek yang bisa ditampilkan di layar
<b>GeoPoint</b>	Posisi garis lintang/bujur (latitude/longitude)

# Telepon



- Memberikan pengaksesan mendasar
- Tidak memungkinkan untuk menciptakan sendiri aktivitas pemanggilan yang ditampilkan layar

# Membuat Panggilan Telepon



- `Intent.ACTION_CALL` :  
secara otomatis memulai panggilan,  
menampilkan aplikasi yang di-panggil.
- `Intent.ACTION_DIAL` :  
Untuk memanggil nomor tersebut dengan  
cepat, (dengan *dialer default* meminta *user*  
untuk secara eksplisit melakukan panggilan).

# Parameter Panggilan Masuk



- `TelephonManager.CALL_STATE_IDLE`  
Bila telepon tidak berdering atau dalam panggilan
- `TelephonManager.CALL_STATE_RINGING`  
Bila telepon berdering
- `TelephonManager.CALL_STATE_OFFHOOK`  
Jika ponsel saat ini dalam panggilan





# Bluetooth



Dalam penggunaan Bluetooth :

*a) Discovery*

*b) Bonding*



# Parameter Bluetooth



Pada layanan Bluetooth :

- a) Kelas BluetoothDevice
- b) Method `getSystemService()`
- c) Konstanta `Context.BLUETOOTH`

# Wi-Fi



Mengelola jaringan dan sambuangan Wi-Fi

- a) Koneksi jaringan
- b) konfigurasi
- c) Mengontrol jaringan radio



# Memonitoring Konektivitas (1)



Memberitahukan perubahan konektivitas :

- `ConnectivityManager.EXTRA_IS_FAILOVER`
- `ConnectivityManager.EXTRA_NO_CONNECTIVITY`
- `ConnectivityManager.EXTRA_REASON`
- `ConnectivityManager.EXTRA_NETWORK_INFO`
- `ConnectivityManager.EXTRA_OTHER_NETWORK_INFO`
- `ConnectivityManager.EXTRA_EXTRA_INFO`

# Memonitoring Konektivitas (2)



Memberitahukan perubahan konektivitas jaringan Wi-Fi:

- `WifiManager.WIFI_STATE_CHANGED_ACTION`
- `WifiManager.SUPPLICANT_CONNECTION_CHANGE_ACTION`
- `WifiManager.NETWORK_STATE_CHANGED_ACTION`
- `WifiManager.RSSI_CHANGED_ACTION`

# Fields public konfigurasi Wi-Fi



konfigurasi jaringan disimpan sebagai objek WifiConfigurasi :

- BSSID
- SSID
- NetworkId
- Prioritas
- Status koneksi

# Multi Media



Kelas-kelas yang menangani multi media:



- MediaPlayer
- MediaRecorder

# Camera

*Fitur camera pada android :*

a) Pengaturan *camera* :  
`getParameter()`

a) *Priview camera* :

- `setPriviewDisplay()`
- `setPriviewCallback()`
- `onPriviewframe()`

b) Mengambil gambar :  
`takePicture()`

c) Menyimpan dan membuka *file* :  
`openfileInput()`  
`openfileOutput()`



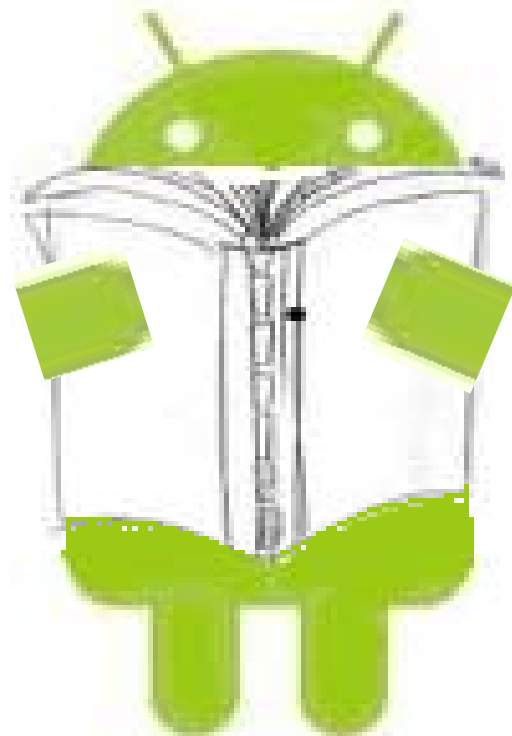


# Data Base



Menggunakan relasi penuh dengan SQLite :

- *Cursor* dan *Content values*
- *Query database*
- Menambah
- Merubah
- Menghapus



# S M S



SMS pada android menggunakan SMS Manager :



- *Mengirim pesan SMS*  
`smsManager ( )`
- *Mengirim pesan text*  
`sendTextMessage ( )`
- *Mengirim pesan data*  
`sendDataMessage ( )`
- *Mendengarkan pesan SMS*

# Konfirmasi dan Pelacakan SMS



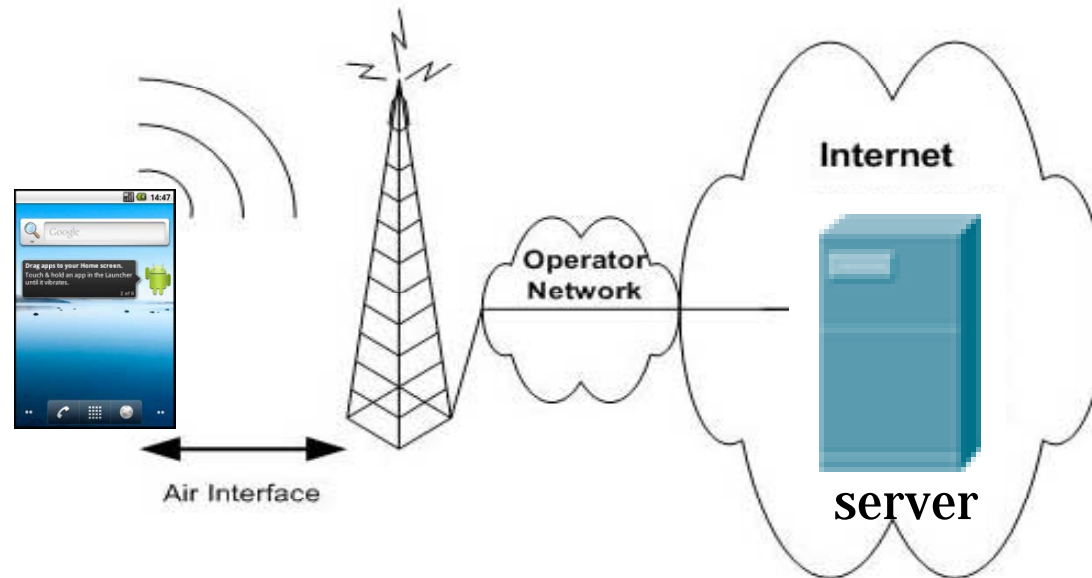
Kode-kode bagi penerima *Broadcast* :

- `Activity.RESULT_OK`  
Untuk menunjukkan transmisi sukses
- `SmsManager.RESULT_ERROR_GENERIC_FAILURE`  
Untuk menunjukkan kegagalan spesifik.
- `SmsManager.RESULT_ERROR_RADIO_OFF`  
Ketika sambungan radio akan dinonaktifkan.
- `SmsManager.RESULT_ERROR_NULL_PDU`  
Untuk menunjukkan kegagalan PDU.



# **INTERKONEKSI CLIENT-SERVER**

# Arsitektur Jaringan dgn Android

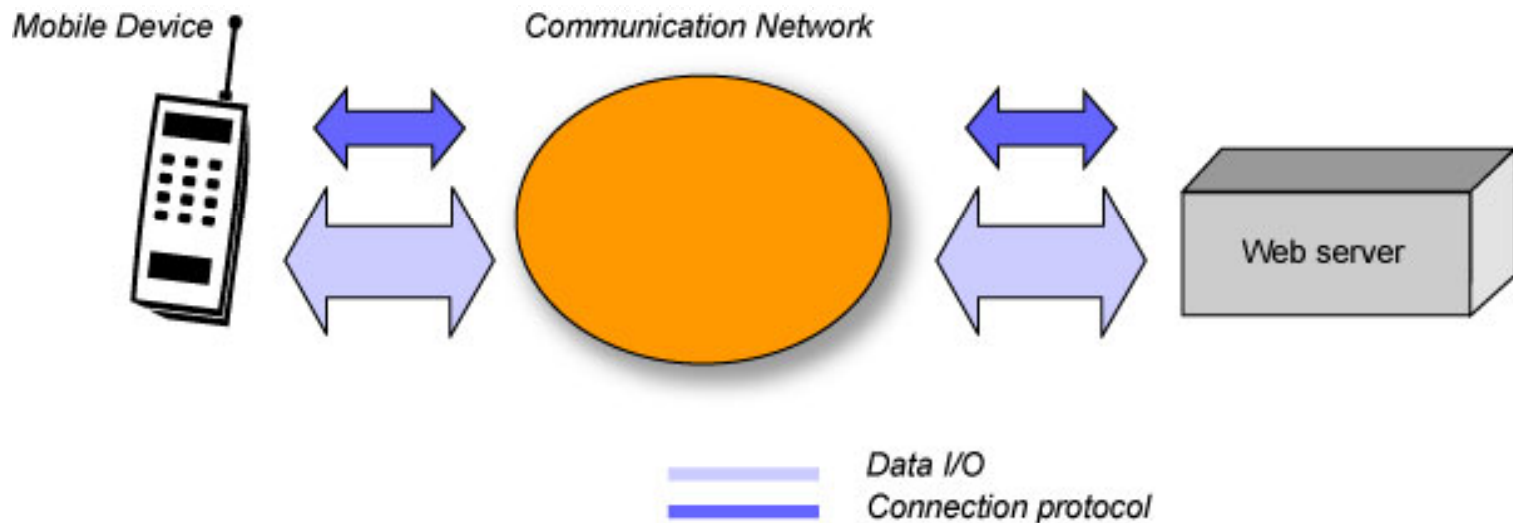


- Digunakan untuk akses internet, mengirim email, atau menampilkan isi suatu situs berita lewat RSS.

# HTTP networking



- Komunikasi antara perangkat mobile dan web server berbasis pada protokol HTTP (Hypertext Transfer Protocol).
- HTTP adalah protokol connection-oriented request-response.

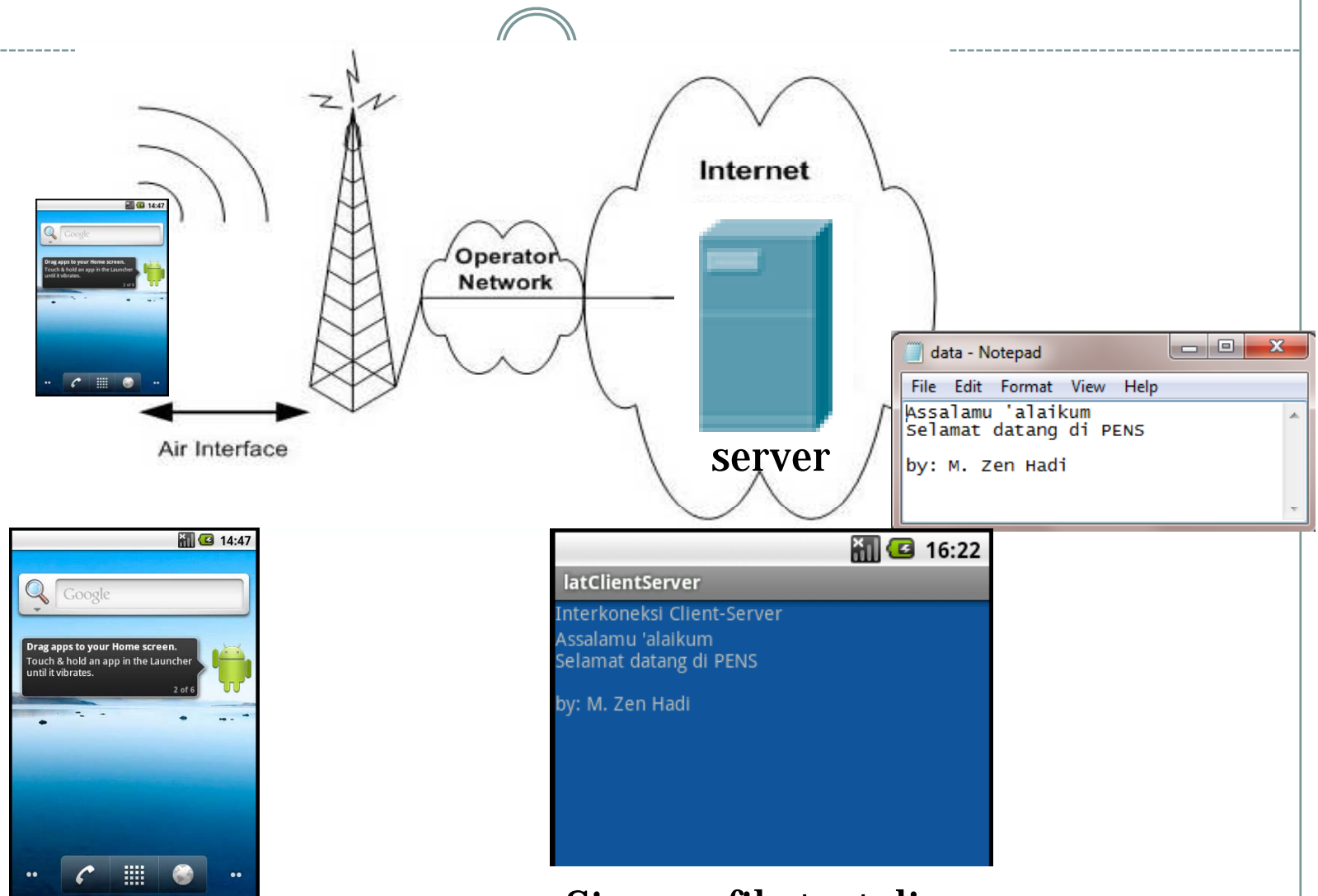


# Penanganan URL dalam Android



- Penanganan URL dalam Android meliputi open koneksi ke web server dari perangkat mobile dan penanganan data I/O diantara keduanya.
- Proses yang terjadi meliputi tahapan berikut :
  - Setup connection
  - Data transfer
  - Closed
- Android mendefinisikan `java.net.HttpURLConnection`, `java.net.URL` dan `java.net.URLConnection` class untuk membuat semua obyek koneksi.
- Dalam penanganan URL, `openConnection()` digunakan untuk membuka URL, yang akan memberikan obyek `HttpURLConnection`.
- Untuk transfer data menggunakan class `java.io.InputStreamReader` yang akan mengirimkan data tiap karakter dari sisi server.
- Untuk akses dari HP Android ke Server tidak bisa menggunakan `localhost / 127.0.0.1`, **harus menggunakan IP private atau publik.**

# Akses file text dari Server



Simpan file text di  
<http://192.168.41.1/data.txt>



# Akses file text dari Server



- File main.xml

```
1<?xml version="1.0" encoding="utf-8"?>
2<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3    android:orientation="vertical"
4    android:layout_width="fill_parent"
5    android:layout_height="fill_parent"
6    android:background="#125698"
7    >
8<TextView
9    android:layout_width="fill_parent"
10   android:layout_height="wrap_content"
11   android:text="Interkoneksi Client-Server"
12   />
13<TextView android:text=""
14   android:id="@+id/TextView01"
15   android:layout_width="wrap_content"
16   android:layout height="wrap content"></TextView>
17</LinearLayout>
```

TextView untuk menampung data dari server



# Akses file text dari Server

- File latClientServer.java



```
1 package pens.edu;
2
3 import java.io.IOException;
4 import java.io.InputStream;
5 import java.io.InputStreamReader;
6 import java.net.HttpURLConnection;
7 import java.net.URL;
8 import java.net.URLConnection;
9
10 import android.app.Activity;
11 import android.os.Bundle;
12 import android.widget.TextView;
13
14 public class latClientServer extends Activity {
15     TextView txt;
16     /** Called when the activity is first created. */
17     @Override
18     public void onCreate(Bundle savedInstanceState) {
19         super.onCreate(savedInstanceState);
20         setContentView(R.layout.main);
21         String str = downloadTeks ("http://192.168.41.1/data.txt");
22         txt = (TextView) findViewById(R.id.TextView01);
23         txt.setText(str);
24     }
```

Jangan gunakan localhost, ini akan memanggil metode **downloadTeks**



# Akses file text dari Server

- File latClientServer.java



Membuat metode OpenHttpConnection untuk membuka koneksi ke Server menggunakan protokol HTTP

```
26 private InputStream OpenHttpConnection (String urlString) throws IOException {
27     InputStream in = null;
28     int response = -1;
29     URL url = new URL(urlString);
30     URLConnection conn = url.openConnection();
31     try {
32         HttpURLConnection httpConn = (HttpURLConnection) conn;
33         httpConn.setAllowUserInteraction(false);
34         httpConn.setInstanceFollowRedirects(true);
35         httpConn.setRequestMethod("GET"); //menggunakan metode GET saja
36         httpConn.connect();
37
38         response = httpConn.getResponseCode();
39         if (response == HttpURLConnection.HTTP_OK) {
40             in = httpConn.getInputStream();
41         }
42     } catch (Exception ex) {
43         throw new IOException ("Error connecting");
44     }
45     return in;
46 }
```

# Akses file text dari Server

- File latClientServer.java



Membuat metode `downloadTeks` yang akan dipanggil dari program utama

```
private String downloadTeks (String URL) {
    InputStream in = null;
    try {
        in = OpenHttpConnection (URL);
    } catch (IOException e){}
    int charRead;
    StringBuffer data = new StringBuffer();
    String str = "";
    try {
        while ((charRead = in.read()) != -1 ){
            data.append((char) charRead);
        }
        str = data.toString();
        in.close();
    } catch (IOException e) {}
    return str;
} //akhir downloadTeks
} // akhir latClientServer
```

Membuka koneksi HTTP

Transfer Data

Menutup koneksi

# Akses file text dari Server

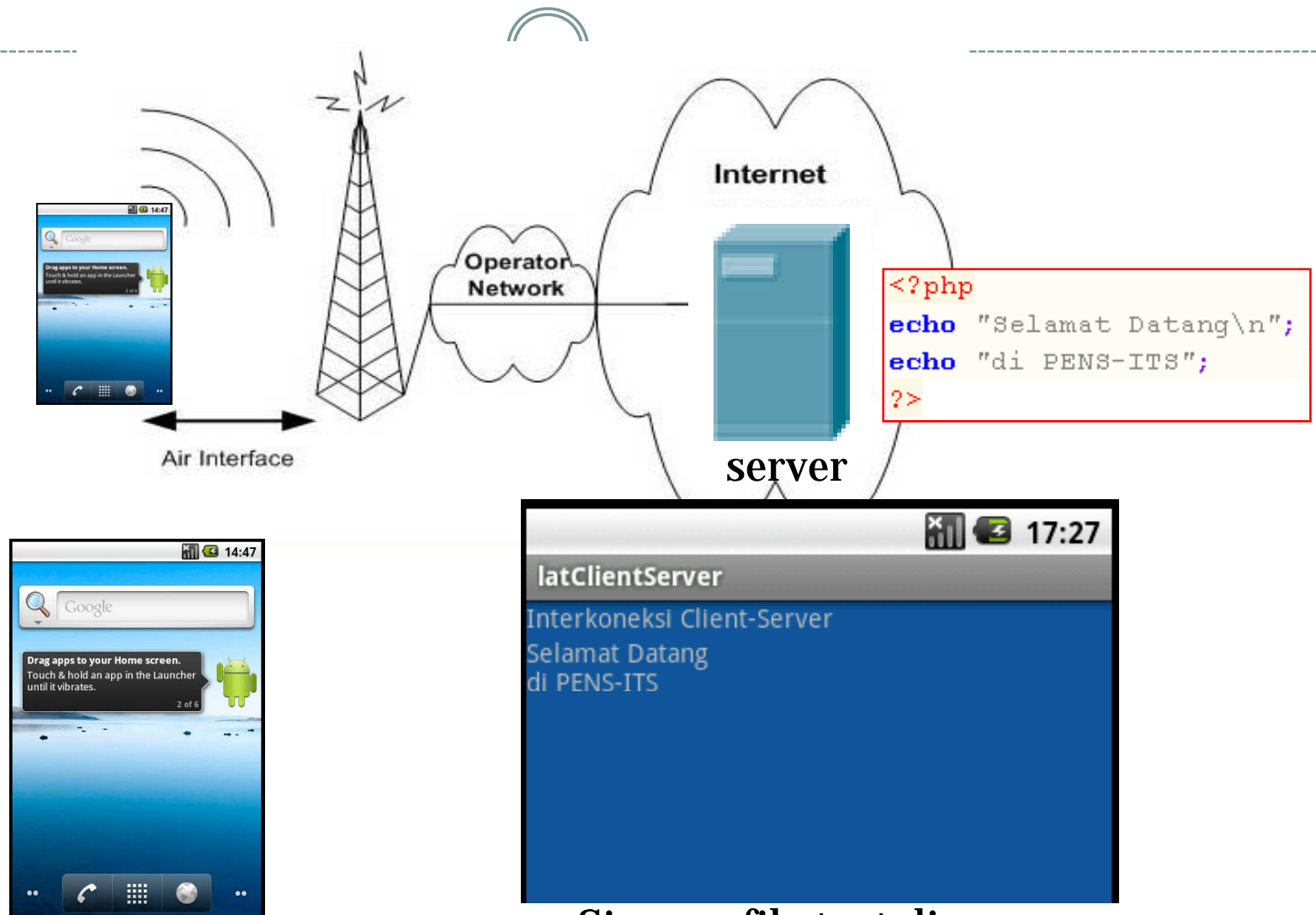
- File AndroidManifest.xml



Menambahkan permission untuk bisa mengakses Internet

```
1<?xml version="1.0" encoding="utf-8"?>
2<manifest xmlns:android="http://schemas.android.com/apk/res/android"
3    package="pens.edu"
4    android:versionCode="1"
5    android:versionName="1.0">
6
7    <application android:icon="@drawable/icon" android:label="@string/app_name">
8        <activity android:name=".latClientServer"
9            android:label="@string/app_name">
10            <intent-filter>
11                <action android:name="android.intent.action.MAIN" />
12                <category android:name="android.intent.category.LAUNCHER" />
13            </intent-filter>
14        </activity>
15
16    </application>
17    <uses-sdk android:minSdkVersion="8" />
18    <uses-permission android:name="android.permission.INTERNET"/>
19</manifest>
```

# Akses file php dari Server



Simpan file text di  
<http://192.168.41.1/data.php>

# Akses file php dari Server

- File latClientServer.java



Semua design dan coding sama dengan akses file text, hanya cuma diganti dengan file php di program utama metode onCreate

```
14 public class latClientServer extends Activity {
15     TextView txt;
16     /** Called when the activity is first created. */
17     @Override
18     public void onCreate(Bundle savedInstanceState) {
19         super.onCreate(savedInstanceState);
20         setContentView(R.layout.main);
21         String str = downloadTeks ("http://192.168.41.1/data.php");
22         txt = (TextView) findViewById(R.id.TextView01);
23         txt.setText(str);
24     }
```

# Passing Parameters ke Server

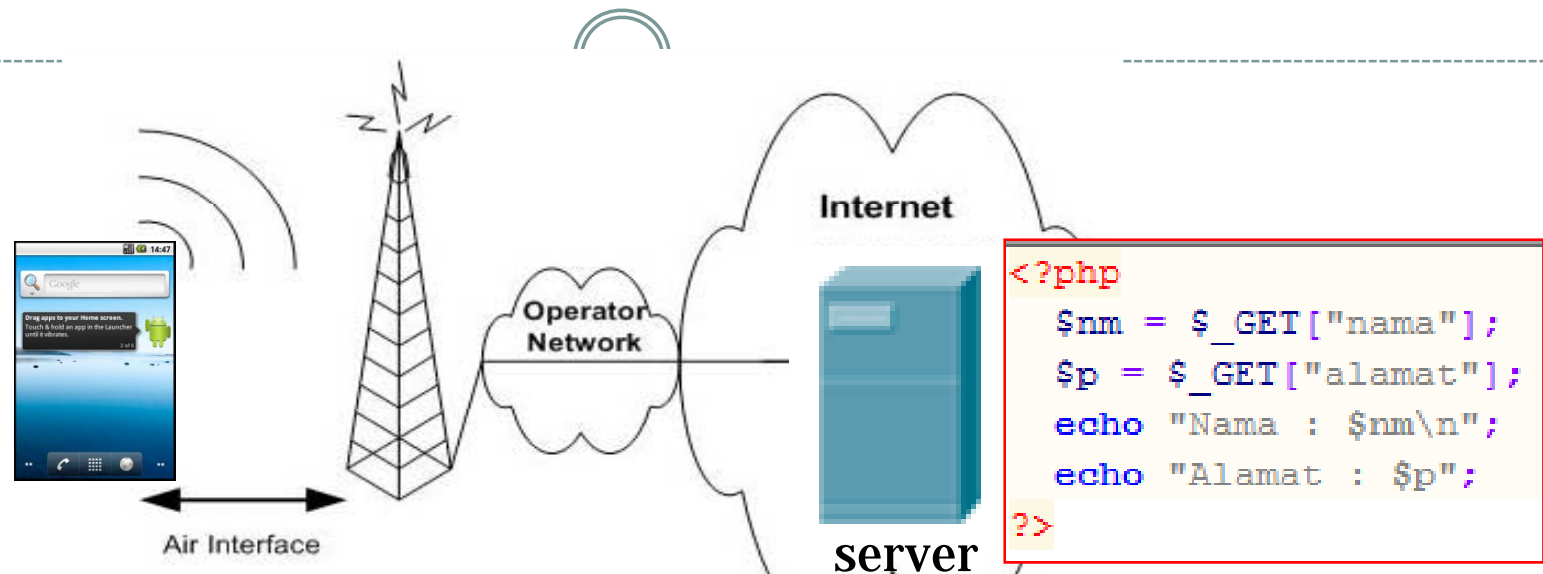


- 2 teknik yang digunakan untuk mengirim data ke server: metode **GET** atau **POST** request
- Metode GET data akan ditambahkan pada URL di server.
- Metode POST, masing-masing nilai dituliskan ke output stream.
- Data dikirim ke server harus dalam pasangan set nilai:
  - field name
  - Nilai yang berhubungan dgn field tsb.
- Field name dan nilai harus dipisahkan oleh tanda sama dengan (=)
- Pasangan nilai dipisahkan dari URL dengan tanda tanya (?)
- Masing-masing nilai dipisahkan dari nilai yang lainnya dengan ampersang (&).
- Karakter spasi diubah dengan tanda plus (+)
- Contoh: request berikut mengirim 2 parameter: pOne="one bit" dan pTwo="two"

`http://192.168.41.1/simple.php?p=one+bit&pTwo=two`



# Akses file php dengan parameter



latClientServer  
Interkoneksi Client-Server  
Username   
Alamat   
Proses

latClientServer  
Interkoneksi Client-Server  
Username Faruq  
Alamat Kediri  
Proses  
Nama : Faruq  
Alamat : Kediri

Simpan file text di  
<http://192.168.41.1/proses.php>

# Akses file php dari Server

- File latClientServer.java

Semua design dan coding sama dengan akses file text, hanya cuma diganti dengan file php di program utama metode onCreate

```
17 public class latClientServer extends Activity {
18     TextView txt;
19     EditText nm, almt;
20     Button proses;
21     String nama2, alamat2;
22     /** Called when the activity is first created. */
23     @Override
24     public void onCreate(Bundle savedInstanceState) {
25         super.onCreate(savedInstanceState);
26         setContentView(R.layout.main);
27         txt = (TextView) findViewById(R.id.TextView03);
28         nm = (EditText) findViewById(R.id.EditText01);
29         almt = (EditText) findViewById(R.id.EditText02);
30         proses = (Button) findViewById(R.id.Button01);
31         proses.setOnClickListener(new klikproses());
32     }
33
34     class klikproses implements Button.OnClickListener {
35     public void onClick (View v) {
36         nama2 = nm.getText().toString();
37         alamat2 = almt.getText().toString();
38         String urlSite = "http://192.168.41.1/proses.php?nama="+nama2+"&alamat="+alamat2;
39         String str = downloadTeks (urlSite);
40         txt.setText(str);
41     }
42     }
43 }
```

# Interaksi dengan Database (Review)



- Dalam pembuatan aplikasi berbasis Database – Android, beberapa perintah di Database baik DML dan DDL bisa digunakan dikarenakan aksesnya masih berbasis PHP.
- Beberapa tahapan dalam interkoneksi dengan Database :
  - ❑ Membuka koneksi ke server MySQL  
`$conn = mysql_connect ("host", "username", "password");`
  - ❑ Memilih database  
`mysql_select_db("namadatabase", $conn);`
  - ❑ Memilih tabel dengan query  
`$hasil = mysql_query("SQL Statement");`
  - ❑ Mengambil record dari tabel  
`$row = mysql_fetch_row($hasil);`

# Contoh aplikasi berbasis Android – Database Server



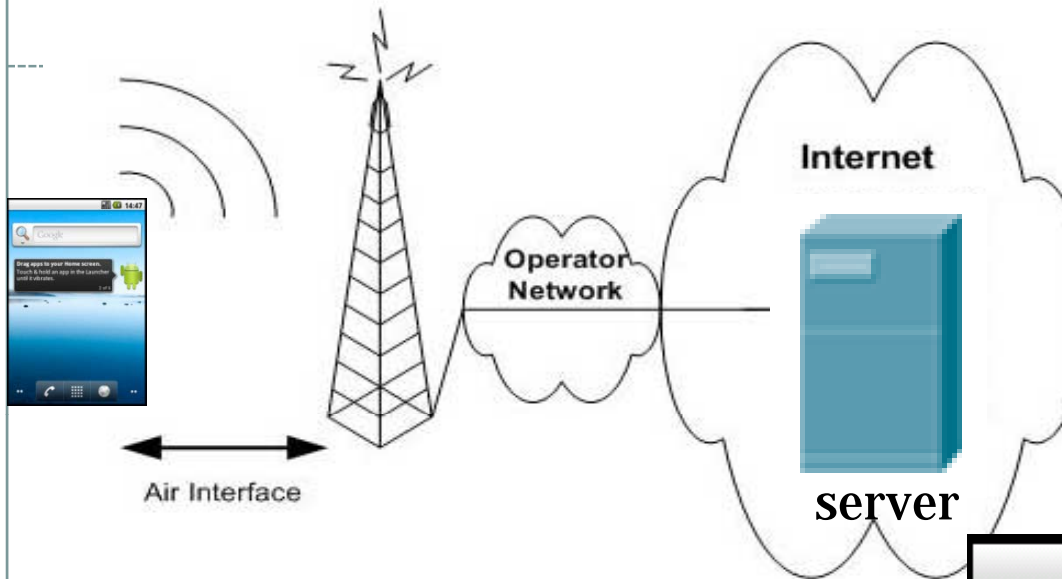
- Buat database KAMPUS, dengan tabel Mahasiswa yang memiliki field sebagai berikut, dengan NRP sebagai Primary Key :

	Field	Type	Collation	Attributes	Null
<input type="checkbox"/>	<u>NRP</u>	int(10)			No
<input type="checkbox"/>	Nama	varchar(20)	latin1_general_ci		No
<input type="checkbox"/>	Alamat	varchar(20)	latin1_general_ci		No

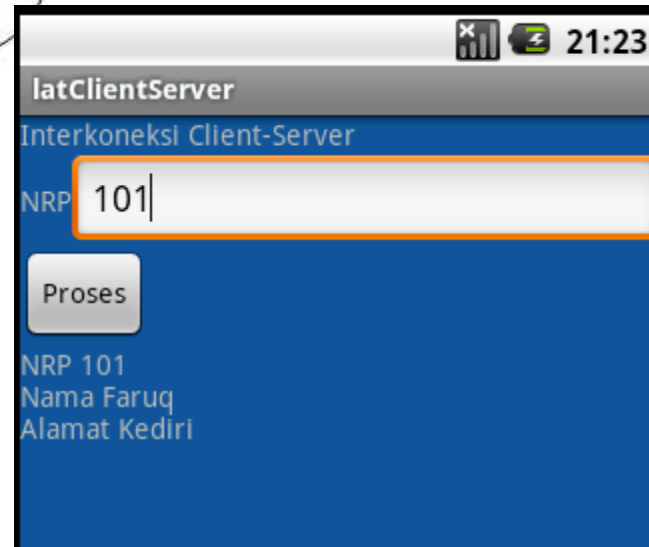
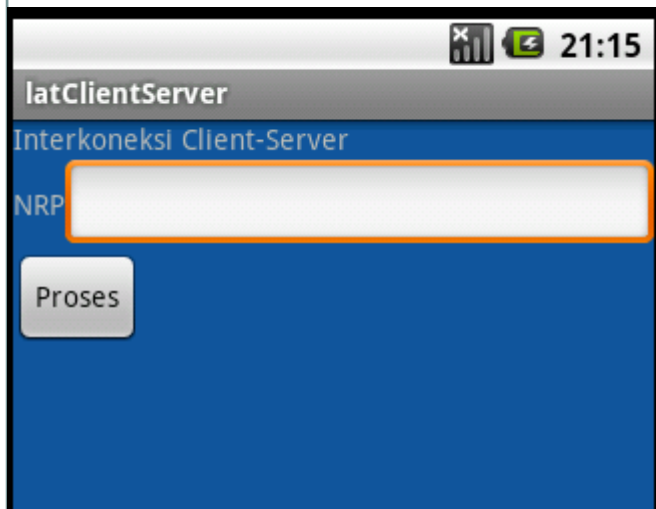
- Akan dibuat aplikasi untuk menampilkan data dengan Android.
- Isikan database, beberapa data berikut untuk diakses :

			<u>NRP</u>	<u>Nama</u>	<u>Alamat</u>
<input type="checkbox"/>			101	Faruq	Kediri
<input type="checkbox"/>			102	Fakhri	Surabaya

# Tampil Data dari Database



```
<?php
$dtmrp = $_GET["nrp"];
$conn=mysql_connect ("localhost","root","")
    or die ("koneksi gagal");
mysql_select_db("kampus",$conn);
$hasil = mysql_query("select * from mahasiswa
    where nrp='$dtmrp'", $conn);
while ($row=mysql_fetch_row($hasil)) {
    echo "NRP " . $row[0];
    echo "\n";
    echo "Nama " . $row[1];
    echo "\n";
    echo "Alamat " . $row[2];
    echo "\n";
}
?>
```



Simpan file text di

<http://192.168.41.1/proses.php>

# Aplikasi Tampil Data dari Database

```
public class latClientServer extends Activity {
    TextView txt;
    EditText nrp;
    Button proses;
    String dtNrp;
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        txt = (TextView) findViewById(R.id.TextView03);
        nrp = (EditText) findViewById (R.id.EditText01);
        proses = (Button) findViewById (R.id.Button01);
        proses.setOnClickListener(new klikproses());
    }
    class klikproses implements Button.OnClickListener {
        public void onClick (View v) {
            dtNrp = nrp.getText().toString();
            String urlSite = "http://192.168.41.1/proses.php?nrp="+dtNrp;
            String str = downloadTeks (urlSite);
            txt.setText(str);
        }
    }
}
```

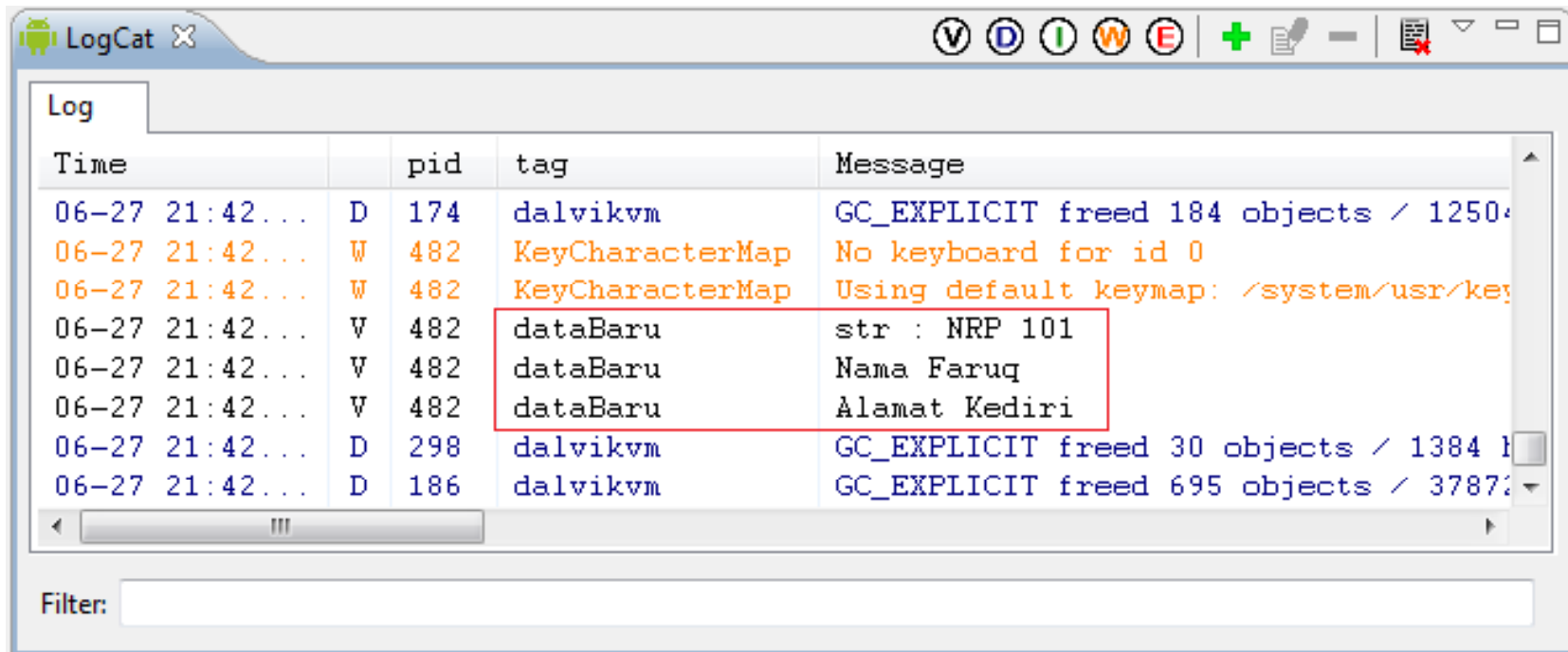
# Debugging di Android

- Untuk melakukan debugging di android, bisa dilakukan dengan menambahkan beberapa script coding ke dalam program.
- Berikut diberikan contoh debugging utk program akses ke database.

```
14 import android.widget.TextView;
15 import android.util.Log;
16
17 public class latClientServer extends Activity {
18     public static final String LOG_TAG = "dataBaru";
19     TextView txt;
20     EditText nrp;
21     Button proses;
22     String dtnrp;
23     /** Called when the activity is first created. */
24     @Override
25     public void onCreate(Bundle savedInstanceState) {
26         super.onCreate(savedInstanceState);
27         setContentView(R.layout.main);
28         txt = (TextView) findViewById(R.id.TextView03);
29         nrp = (EditText) findViewById (R.id.EditText01);
30         proses = (Button) findViewById (R.id.Button01);
31         proses.setOnClickListener(new klikproses());
32
33     }
34     class klikproses implements Button.OnClickListener {
35         public void onClick (View v) {
36             dtnrp = nrp.getText().toString();
37             String urlSite = "http://192.168.41.1/proses.php?nrp="+dtnrp;
38             String str = downloadTeks (urlSite);
39             txt.setText(str);
40             Log.v(LOG_TAG, "str : " +str);
41         }
42     }
```

# Debugging di Android

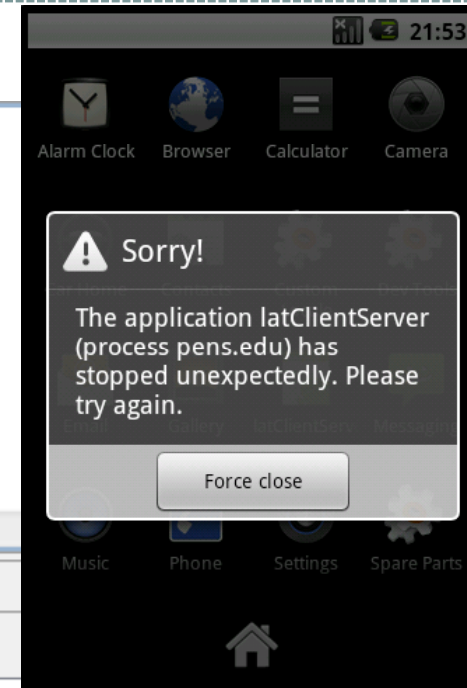
- Kemudian Run Program tersebut.
- Buka : Window | Open Perspective | Debug





# Debugging di Android

- Bisa juga untuk mengetahui problem yang terjadi seperti ini.



```
70     in = OpenHttpConnection (URL);
71     } catch (IOException e){}
72     int charRead;
73     StringBuffer data = new StringBuffer();
74     String str = "";
75     try {
76         while ((charRead = in.read()) != -1 ){
77             data.append((char) charRead);
78         }
79         str = data.toString();
80         in.close();
```

LogCat

pid	tag	Message
542	AndroidRuntime	java.lang.NullPointerException
542	AndroidRuntime	at pens.edu.latClientServer.downloadTeks(latClientServer.java:76)
542	AndroidRuntime	at pens.edu.latClientServer.access\$0(latClientServer.java:67)
542	AndroidRuntime	at pens.edu.latClientServer\$klikproses.onClick(latClientServer.java:38)
542	AndroidRuntime	at android.view.View.performClick(View.java:2408)
542	AndroidRuntime	at android.view.View\$PerformClick.run(View.java:8816)
542	AndroidRuntime	at android.os.Handler.handleCallback(Handler.java:587)
542	AndroidRuntime	at android.os.Handler.dispatchMessage(Handler.java:92)