

Photo Gallery App



Tutorial Membuat Aplikasi Galeri Foto Android

Aplikasi Galeri Foto Android

Detail & Tujuan :

Membuat Aplikasi Galeri Foto untuk menampilkan gambar yang tersedia dalam folder Pictures di MMC. Aplikasi dengan konsep sederhana meload semua gambar dan ditampilkan secara thumbnail dan masing masing thumbnail terdapat feature untuk melihat nya secara full screen.

Materi :

- Splashscreen
- Bitmap Converter
- Environment
- GridView

Tutorial :

Pembangunan aplikasi ini menggunakan tool android studio namun secara umum masih tak jauh berbeda dengan eclipse untuk file core terkait class dan layout xml nya, mungkin langsung saja silahkan membuat project baru, kemudian dalam project tersebut untuk pertama kita buat tampilan splash dengan layout xml seperti dibawah ini.



splash.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/splash">

    <ProgressBar
        style="?android:attr/progressBarStyleLarge"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/progressBar"
        android:layout_centerVertical="true"
        android:layout_centerHorizontal="true" />

</RelativeLayout>
```

bisa kita lihat palette diatas menggunakan tambahan progress bar untuk animasi tunggu saat activity tertahan dalam activity, background pada activity tersebut menggunakan splash.png yang telah tersedia pada folder drawable.

Sedangkan untuk class yang dijadikan core dari activity ini adalah Splash.java silahkan buat class tersebut.

Splash.java

```
package imagegallery.creatorb.id.androidimagegallery;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.os.Handler;

public class Splash extends Activity {

    //set durasi selama 3 detik
    private final int SPLASH_DURATION = 3000;

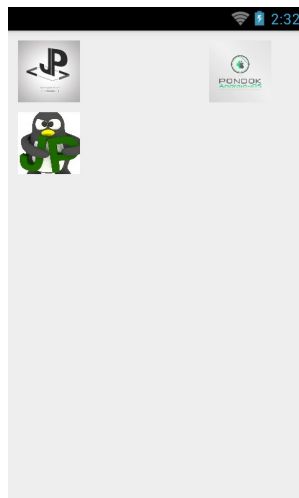
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.splash);

        //buat handler baru untuk memulai activity selanjutnya
        new Handler().postDelayed(new Runnable() {
            @Override
            public void run() {
                //pasang intent untuk menuju activity selanjutnya
                Intent next = new Intent(getApplicationContext(),MainActivity.class);
                startActivity(next);
                //akhiri activity splash
                finish();
            }
        }, SPLASH_DURATION);
    }
}
```

Untuk penjelasan rinci telah disampaikan melalui komen pada sintak diatas, dimana pada intinya memasang parameter untuk durasi hold activity selama 3000ms atau 3 Detik saat activity meload layout splash.xml

hingga nantinya jika waktu habis maka activity tersebut akan berpindah menggunakan intent menuju activity berikutnya yaitu class MainActivity dan akhiri aplikasi yang berjalan saat ini dengan finish();

Setelah berakhir, maka aplikasi akan meload activity_main dimana layout ini sendiri digunakan untuk class MainActivity.



activity_main.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <GridView
        android:id="@+id/gridview"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_margin="5dp"
        android:numColumns="auto_fit"
        android:columnWidth="100dp" />

</LinearLayout>
```

Layout diatas menggunakan linear layout dengan palette gridview yang nantinya akan digunakan untuk meload gambar gambar yang secara dinamis pada id gridview.

untuk class dari layout diatas berada pada *MainActivity.java*

```
package imagegallery.creatorb.id.androidimagegallery;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.os.Environment;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.AdapterView.OnItemSelectedListener;
import android.widget.AdapterView.OnItemSelectedListener;
import android.widget.Toast;

import java.io.File;

public class MainActivity extends Activity {

    // deklarasikan variabel
    private String[] FilePathStrings;
    private String [] FileNameStrings;
    private File[] listFile;
    GridView grid;
    GridViewAdapter adapter;
    File file;

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

        //pertama cek sdcard terlebih dahulu
        if(!Environment.getExternalStorageState().equals(Environment.MEDIA_MOUNTED)){
            Toast.makeText(this, "Masukan kartu memori", Toast.LENGTH_LONG)

```

```

        .show();
    }else {
        //set lokasi folder photos di sdcard
        file = new File(Environment.getExternalStorageDirectory() + File.separator + "Pictures");
        //jika folder Pictures tidak ada maka buat folder Pictures
        file.mkdirs();
    }

    if (file.isDirectory()){
        listFile = file.listFiles();
        //buat array string utk path
        FilePathStrings = new String[listFile.length];
        //buat array string utk nama file
        FileNameStrings = new String[listFile.length];

        for (int b = 0; b < listFile.length; b++){
            //get path dari photos
            FilePathStrings[b] = listFile[b].getAbsolutePath();
            //get nama photos
            FileNameStrings[b] = listFile[b].getName();
        }
    }

    //locate gridview
    android.widget.GridView grid = (android.widget.GridView) findViewById(R.id.gridview);
    //masukkan string dalam array pada class GridViewAdapter
    adapter = new GridViewAdapter(this, FilePathStrings, FileNameStrings);
    //set LA ke gridview
    grid.setAdapter(adapter);

    //gridview select/item clicked
    grid.setOnItemClickListener(new AdapterView.OnItemClickListener() {
        @Override
        public void onItemClick(AdapterView<?> parent, View view, int position, long id) {

            Intent b = new Intent(MainActivity.this, LihatPhotos.class);

            b.putExtra("filepath",FilePathStrings);
            b.putExtra("filename",FileNameStrings);
            b.putExtra("position", position);
        }
    });

```

```
        startActivity(b);
    }
});

}
}
```

Alur algoritma diatas diawali dengan pengecekan kartu memori terlebih dahulu, apakah ada memori yang terpasang pada device android. Dimana jika tidak ada maka akan muncul pesan untuk memasukkan kartu memori dan selanjutnya jika memori tersedia maka aplikasi akan mencoba mencari folder bernama Pictures di memori dan meload isi dari folder dan menampilkan nya pada gridView. Jika tidak ada folder bernama Pictures maka aplikasi akan membuat folder bernama Pictures yang nantinya jika terisi gambar maka gambar tersebut yang akan di load pada gridView di layout activity_main. Sedangkan dalam alur algoritma keseluruhan tidak sampai disitu saja tetapi terdapat fungsi `setOnClickListener` dimana item yang dimaksud disini adalah value dari gridView yang berarti gambar gambar yang ada pada folder Pictures yang muncul secara thumbnail memiliki aksi yang sama ketika di click yaitu akan membawa menuju algoritma intent b dimana intent b sendiri merupakan pengalihan dari class MainActivity menuju class LihatPhotos dengan memasukkan beberapa value tambahan terkait detail salah satunya yaitu nama gambar itu sendiri. Sebelum membahas activity LihatPhotos ada baiknya kita membahas adapter dari gridView sendiri berikut class nya.

LihatPhotos.java


```
package imagegallery.creatorb.id.androidimagegallery;

import android.app.Activity;
import android.content.Context;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.BaseAdapter;
import android.widget.ImageView;

/**
 * Created by creatorbe on 3/22/15.
 */
public class GridViewAdapter extends BaseAdapter{

    //deklarasikan variabel
    private Activity activity;
    private String[] filepath;
    private String[] filename;

    private static LayoutInflater inflater = null;

    public GridViewAdapter(Activity b, String[] fpath, String[] fname){
        activity = b;
        filepath = fpath;
        filename = fname;
        inflater = (LayoutInflater) activity.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
    }

    public int getCount(){
        return filepath.length;
    }

    public Object getItem(int position){
        return position;
    }

    public long getItemId(int position){
```

```

    return position;
}

public View getView(int position, View convertView, ViewGroup parent){
    View vi = convertView;
    if (convertView == null)
        vi = inflater.inflate(R.layout.grid_item, null);
    //locate bbrapa palette di layout grid_item
    //tampilkan title
    //  TextView titles = (TextView) vi.findViewById(R.id.titles);
    //tampilkan gambar
    ImageView photos = (ImageView) vi.findViewById(R.id.photos);

    //settext by position
    //  titles.setText(filename[position]);

    //decode filepath dg bitmapfactory by position
    Bitmap bmp = BitmapFactory.decodeFile(filepath[position]);

    //set result to imageview
    photos.setImageBitmap(bmp);
    return vi;
}
}

```

Diatas bisa kita lihat core java dari adapter nya, dengan beberapa pendeklarasian dan method, konstruktor yang juga membantu menangani activity di MainActivity, bisa kita lihat dengan komponen path, nama file, get Item dan posisi dari perulangan nya dalam menghandle grid_item dimana itu digunakan untuk memunculkan data per item gambar yang muncul di activity_main. Disana juga dilakukan decode bitmap pada masing masing data dan memunculkannya untuk script set text diatas di komen agar yang muncul di activity_class hanya berupa thumbnail saja jika anda ingin memunculkan nama maka di

uncomment set text nya maka itu akan memunculkan nama file sesuai posisi perulangannya. Jika hendak mengaktifkan settext maka di layout `grid_item` yang menangani tampilan per item juga di beri palette textview untuk meload string string nama file namun dikarenakan dalam tutorial ini tidak mencantumkan nama file cukup thumbnail dikarenakan untuk detail terkait nama file lebih baik diletakkan pada class `LihatPhotos`.

Berikut tampilan layout per item.

grid_item.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="horizontal"
    android:padding="5dip">

    <ImageView
        android:id="@+id/photos"
        android:layout_width="100px"
        android:layout_height="100px"
        android:columnWidth="90dp"
        android:numColumns="auto_fit"
        android:stretchMode="columnWidth" />

</LinearLayout>
```

Selanjutnya activity yang menangani class terkait detail keterangan gambar yaitu `LihatPhotos`. Berikut tampilan layout nya.



```
<?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="match_parent"
    android:background="@android:color/black">

    <ImageView
        android:id="@+id/full_photos"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_centerInParent="true" />

    <LinearLayout
        android:orientation="vertical"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:id="@+id/linearLayout">
        <TextView
            android:id="@+id/full_text"
            android:text="Title"
            android:gravity="center_horizontal"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:textColor="@android:color/white" />

    </LinearLayout>

</RelativeLayout>
```

Diatas adalah tampilan respon intent b saat aplikasi menjalankan `setOnClickListener` pada item data di activity `MainActivity`. Maka akan di activity akan menuju tampilan diatas dengan dua palette pokok yaitu `imageView` untuk memunculkan data thumbnail sebelumnya namun kali ini dalam ukuran `match_parent` dengan kata lain menyesuaikan ukuran gambar sebenarnya dalam layout, dibawah itu ada `textView` itu adalah palette yang menampung string string dari filename gambar itu sendiri. Oke mungkin langsung saja menuju class `LihatPhotos.java`

```
package imagegallery.creatorb.id.androidimagegallery;

import android.app.Activity;
import android.content.Intent;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.os.Bundle;
import android.widget.ImageView;
import android.widget.TextView;

/**
 * Created by creatorbe on 3/22/15.
 */
public class LihatPhotos extends Activity {
    //Deklarasi Variabel
    TextView titles;
    ImageView photos;

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

        //menerima data dari MainActivity ketika mengklik item grid
        Intent b = getIntent();
```

```

//get posisi
int position = b.getExtras().getInt("position");

//get string array path
String[] filepath = b.getStringArrayExtra("filepath");

//get string array name
String[] filename = b.getStringArrayExtra("filename");

//locate textview di lihat_photos
titles = (TextView) findViewById(R.id.full_text);

//parsing text ke textview sesuai posisi
titles.setText(filename[position]);

//locate imageview di lihat_photos
photos = (ImageView) findViewById(R.id.full_photos);

//decode bmp sesuai posisi
Bitmap bmp = BitmapFactory.decodeFile(filepath[position]);

//set decoded ke imageview
photos.setImageBitmap(bmp);
}
}

```

Ya benar isinya tak lebih hanya sekedar pencocokan key dan value saja, dimana string title dimasukkan pada id full_text sesuai posisi yang di dapat dan gambar juga di masukkan pada imageview dengan id full_photos.

Sekian, semoga bermanfaat dan silahkan dikembangkan untuk menjadi lebih baik lagi.