

Mata Kuliah	:	IF330324-IF430324 Pembangunan Aplikasi Mobile
Semester	:	2 (Genap)
Topik	:	User interface ( <i>Notification, Listview</i> )
Tanggal Sesi	:	22 Maret 2016
Minggu ke-/sesi	:	7/3
Waktu pengerjaan	:	2x50 menit
Aktifitas	:	Mahasiswa menggunakan <i>notification</i> dan <i>listview</i> pada Android
Setoran	:	Folder hasil pengerjaan bagian DELIVERABLES pada petunjuk praktikum yang telah dikompres menjadi file dengan tipe .zip/.rar.
Batas penyerahan	:	23 Maret 2016, sebelum pukul 21.00 WIB
Tujuan praktikum	:	Mahasiswa mampu memahami dan menggunakan <i>notification</i> dan <i>listview</i> pada Android
Sarana	:	Android Studio
Jenis kegiatan	:	Praktikum

Mulai minggu ke-3 sesi 2, mahasiswa akan melakukan praktikum PAM yang dibagi ke dalam dua bagian, yaitu:

1. Mengikuti langkah-langkah praktikum untuk mengulas atau memperkuat kemampuan teknis mahasiswa yang didasarkan pada materi yang diajarkan pada kelas teori atau materi yang menjadi pelengkap atau tambahan dari materi yang telah disampaikan pada kelas teori.
2. Mengerjakan bagian *deliverables* pada dokumen praktikum untuk kemudian disetor via Moodle. Tugas yang diberikan tersebut harus dikerjakan dan disetor sesuai dengan waktu deadline yang diberikan. Perbedaan bagian ini dengan praktikum yang telah dilakukan sebelumnya adalah mahasiswa diharapkan akan menghasilkan sebuah aplikasi yang utuh yang akan dikumpulkan pada minggu ke-14 masa perkuliahan PAM. Deskripsi aplikasi yang akan dibangun dijelaskan pada bagian *deliverables*. Mahasiswa diharapkan akan membangun komponen atau bagian-bagian dari aplikasi tersebut secara bertahap setiap minggu, sesuai dengan topik praktikum yang sedang diberikan pada setiap sesi.

#### **BAGIAN I: LANGKAH-LANGKAH PRAKTIKUM**

Pada sesi teori, anda sudah mempelajari mengenai *notification* dan *listview* pada Android. *Notification* adalah pesan yang ditampilkan diluar UI dari aplikasi anda, notifikasi akan muncul pada *notification bar*. *Notification* biasanya dipasangkan dengan *Broadcast Receiver* dan *Service*. Contoh pada aplikasi BBM, *Broadcast Receiver* akan menerima pesan masuk, lalu *Broadcast Receiver* akan memanggil *Service*, lalu *Service* akan membuat *notification*. Kali ini anda akan membuat sebuah notifikasi sederhana yang di-trigger dengan sebuah *Button*. Selain itu, Anda juga akan membuat sebuah *listview* yang berisi daftar produk pada praktikum kali.

## Notification

- a. Buat sebuah **empty activity** beri nama **MainActivity** dan ubahlah MainActivity.java seperti contoh dibawah ini.

```
package com.cesario.notification;

import android.app.NotificationManager;
import android.app.PendingIntent;
import android.app.TaskStackBuilder;
import android.content.Context;
import android.content.Intent;
import android.os.Bundle;
import android.support.v4.app.NotificationCompat;
import android.support.v7.app.AppCompatActivity;
import android.view.View;

public class MainActivity extends AppCompatActivity {

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

    //Dijalankan ketika button notify ditekan
    public void notify(View v) {
        //Setting notification yang ingin ditampilkan
        NotificationCompat.Builder mBuilder =
            new NotificationCompat.Builder(this)
                .setSmallIcon(R.drawable.notif)
                .setContentTitle("My notification")
                .setContentText("Hello World!")
                .setAutoCancel(true) //menghilangkan notifikasi
ketika notifikasi di tekan
                .setVibrate(new long[]{1000, 1000, 1000, 1000,
1000});
        // Membuat intent yang dituju ketika notifikasi ditekan
        Intent resultIntent = new Intent(this, ResultActivity.class);

        // objek stack builder digunakan untuk memastikan ketika back
ditekan dari, aplikasi anda akan menuju homescreen
        TaskStackBuilder stackBuilder = TaskStackBuilder.create(this);
        stackBuilder.addParentStack(ResultActivity.class);
        stackBuilder.addNextIntent(resultIntent);
        PendingIntent resultPendingIntent =
            stackBuilder.getPendingIntent(
                0,
                PendingIntent.FLAG_UPDATE_CURRENT
            );

        mBuilder.setContentIntent(resultPendingIntent);
        NotificationManager mNotificationManager =
            (NotificationManager)
getSystemService(Context.NOTIFICATION_SERVICE);
        // 101 adalah ID notifikasi, digunakan jika kita ingin meng-
update notifikasi.
        mNotificationManager.notify(101, mBuilder.build());
    }
}
```

- b. Tambahkan 1 Button pada activity\_main.xml yang memanggil method notify pada MainActivity.java

- c. Buat sebuah **empty activity** beri nama **ResultActivity** , biarkan kosong atau silahkan isi sesuai keinginan anda
- d. Tambahkan *notification icon*, silahkan *copy file icon.png* ke *folder* `<PATH_TO_PROJECT>/app/src/main/res/drawable/`
- e. Ubah file *manifest* anda pada *tag activity* seperti contoh dibawah ini.

```
<activity android:name=".MainActivity">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
<activity
    android:name=".ResultActivity"
    android:parentActivityName=".MainActivity">

</activity>
```

- f. *Run project* anda.

### Listview

- a. Buat sebuah file .xml pada folder `res/layout/` bernama **item**. Dan ubah file tersebut seperti contoh dibawah ini

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

    <ImageView
        android:layout_width="50dp"
        android:layout_height="50dp"
        android:id="@+id/gambar"
        android:layout_alignParentTop="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Nama"
        android:id="@+id/nama"
        android:textSize="12dp"
        android:layout_centerVertical="true"
        android:layout_toRightOf="@+id/gambar"
        android:layout_toEndOf="@+id/gambar" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Rp. Harga"
        android:id="@+id/harga"
        android:textSize="12dp"
        android:layout_centerVertical="true"
        android:layout_alignParentRight="true"
        android:layout_alignParentEnd="true" />

</RelativeLayout>
```

- b. Buat sebuah class bernama Produk dan ubahlah `Produk.java` seperti contoh dibawah ini

```
public class Produk {  
    private String nama;  
    private int harga;  
  
    public Produk(String nama, int harga) {  
        this.nama = nama;  
        this.harga = harga;  
    }  
  
    public String getNama() {  
        return nama;  
    }  
  
    public void setNama(String nama) {  
        this.nama = nama;  
    }  
  
    public int getHarga() {  
        return harga;  
    }  
  
    public void setHarga(int harga) {  
        this.harga = harga;  
    }  
}
```

- c. Buat sebuah class adapter bernama `ProdukAdapter.java` dan ubahlah seperti berikut

```
public class ProdukAdapter extends ArrayAdapter<Produk> {
    Context context;
    ArrayList<Produk> produks;

    //View holder digunakan untuk menyimpan objek item yang sudah
    pernah dibuat
    static class ViewHolder {
        public TextView nama;
        public TextView harga;
        public ImageView gambar;
    }

    //konstuktur
    public ProdukAdapter(Context context, ArrayList<Produk> produks) {
        super(context, R.layout.item, produks);
        this.context = context;
        this.produks = produks;
    }

    @Override
    public View getView(int position, View convertView, ViewGroup
parent) {
        // Get the data item for this position
        Produk produk = getItem(position);

        // Cek data, jika item sudah pernah dibuat maka akan di
        digunakan ulang, jika tidak maka akan dibuat
        ViewHolder viewHolder;
        if (convertView == null) {
            //membuat baru item
            viewHolder = new ViewHolder();
            LayoutInflater inflater =
LayoutInflater.from(getContext());
            convertView = inflater.inflate(R.layout.item, parent,
false);
            viewHolder.nama = (TextView)
convertView.findViewById(R.id.nama);
            viewHolder.harga = (TextView)
convertView.findViewById(R.id.harga);
            viewHolder.gambar = (ImageView)
convertView.findViewById(R.id.gambar);

            convertView.setTag(viewHolder);
        } else {
            //menggunakan item yang sudah pernah dibuat
            viewHolder = (ViewHolder) convertView.getTag();
        }
        // Set item dengan value dari objek
        viewHolder.nama.setText(produk.getNama());
        viewHolder.harga.setText("Rp. " +
String.valueOf(produk.getHarga()));
        viewHolder.gambar.setImageResource(R.drawable.notif);

        return convertView;
    }
}
```

d. Ubah ResultActivty.java dan activity\_result.xml seperti berikut

```
public class ResultActivity extends AppCompatActivity {
    ArrayList<Produk> produks;
    ProdukAdapter adapter;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_result);

        //array penampung data produk yang ingin ditampilkan
        produks = new ArrayList<>();

        //listview
        ListView listView = (ListView) findViewById(R.id.listProduk);

        //menambahkan produk ke array of produk
        tambahProduk();

        //instansiasi Produk Adapter
        adapter = new ProdukAdapter(getApplicationContext(), produks);

        //set adapter pada listview
        listView.setAdapter(adapter);

        //listener jika item pada listview di klik
        listView.setOnItemClickListener(new
        AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> parent, View view,
            int position, long id) {
                Toast.makeText(getApplicationContext(),
                produks.get(position).getNama(), Toast.LENGTH_LONG).show();
            }
        });
    }

    void tambahProduk() {
        Produk p1 = new Produk("Samsung", 1200000);
        for (int i = 0; i < 20; i++) {
            produks.add(p1);
        }
    }
}
```

```

<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin">

    <ListView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/listProduk"
        android:layout_centerVertical="true"
        android:layout_centerHorizontal="true" />
</RelativeLayout>

```

e. *Run* aplikasi anda.

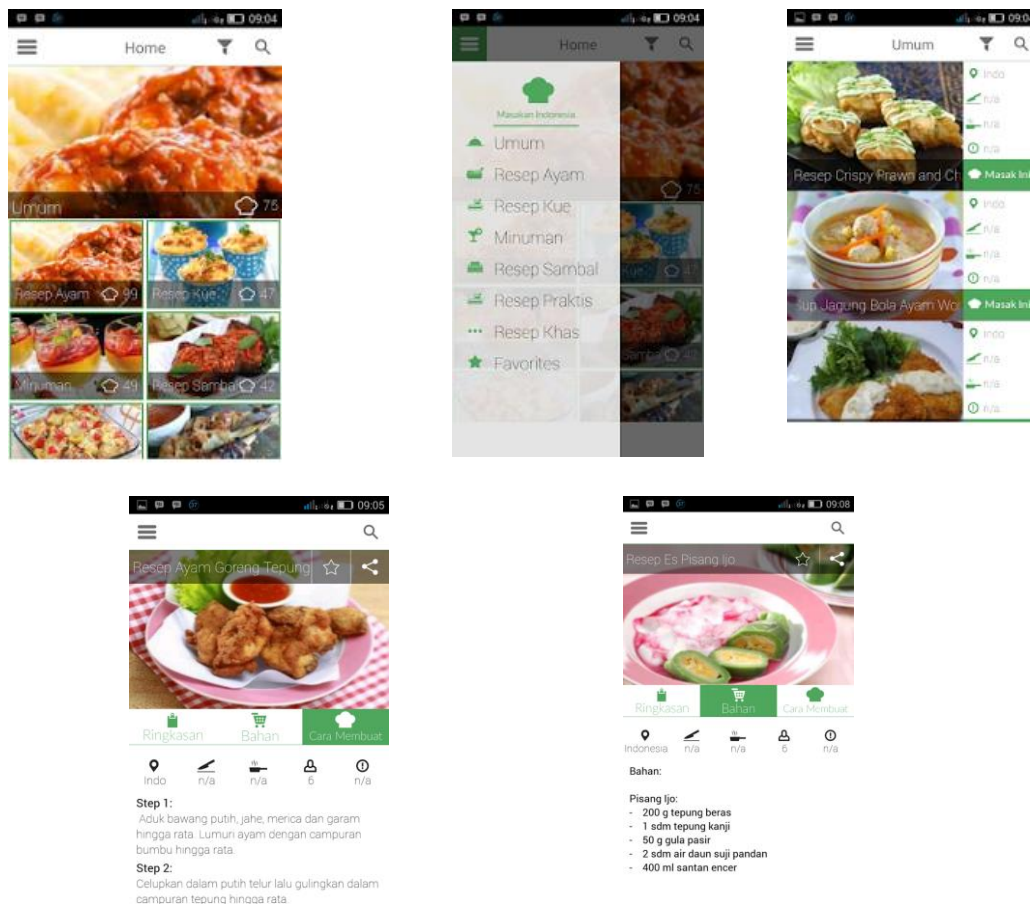
## **BAGIAN II: DELIVERABLES**

Pada minggu ke-14, Anda diharapkan menyelesaikan pembuatan aplikasi **Buku Resep Makanan**. Aplikasi ini dibuat dengan menggunakan keterangan aplikasi “**BUKU RESEP MASAKAN INDONESIA**” karya JogjaTech Studio<sup>1</sup> pada Google Play Store sebagai referensi. Tujuan kegiatan ini adalah Anda dapat memahami komponen-komponen dan detail teknis dari sebuah aplikasi populer yang sudah di-publish ke pasar aplikasi *mobile*. Dengan demikian, Anda juga nantinya dapat membangun aplikasi sejenis dan menjadi potfolio Anda. Sebagai informasi, jumlah *install* aplikasi ini per tanggal 22 Februari 2016 berada di kisaran 500,000 - 1,000,000 *install* (update terakhir aplikasi ini, yaitu versi 3.0 adalah 15 Desember 2014).

### **Deskripsi Umum Aplikasi**

Aplikasi **Buku Resep Makanan** ini bertujuan untuk menampilkan informasi kumpulan resep makanan. Untuk pembuatan aplikasi ini, Anda dibebaskan untuk menyediakan data resep makanan yang Anda inginkan, termasuk gambar, teks dan *resources* yang lainnya. Gambar 1 menunjukkan struktur umum aplikasi “**BUKU RESEP MASAKAN INDONESIA**” karya JogjaTech Studio. Desain struktur aplikasi pada praktikum ini diserahkan kepada Anda, tapi diharapkan tidak terlalu kompleks jika dibandingkan dengan aplikasi “**BUKU RESEP MASAKAN INDONESIA**” karya JogjaTech Studio.

<sup>1</sup> Referensi: <https://play.google.com/store/apps/details?id=com.kumpulan.resep.masakan.indonesia.app>



Gambar 1. Contoh tampilan aplikasi Buku Resep Makanan merujuk pada “BUKU RESEP MASAKAN INDONESIA” karya JogjaTech Studio

Pada praktikum-praktikum di minggu-minggu mendatang, Anda diharapkan untuk mulai membangun komponen-komponen aplikasi tersebut. Contohnya, Anda akan membuat komponen *Activity* aplikasi Buku Resep Makanan buatan Anda pada praktikum minggu ke-3 sesi-2 (topik pada praktikum ini adalah *Activity*). Hasil pembuatan komponen tersebut yang akan menjadi Deliverables Anda setiap sesi praktikum. Di minggu ke-14, Anda diharapkan menyetor aplikasi Buku Resep Makanan secara lengkap.

### Deliverables Sesi Praktikum Ini

1. Implementasikan sebuah *notification* pada aplikasi BUKU RESEP MAKANAN anda, dan kombinasikan dengan *Broadcast Receiver* dan *Service* yang pernah anda buat sebelumnya.
2. Implementasikan sebuah *listview* pada salah satu *activity* di aplikasi anda.
3. Sebagai tambahan submission file untuk praktikum sesi ini, Anda diharuskan untuk melampirkan sebuah file .txt yang berisi keterangan apa yang dilakukan oleh *broadcast receiver*, *service* dan *notification* yang Anda buat untuk aplikasi anda.