

MODUL 14

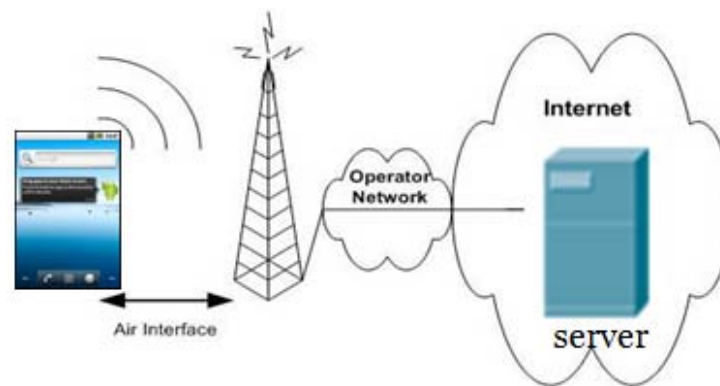
INTERKONEKSI CLIENT-SERVER ANDROID

A. Tujuan :

1. Memahami interkoneksi client-server dengan php di android
2. Memahami interkoneksi client-server dengan php-mysql
3. Memahami passing parameter dalam koneksi client-server

B. Dasar Teori

Interkoneksi client-server pada Android digunakan untuk akses internet, mengirim email, atau menampilkan isi suatu situs berita lewat RSS.



Gambar 1. Interkoneksi client-server dengan 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.

C. Tugas Pendahuluan

Buatlah desain flowchart untuk setiap soal dalam percobaan

D. Percobaan

D.1. Latihan

Latihan 1: Membuat aplikasi client-server

a. Design tampilan di : `res/layout/main.xml`

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:background="#125698"
    >
    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="Interkoneksi Client-Server"
        />
    <LinearLayout
        android:orientation="horizontal"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        >
        <TextView android:text="Username"
            android:id="@+id/TextView01"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"></TextView>
        <EditText android:id="@+id/EditText01"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"></EditText>
    </LinearLayout>
    <LinearLayout
        android:orientation="horizontal"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        >
        <TextView android:id="@+id/TextView02"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Alamat"></TextView>
        <EditText android:id="@+id/EditText02"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"></EditText>
    </LinearLayout>
    <Button android:id="@+id/Button01"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Proses"></Button>
    <TextView android:id="@+id/TextView03"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"></TextView>
</LinearLayout>
```

b. Buat file : latClientServer.java

```

package pens.edu;

import java.io.IOException;
import java.io.InputStream;
import java.net.HttpURLConnection;
import java.net.URL;
import java.net.URLConnection;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.util.Log;

public class latClientServer extends Activity {
    public static final String LOG_TAG = "dataBaru";
    TextView txt;
    EditText nm, almt;
    Button proses;
    String nama2, alamat2;
    /** 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);
        nm = (EditText) findViewById(R.id.EditText01);
        almt = (EditText) findViewById(R.id.EditText02);
        proses = (Button) findViewById(R.id.Button01);
        proses.setOnClickListener(new klikproses());
    }
    class klikproses implements Button.OnClickListener {
        public void onClick (View v) {
            nama2 = nm.getText().toString();
            alamat2 = almt.getText().toString();
            String urlSite =
                "http://192.168.41.1/proses.php?nama="+nama2+"&alamat="
                "+alamat2;
            String str = downloadTeks (urlSite);
            txt.setText(str);
            Log.v(LOG_TAG, "str : " +str);
        }
    }

    private InputStream OpenHttpConnection (String urlString) throws IOException {
        InputStream in = null;
        int response = -1;
        URL url = new URL(urlString);
        URLConnection conn = url.openConnection();
        try {
            HttpURLConnection httpConn = (HttpURLConnection) conn;
            httpConn.setAllowUserInteraction(false);
            httpConn.setInstanceFollowRedirects(true);
            httpConn.setRequestMethod("GET"); //menggunakan metode GET saja
            httpConn.connect();

            response = httpConn.getResponseCode();
            if (response == HttpURLConnection.HTTP_OK) {
                in = httpConn.getInputStream();
            }
        } catch (Exception ex) {
            throw new IOException ("Error connecting");
        }
        return in;
    }
}

```

```

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

```

c. Edit file AndroidManifest.xml agar bisa koneksi ke internet

```

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

    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <activity android:name=".latClientServer"
            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>
    <uses-sdk android:minSdkVersion="8" />
    <uses-permission android:name="android.permission.INTERNET" />
</manifest>

```

d. Buat file : proses.php yang ditaruh di server untuk memproses data dari HP Android

```

<?
$dtnama = $_GET["nama"];
$dttalamat = $_GET["alamat"];
echo "Nama : $dtnama";
echo "\n";
echo "Alamat : $dttalamat";
?>

```

D.2. Permasalahan

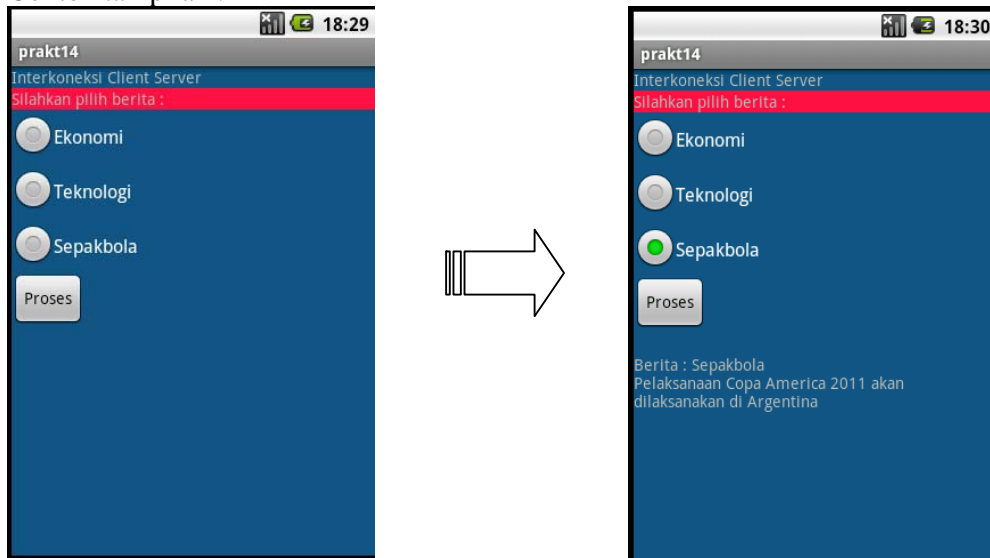
Untuk permasalahan dalam praktikum, lakukan langkah-langkah berikut :

- a. Buatlah tampilan seperti dibawah ini untuk koneksi ke server.
- b. Buatlah database di sisi server

| berita | info |
|-----------|---|
| Ekonomi | Tingkat pertumbuhan perekonomian Indonesia di tahu... |
| Teknologi | Android diprediksi akan menguasai pasar gadget |
| Sepakbola | Pelaksanaan Copa America 2011 akan dilaksanakan di... |

- c. Buat file php untuk memproses koneksi dari HP Android ke Database server
- d. Beberapa file yang dibutuhkan :
 - Design tampilan di : res/layout/main.xml
 - Coding di file java
 - Edit file AndroidManifest.xml agar bisa koneksi ke internet

Contoh tampilan :



E. Laporan Resmi

Analisa dari program yang telah anda buat diatas.