



This repository Search

Pull requests Issues Gist



stikompcir / Android-Fundamental

Unwatch 1

Star 0

Fork 0

<> Code

Issues 0

Pull requests 0

Wiki

Pulse

Graphs

Settings

Pengembangan Aplikasi Android — Edit

1 commit

1 branch

0 releases

1 contributor

Tree: 9ea8dbd717

New pull request

Create new file

Upload files

Find file

Clone or download



stikompcir committed on GitHub Create README.md

Latest commit 9ea8dbd 15 hours ago

README.md

Create README.md

15 hours ago

README.md

Android-Fundamental

Sejarah android

Sejak tahun 2008, Android secara bertahap telah melakukan sejumlah pembaruan untuk meningkatkan kinerja sistem operasi, menambahkan fitur baru, dan memperbaiki bug yang terdapat pada versi sebelumnya. Setiap versi utama yang dirilis dinamakan secara alfabetis berdasarkan nama-nama makanan pencuci mulut atau cemilan bergula; misalnya, versi 1.5 bernama Cupcake, yang kemudian diikuti oleh versi 1.6 Donut. Versi terbaru adalah 6.0 Marshmallow, yang dirilis pada 19 Agustus 2015.

Kode sumber untuk Android tersedia di bawah lisensi perangkat lunak sumber terbuka dan bebas. Google menerbitkan sebagian besar kode (termasuk kode jaringan dan telepon) di bawah Lisensi Apache versi 2.0.[135][136][137] Sisanya, perubahan kernel Linux berada di bawah GNU General Public License versi 2. Open Handset Alliance mengembangkan perubahan kernel Linux dengan kode sumber terbuka yang dipublikasikan setiap saat. Selebihnya, Android dikembangkan secara pribadi oleh Google, dengan kode sumber yang diterbitkan untuk umum ketika versi baru diluncurkan. Biasanya Google bekerja sama dengan produsen perangkat keras untuk mengembangkan sebuah perangkat "andalan" (misalnya seri Google Nexus) yang disertai dengan versi baru Android, kemudian menerbitkan kode sumbernya setelah perangkat tersebut dirilis.

Versi	Nama kode	Tanggal rilis	Level API	Distribusi
6.0	Marshmallow	19 Agustus 2015	23	
5.x	Lollipop	15 Oktober 2014	21	
4.4.x	KitKat[179]	31 Oktober 2013	[180] 19	24,5%
4.3.x	Jelly Bean	24 Juli 2013	18	8%
4.2.x	Jelly Bean	13 November 2012	17	20,7%
4.1.x	Jelly Bean	9 Juli 2012	16	25,1%
4.0.3–4.0.4	Ice Cream Sandwich	16 Desember 2011	15	9,6%
3.2	Honeycomb	15 Juli 2011	13	
3.1	Honeycomb	10 Mei 2011	12	

2.3.3–2.3.7	Gingerbread	9 Februari 2011	10	11,7%
2.3–2.3.2	Gingerbread	6 Desember 2010	9	
2.2	Froyo	20 Mei 2010	8	0,7%
2.0–2.1	Eclair	26 Oktober 2009	7	
1.6	Donut	15 September 2009	4	
1.5	Cupcake	30 April 2009	3	

Link Sumber [https://id.wikipedia.org/wiki/Android_\(sistem_operasi\)](https://id.wikipedia.org/wiki/Android_(sistem_operasi))

Perkakas yang dibutuhkan

- * JDK
- * SDK
- * IDE Android Studio

- ketiga perkakas yang dibutuhkan bersifat freeware atau gratis

silahkan download di situs resminya

- [JDK](#)
- [IDE Android Studio](#)
- [SDK](#)

Pemasangan perkakas yang di butuhkan

Setelah Semua perkakas sudah di download, mari kita pasang pada komputer kita

Windows

- Cukup secara default saja, klik next, next sampai proses pemasangan selesai.
- Konfigurasi JDK
 - Buka Windows Explorer, lalu klik kanan My Computer. Klik Properties
 - Cari Advanced sistem setting [gb]
 - Klik tombol Environment Variable
 - Pada bagian User variables fo . klik tombol new
 - Pada kolom isian Variable name PATH
 - pada Variable value C:\Program Files\Java\jdk1.7.0_60\bin
 - Pada bagian System variables, cari Variable PATH lalu klik tombol edit
 - Hati-hati ketika mengedit konfigurasi System Variable, karna jika sampai terhapus akan terjadi eror yang tidak diinginkan.
 - Isikan lokasi instalasi JDK ch: C:\Program Files\Java\jdk1.7.0_60\bin
 - Setelah Selesai, periksa kembali langkah sebelumnya yang telah dijelaskan
 - Klik oke dan Oke :)
 - Jangan tenang dulu, kita lihat apakah sudah berhasil atau muncul error !
 - Buka cmd dengan menekan tombol Window + r , ketikan cmd lalu enter
 - Ketikan `$ java -version & $ javac -version`

Linux

- Copy file JDK yang telah di Unduh dengan Command line
 - Buat folder /usr/lib/jvm dengan command


```
$ sudo mkdir /usr/lib/jvm/
```
 - copykan file JDK yang telah di download


```
$ sudo mv jdk1.7.x_xx.tar.gz
```
 - Masuk ke folder


```
$ sudo cd /usr/lib/jvm/
```
 - Extract


```
$ sudo tar xfv jdk1.7.x_xx.tar.gz
```
 - Membuat Path


```
$ sudo update-alternatives --install "/usr/bin/java" "java" "/usr/lib/jvm/jdk1.7.x_xx/bin/java" 1
$ sudo update-alternatives --install "/usr/bin/javac" "javac" "/usr/lib/jvm/jdk1.7.x_xx/bin/javac" 1
$ sudo update-alternatives --install "/usr/bin/javaws" "javaws" "/usr/lib/jvm/jdk1.7.x_xx/bin/javaws" 1
```

Instal IDE Android Studio

- Windows Klik dua kali installer lalu biarkan instal secara default saja sampai selesai
- Linux buka command line


```
$ sudo add-apt-repository ppa:paolorotolo/android-studio
$ sudo apt-get update
$ sudo apt-get install android-studio
```

SDK

Setelah android studio berhasil di install, maka hal yang harus dilakukan adalah mengupdate SDK sesuai kebutuhan. dengan cara klik icon SDK manager lalu klik update

Memrogram

Skill yang dibutuhkan untuk pengembangan aplikasi android adalah :

- Pernah belajar tentang Bahasa pemrograman Java
- Object Oriented Programing
- Struktur Data

namun jika belum pernah sama sekali belajar tentang hal di atas, tak apa asal **Punya Kemauan** yang kuat untuk tetap belajar.

Komponen Aplikasi

Activity

Activity adalah apa yang dilihat oleh user di dalam aplikasi, user bisa melakukan inputan. Dan apa yang dilihat oleh user itu adalah Activity

- Setiap class java dinyatakan sebuah activity jika class tersebut meng-extends pada class activity.

```
package com.dicoding.maman.hitungluas;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {
    private EditText edtPanjang, edtLebar;
```

```

private Button btnHitung;
private TextView txtLuas;

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

    setContentView(R.layout.activity_main);

    getSupportActionBar().setTitle("Hitung Luas Persegi Panjang");

    edtPanjang = (EditText) findViewById(R.id.edt_panjang);
    edtLebar = (EditText) findViewById(R.id.edt_lebar);
    btnHitung = (Button) findViewById(R.id.btn_hitung);
    txtLuas = (TextView) findViewById(R.id.txt_luas);

    btnHitung.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            try{
                String panjang = edtPanjang.getText().toString().trim();
                String lebar = edtLebar.getText().toString().trim();

                Double p = Double.parseDouble(panjang);
                Double l = Double.parseDouble(lebar);

                Double luas = p * l ;

                txtLuas.setText("Luas : " + luas);
            }catch (NumberFormatException e){
                e.printStackTrace();
            }
        }
    });
}

```

- Activity pada umumnya memiliki sebuah tampilan dalam format layout.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
    android:orientation="vertical"
    tools:context="com.dicoding.maman.hitungluas.MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Panjang"
        android:layout_marginBottom="8dp"/>

    <EditText
        android:id="@+id/edt_panjang"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="0"
        android:inputType="numberDecimal"
        android:layout_marginBottom="5dp"/>

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Lebar"

```

```

        android:layout_marginBottom="8dp"/>
<EditText
    android:id="@+id/edt_lebar"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="0"
    android:inputType="numberDecimal"
    android:layout_marginBottom="5dp"/>
<Button
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/btn_hitung"
    android:text="Hitung"
    android:layout_marginBottom="8dp"/>
<TextView android:text="Hasil" android:id="@+id/txt_luas" android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="8dp"
    android:textSize="18sp"
    android:textStyle="bold"/>
</LinearLayout>

```

- Activity menganut konsep Stack: Last in First Out (LIFO) yang mana setiap activity yang terakhir diciptakan atau ditampilkan dilayar akan keluar pertama kali dari stack jika pengguna menekan tombol:
 - physical back button atau
 - Back Button pada Action bar Aplikasi.

Contoh aplikasi menggunakan activity[link_gambar, tambah HitungLuas.apk]

Intent

Mekanisme pengiriman pesan dari satu komponen ke komponen lainnya

Intent terbagi menjadi dua yaitu

- Intent explicit untuk mengaktifkan komponen-komponen dalam satu aplikasi yang sama misal: Berpindah activity ke activity lainnya
- Intent implicit untuk mengaktifkan komponen dari aplikasi lain misal : memanggil dial number, dll

Contoh Kode yang menggunakan Class Intent

```

package com.stikompcir.maman.contohintent;

import android.content.Intent;
import android.net.Uri;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;

public class MainActivity extends AppCompatActivity {

    private Button btnSub1, btnSub2, btnDial;

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

        btnSub1 = (Button)findViewById(R.id.btn_activity_sub_1);
        btnSub2 = (Button)findViewById(R.id.btn_activity_sub_2);
        btnDial = (Button)findViewById(R.id.btn_activity_dial);

        btnSub1.setOnClickListener(new View.OnClickListener() {

```

```

@Override
public void onClick(View view) {
    Intent intent = new Intent(MainActivity.this, Sub1Activity.class);
    startActivity(intent);
}
});

btnSub2.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View view) {
    Intent intent = new Intent(MainActivity.this, Sub2Activity.class);
    intent.putExtra(Sub2Activity.KEY_DATA, "Training Aplikasi Android");
    startActivityForResult(intent, 0);
}
});

btnDial.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View view) {
    Intent intent = new Intent(Intent.ACTION_DIAL, Uri.parse("tel:085283868252"));
    startActivity(intent);
}
});
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is present.
    getMenuInflater().inflate(R.menu.menu_main, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    // Handle action bar item clicks here. The action bar will
    // automatically handle clicks on the Home/Up button, so long
    // as you specify a parent activity in AndroidManifest.xml.
    int id = item.getItemId();

    //noinspection SimplifiableIfStatement
    if (id == R.id.action_settings) {
        return true;
    }

    return super.onOptionsItemSelected(item);
}
}

```

Fragment

Fragment adalah sub set dari Activity, tujuannya sama degan Activity menampilkan UI, membiarkan user berinteraksi didalamnya. Perbedaan mendasar antara Fragment & Activity adalah Fragment selalu bergantung kepada Activity. Kenapa harus ada Fragment karena density layar yang berbeda diperlukan mekanisme dimana sebuah aplikasi kita bisa responsive terhadap ukuran device. [kode: contoh class menggunakan fragment] untuk membuat sebuah fragment kita perlu meng-extend Class Fragment

```

package com.stikompcir.maman.fragment;

import android.app.Fragment;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;

import com.stikompcir.maman.Communicator;
import com.stikompcir.maman.R;

```

```
/**
 * Created by Maman on 7/26/2016.
 */
public class FragmentA extends Fragment implements View.OnClickListener {

    Button button;
    int counter = 0;
    Communicator comm;

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState){
        return inflater.inflate(R.layout.fragment_a,container,false);
    }

    @Override
    public void onActivityCreated(Bundle savedInstanceState){
        super.onActivityCreated(savedInstanceState);
        comm = (Communicator) getActivity();
        button = (Button) getActivity().findViewById(R.id.button);
        button.setOnClickListener(this);
    }

    @Override
    public void onClick(View view){ /* untuk menghendle method onclick kita perlu implements View.OnClickListener */
        counter++; // setiap diclick akan bertambah nilai dari variabel counter.
        comm.respond("Tombol telah di sentuh "+counter+" kali"); //dikirim ke MainActivity melalui method pada interf
    }
}
```

[aplikasi: FragmentBasic]

