

BADOY STUDIO



ANDROID STUDIO MASTERY

By Gunther

PANDUAN LENGKAP MEMBUAT
APLIKASI ANDROID UNTUK
PEMULA DALAM 9 HARI

ANDROID STUDIO MASTERY
“ Panduan Membuat Aplikasi Android Untuk
Pemula dalam 9 Hari “

Penulis :

Guntoro

Editor :

Meidiana Putri

Desain Cover :

Irfan Resdiyanto

Penerbit :

Badoy Studio

Rangkasbitung – Kabupaten Lebak -Banten

Hak Cipta dilindungi undang-undang

*Dilarang mengcopy dan memperbanyak sebagian atau
seluruh isi buku ini dalam bentuk apapun tanpa seizin dari
penerbit.*

All right reserved

Daftar Isi

Daftar Isi	3
Kata Pengantar	4
Bab 1 Pengenalan Android dan Java	5
Bab 2 Installasi Tools	12
Bab 3 Membuat Project Android Pertama	31
Bab 4 Running Project Menggunakan AVD	37
Bab 5 Install Emulator Genymotion	42
Bab 6 Running Project Langsung Di Smartphone	52
Bab 7 Struktur Project dan Antarmuka Android Studio	57
Bab 8 Activity dan Siklus Hidup Android	68
Bab 9 Pengertian, Fungsi, dan Jenis Intent	75
Bab 10 Komponen View	88
Bab 11 Komponen View Group	117
Bab 12 Style dan Tema Aplikasi	146
Bab 13 Membuat Aplikasi Tebak Gambar	152
Bab 14 Build APK	163
Bab 15 Daftar Akun Google Developer (Google Console)	169
Bab 16 Rilis Aplikasi Ke Google Playstore	176

Kata Pengantar

Menjadi android developer saat ini merupakan salah satu profesi yang cukup menjanjikan. Ada banyak peluang ketika kita sudah mempunyai keahlian untuk membuat aplikasi android. Tinggal pilih saja mau bekerja di startup, corporate, pengajar, freelancer, atau admob publisher.

Rate gaji untuk junior android developer adalah 5 – 8 juta / bulan. Angka yang cukup besar untuk pemula. Berapa jumlahnya kalau kita sudah miliki skill yang mumpuni lagi. Sekitar 8 – 15 juta, cukup tinggi bukan ?

Lalu kini giliran anda apakah mau jadi penonton atau ikut andil bagian dalam moment yang baik untuk mendapatkan penghasilan yang lebih besar dari menjadi android developer.

Dengan mempelajari ebook Android Studio Mastery anda akan belajar step by step cara membuat aplikasi android dari yang paling dasar.

Selamat belajar dan semoga dipermudah.

BAB 1

PENGENALAN OS ANDROID DAN JAVA



Android

Android merupakan *Operating System* (OS) berbasis Linux yang sifatnya open source. Andy Rubin adalah orang dibalik OS ini. Tahun 2003 Ia bersama dengan kawannya mengumumkan perusahaan Android inc yang bertempat di California. kemudian Google secara penuh membelinya pada tahun 2005.

Saat ini Android digunakan sebagai sistem operasi untuk perangkat mobile (bergerak) seperti : *smartphone*, *smartwatch*, dan lainnya.



Versi-versi android sendiri sampai saat ini sudah cukup banyak. Dirilis pertama kali pada tahun 2009 yaitu versi Android 1.1. Uniknya hampir semua versi android menggunakan nama makanan.

Inilah versi-versi android :

No	Nama Versi	Tahun Rilis
1.	Cupcake	2009
2.	Donut	2009
3.	Éclair	2009
4.	Froyo	2010
5.	Gingerbread	2010
6.	Honeycomb	2011
7.	Ice Cream Sandwich	2011
8.	Jelly Bean	2012
9.	Kitkat	2013
10.	Lollipop	2014
11.	Marshmallow	2015
12.	Nougat	2016
13.	Oreo	2017

Keunggulan Android

Keunggulan android cukup banyak, diantaranya yaitu :

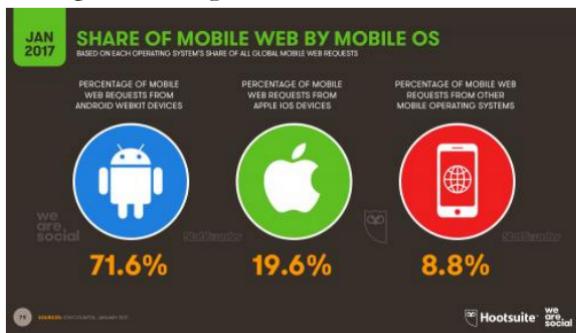
- Android merupakan sistem operasi yang open source sehingga banyak sekali orang-orang yang mengembangkan aplikasi untuk Android.

- Perangkatnya beragam dari mulai yang low end (murah) sampai dengan high end (mahal)
- Fitur dari android sangat lengkap. Sehingga developer android nantinya akan bisa membuat aplikasi yang powerfull.

Pengguna Android

Apakah anda salah satu pengguna smartphone dengan sistem operasi android ? jika ya berarti anda termasuk dari 71 % pengguna OS android di dunia.¹

Sedangkan di Indonesia sendiri jumlah pengguna smartphone dengan OS android berkisar sekitar 80 juta.



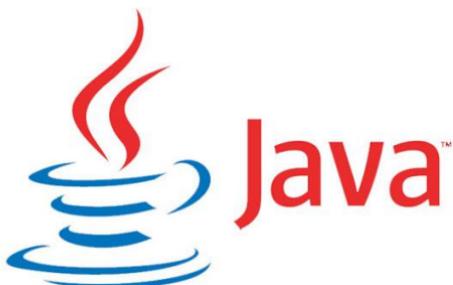
¹ Survey yang dilakukan oleh *we are social* pada tahun 2017 menyebutkan dari ketiga OS : IOS, Android, dan sistem operasi mobile lainnya, Android menempati peringkat pertama

Ini merupakan peluang yang bagus untuk para developer android. Karena pasti kebutuhan akan aplikasi android semakin meningkat.

Java

Java merupakan Salah satu bahasa pemrograman yang paling popular.² Dikarenakan bisa dijalankan disemua platform dan OS yang berbeda-beda.

Walaupun google telah resmi mendukung c++ dan baru-baru ini kotlin, tetap saja java jadi primadona untuk para developer android. Ini dikarenakan library java cukup lengkap sehingga memudahkan programmer.



² Survey dilakukan oleh stackoverflow.com tahun 2018.

Ada baiknya pelajari dasar dan oop bahasa pemrograman java agar pada saat membuat aplikasi android studio tidak kesulitan dalam pengkodingan javanya.

BAB 2

INSTALL TOOLS



JDK

Untuk menulis bahasa pemrograman Java kita harus menginstall JDK. JDK merupakan perangkat lunak yang digunakan untuk melakukan proses kompilasi dari kode java ke bytecode yang dapat dimengerti dan dijalankan oleh JRE (Java Runtime Envirotment).³

Itu artinya sebelum menginstall android studio dan melakukan proses ngoding menggunakan bahasa pemrograman java kita harus menginstall juga menyetting jdk pada komputer yang kita gunakan.

Menginstall JDK

Berikut ini langkah-langkah untuk menginstall JDK :

- o Download jdk 8 dari situs oracle atau langsung, pilih sesuai byte OS yang kamu gunakan.⁴

³ Sumber : medium.com

⁴ Link download lengkap JDK :
<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

Java SE Development Kit 8u101		
You must accept the Oracle Binary Code License Agreement for Java SE to download this software.		
<input type="radio"/> Accept License Agreement	<input checked="" type="radio"/> Decline License Agreement	
Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	77.77 MB	jdk-8u101-linux-arm32-vfp-hfll.tar.gz
Linux ARM 64 Hard Float ABI	74.72 MB	jdk-8u101-linux-arm64-vfp-hfll.tar.gz
Linux x86	160.28 MB	jdk-8u101-linux-i586.tar.gz
Linux x86	174.96 MB	jdk-8u101-linux-i586.rpm
Linux x86	158.27 MB	jdk-8u101-linux-x64.rpm
Linux x86	172.95 MB	jdk-8u101-linux-x64.tar.gz
Mac OS X	227.36 MB	jdk-8u101-macosx-x64.dmg
Solaris SPARC 64-bit	139.66 MB	jdk-8u101-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	98.96 MB	jdk-8u101-solaris-sparcv9.tar.gz
Solaris x64	140.33 MB	jdk-8u101-solaris-x64.tar.Z
Solaris x64	96.78 MB	jdk-8u101-solaris-x64.tar.gz
Windows x86	188.32 MB	jdk-8u101-windows-i586.exe
Windows x64	193.68 MB	jdk-8u101-windows-x64.exe

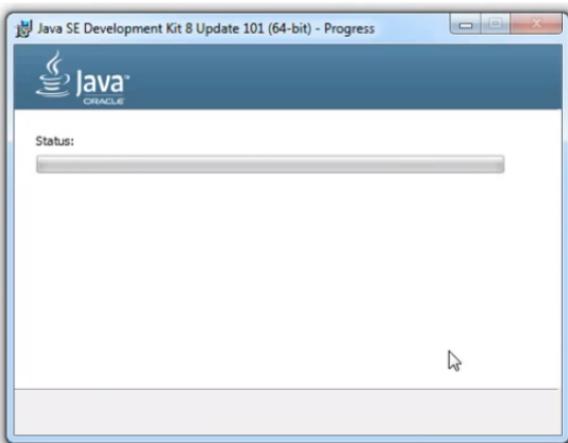
- Jika sudah terdownload buka file exe yang sudah terdownload di komputer kemudian klik **next**, seperti gambar berikut :



- o Pilih **feature** (ikuti gambar berikut) kemudian klik **next.**



- o Tunggu progress sampai dengan selesai.



- o Tentukan lokasi penyimpanan folder jdk dimana nantinya disimpan, lalu klik **next**.



- o Proses instalasi akan dimulai dan tunggu sampai selesai.



- o Jika proses instalasi sudah selesai silahkan klik **close**



Konfigurasi JDK (Path)

Supaya Android Studio bisa kita gunakan dengan normal maka setelah menginstall JDK kita harus melakukan konfigurasi PATH JDKnnya.⁵

Langkah-langkahnya sebagai berikut :

- o Copykan alamat tempat folder penginstallan jdk dengan cara : **Computer > local Disk (c) >**

⁵ OS yang digunakan untuk menjalankan android studio yang penulis gunakan adalah Windows 7

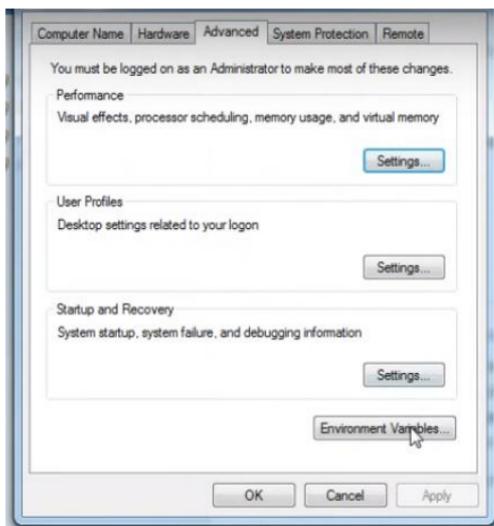
program files > java > folder jdk > bin >
javac



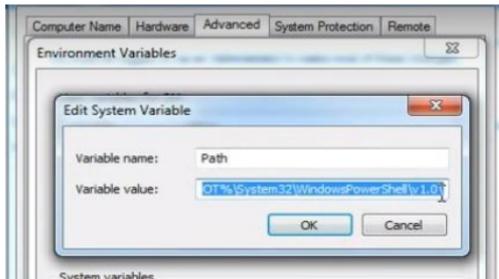
- o Setelah itu klik kanan pada **Computer** > **Properties > Advanced System Settings**



- o Pilih **Environment Variable**



- Cari variable **Path** dan pastekan lokasi penyimpanan folder jdk pada **variable value** lalu klik ok.



- Install dan Setting Path JDK sudah selesai.

Android Studio

Dikutip dari **developer.android.com** Android Studio adalah Integrated Development Environment (IDE) untuk pengembangan aplikasi Android. Pada tahun 2013 Google mengumumkan bahwa Android Studio menjadi IDE resmi untuk mengembangkan aplikasi android.

Android Studio menawarkan fitur lebih banyak untuk meningkatkan produktivitas Anda saat membuat aplikasi Android, di antaranya :

- o Sistem versi berbasis Gradle yang fleksibel
- o Emulator yang cepat dan kaya fitur
- o Lingkungan yang menyatu untuk pengembangan bagi semua perangkat Android

Persyaratan Hardware

Spesifikasi hardware minimum yang bisa digunakan untuk menjalankan Android Studio dikutip dari web resmi android yaitu :

1. Windows

- Microsoft® Windows® 7/8/10 (32/ 64-bit)
- RAM minimum 3 GB, RAM yang disarankan 8 GB; tambah 1 GB untuk Emulator Android
- Ruang disk minimum yang tersedia 2 GB, Disarankan 4 GB (500 MB untuk IDE + 1,5 GB untuk Android SDK dan gambar sistem emulator)
- Resolusi layar minimum 1280 x 800
- Untuk emulator akselerasi: sistem operasi 64-bit dan prosesor Intel® dengan dukungan untuk Intel® VT-x, Intel® EM64T(Intel® 64), dan fungsionalitas Execute Disable (XD) Bit

2. Mac Os

- Mac® OS X® 10.10 (Yosemite) atau lebih tinggi, hingga 10.12 (macOS Sierra)
- RAM minimum 3 GB, RAM yang disarankan 8 GB; tambah 1 GB untuk Emulator Android
- Ruang disk minimum yang tersedia 2 GB, Disarankan 4 GB (500 MB untuk IDE + 1,5 GB

untuk Android SDK dan gambar sistem emulator)

- Resolusi layar minimum 1280 x 800

3. Linux

- Desktop GNOME atau KDE

Telah diuji pada Ubuntu® 12.04, Precise Pangolin (distribusi 64-bit yang mampu menjalankan aplikasi 32-bit).

- distribusi 64-bit yang mampu menjalankan aplikasi 32-bit
- GNU C Library (glibc) 2.19 atau yang lebih baru
- RAM minimum 3 GB, RAM yang disarankan 8 GB; tambah 1 GB untuk Emulator Android
- Ruang disk minimum yang tersedia 2 GB, Disarankan 4 GB (500 MB untuk IDE + 1,5 GB untuk Android SDK dan gambar sistem emulator)
- Resolusi layar minimum 1280 x 800
- Untuk emulator akselerasi: Prosesor Intel® dengan dukungan untuk Intel® VT-x, Intel® EM64T(Intel® 64), dan fungsionalitas Execute

Disable (XD) Bit, atau prosesor AMD dengan dukungan untuk AMD Virtualization™ (AMD-V™)

Menginstall Android Studio

Berikut ini langkah-langkah menginstall Android Studio⁶ :

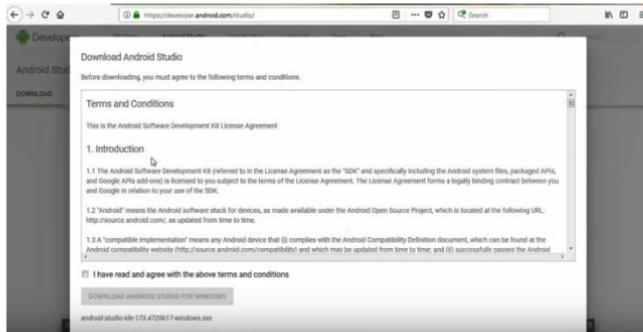
- Unduh android studio pada link :

<https://developer.android.com/studio/?hl=id>



- Klik **download android studio**, centang **I have read and agree with the above terms and conditions** lalu klik download.

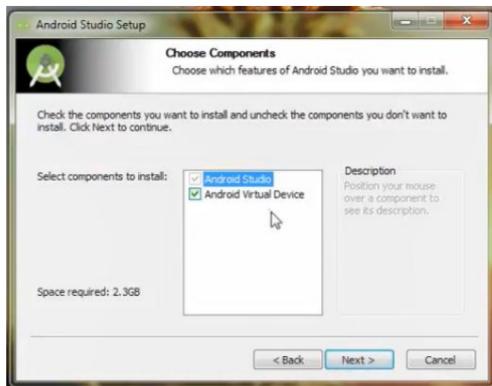
⁶ Versi android studio yang diinstall pada ebook ini yaitu versi 3.1.2. Dalam 1 tahun biasanya terdapat beberapa kali update terbaru dari android studio.



- Jika download selesai, **buka file exanya** yang sudah tersimpan pada komputer. Tunggu sampai **proses verifying** selesai
- **Klik next** setelah tampilan Install Android Studio setup muncul.



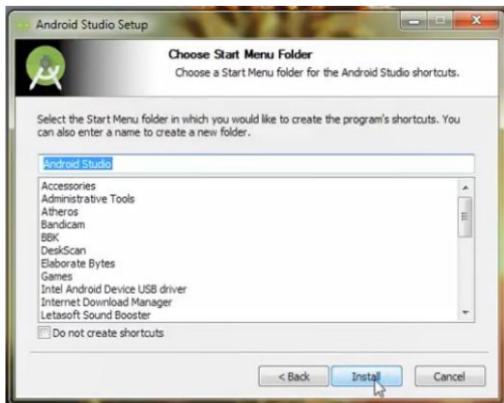
- **Klik next** untuk memasang semua komponen dari Android Studio



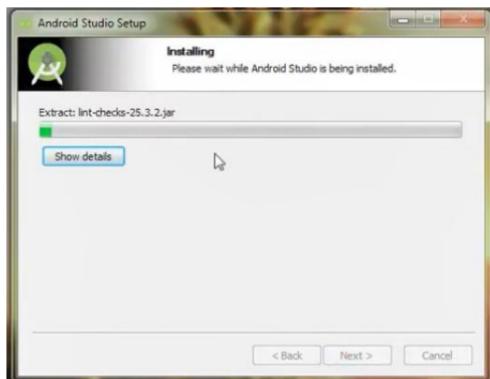
- Tentukan lokasi penyimpanan Android Studio dan sdknya. **Defaultnya tersimpan di Local Disk (C)** lalu **klik next**



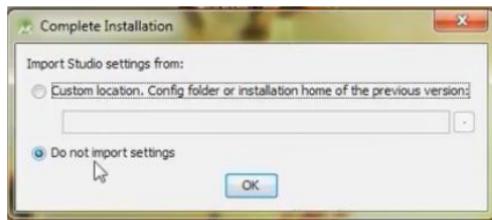
- **Klik Install**



- Tunggu sampai proses penginstallan selesai, **klik next** dan **finish**.



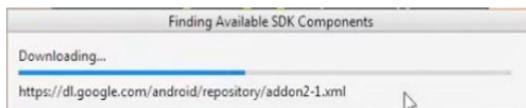
- Kemudian akan ada dua pilihan seperti gambar dibawah ini, karena baru pertama menginstall IDE Android Studio maka klik **Do not import settings**.



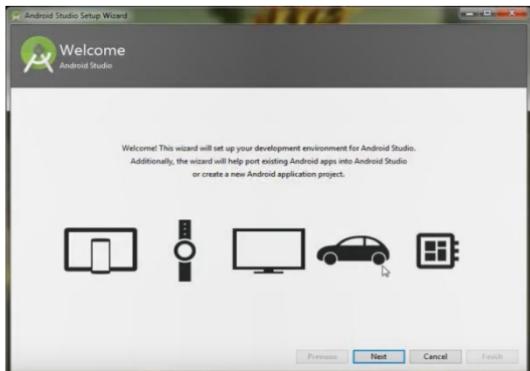
- Maka android studio akan terbuka



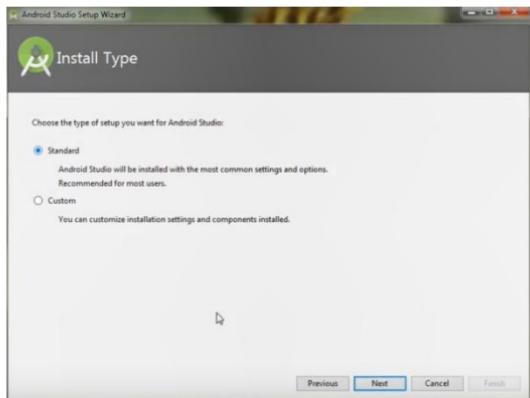
- Tunggu sampai proses download komponen SDK terdownload sampai selesai.



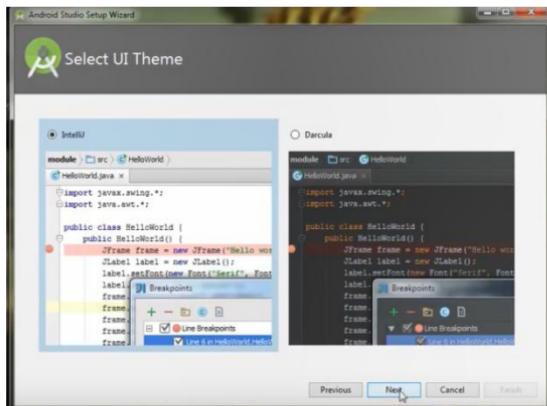
- Klik next saat Android Studio Setup Wizard terbuka



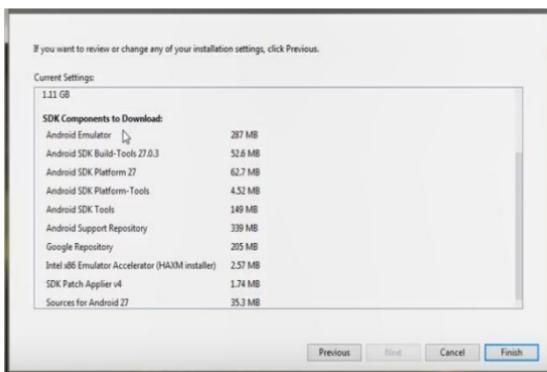
- Pilih standard lalu next



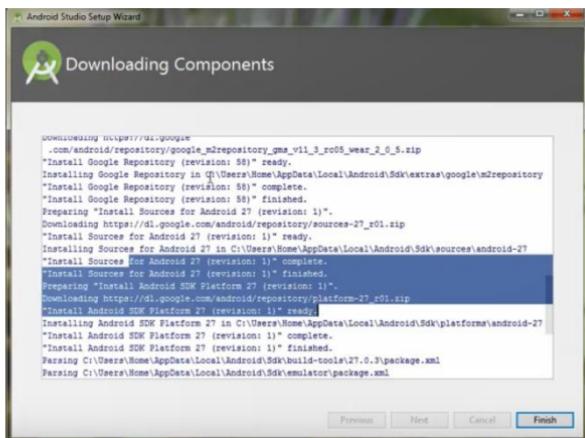
- Pilih tema android studio lalu **next**



- Klik **finish**, pastikan koneksi internetnya bagus. Krena proses ini akan memakan waktu yang cukup lama.



- Tunggu sampai proses download selesai, lalu klik finish.



- Setelah itu akan terbuka jendela awal android studio



Yuhu !! Proses install android studio telah selesai.

BAB 3

MEMBUAT PROJECT ANDROID PERTAMA



Membuat Project Hello World

Langsung saja kita akan membuat project android pertama menggunakan android studio.

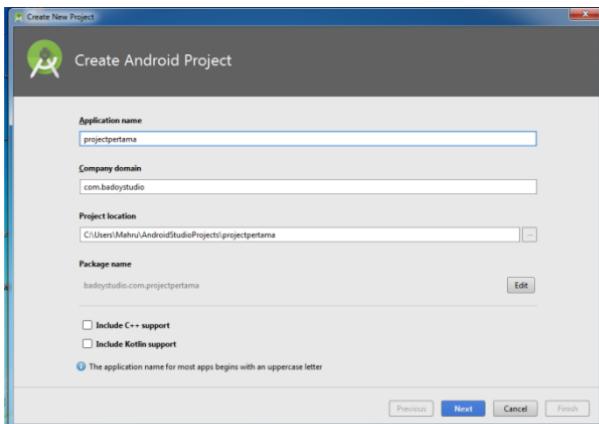
Adapun Langkah-langkahnya adalah :

- Buka android studio pilih **start a new android studio project**

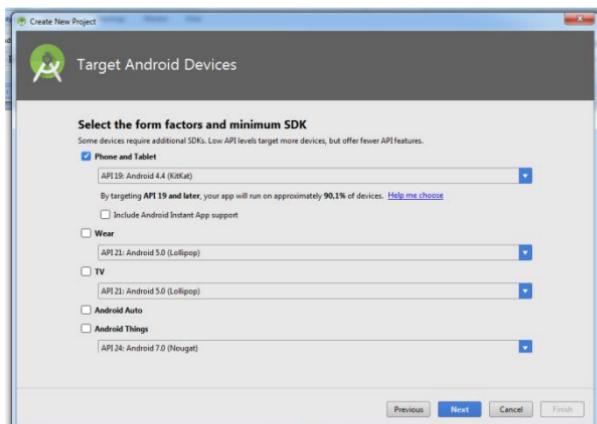


- Kemudian isikan **application name** (nama aplikasi android yang akan kita buat) dan **company domain** (nama developer/perusahaan pengembang aplikasi).
- Silahkan isi sesuai dengan keinginan. Untuk **project location** akan terisi secara otomatis. **Project location**

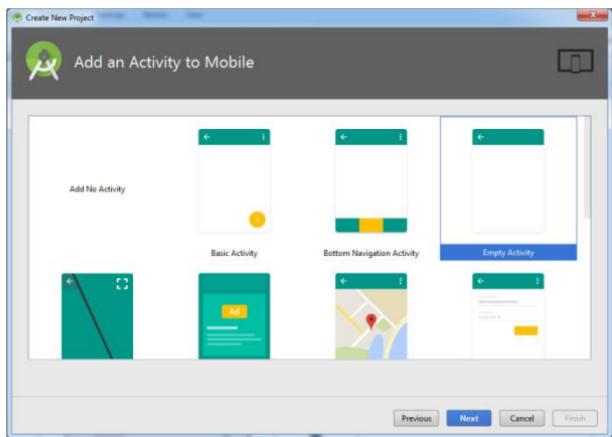
ini akan menunjukan project android yang kita buat tersimpan di folder apa. Kemudian **klik next**.



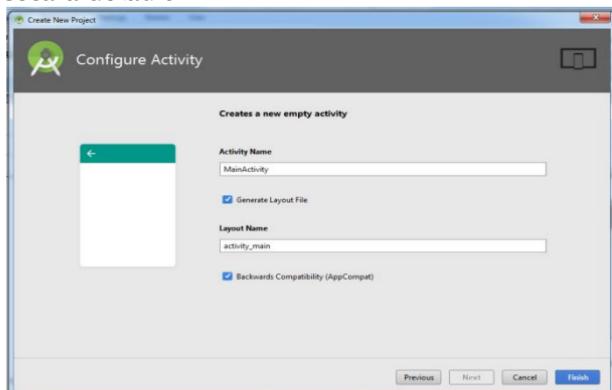
- Selanjutnya akan terbuka jendela **Target Android Devices**.



- Pilih **Empty Activity** pada jendela **Add an activity on mobile**

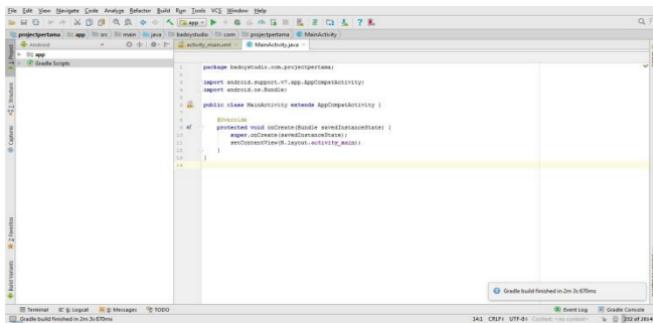


- Untuk **activity name** dan **layout name** biarkan terisi secara default.



- Tunggu sampai **workspace android** studio terbuka.⁷

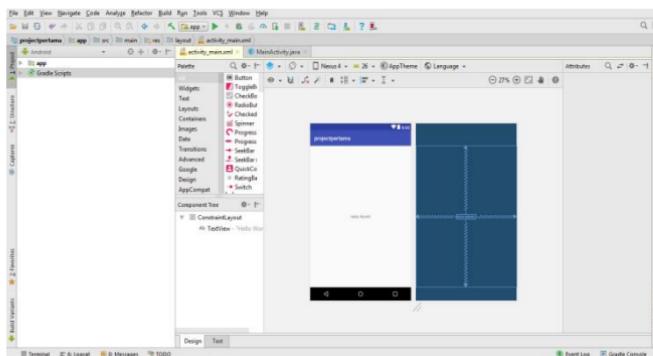
Tampilannya seperti gambar di bawah ini.



- Saat pertama workspace android terbuka terdapat dua tab **activity_main.xml** dan **MainActivity.java**. **activity_main.xml** merupakan layout utama aplikasi android.
- Disinilah kita akan mengedit semua tampilan aplikasi android. Sedangkan **MainActivity.java** digunakan untuk memberikan aksi (kodingan java) terhadap komponen-komponen yang sudah diberikan pada **activity_main.xml**.

⁷ Lama atau tidaknya proses ini disesuaikan dengan kapasitas ram yang digunakan oleh laptop atau komputer

- Contohnya kita memberikan memasukan tombol pada **activity_main.xml** agar tombol bisa melakukan suatu aksi kodingannya terdapat pada **MainActivity.java**.
- Sekarang pilih layout **activity_main.xml**. Secara default pada layout ini terdapat komponen **TextView** “Hello World”.



- Kita akan merunning project pertama ini pada bab selanjutnya.

BAB 4

RUNNING PROJECT MENGGUNAKAN AVD

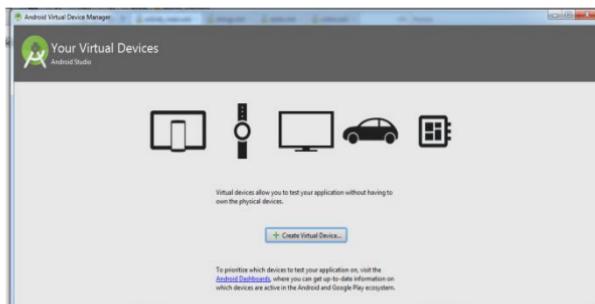


Android Virtual Device (AVD)

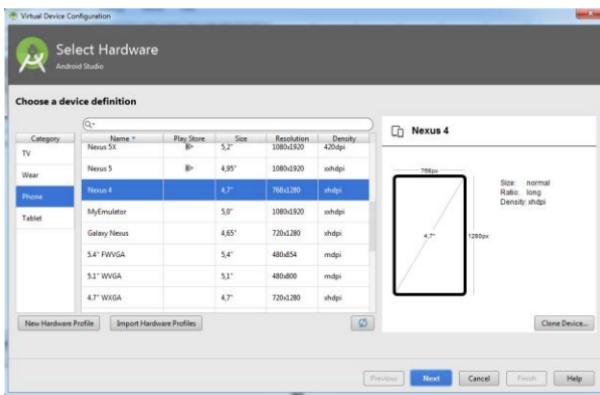
Android merupakan emulator bawaan android studio. Disinilah kita akan menjalankan (merunning) aplikasi android yang kita buat menggunakan android studio.

Berikut ini adalah langkah-langkahnya untuk membuat Android Virtual Manager yaitu :

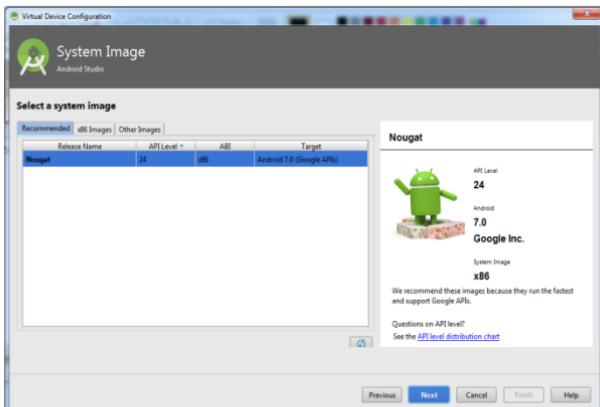
- Klik icon **AVD Manager**
- **Create Virtual Device**



- Pilih ukuran device yang akan digunakan untuk menjalankan aplikasi android. Di buku ini kita akan menggunakan **Nexus 4**



- Klik **Next** > pilih system image (ditutorial ini menggunakan nougat)> Klik **Next**

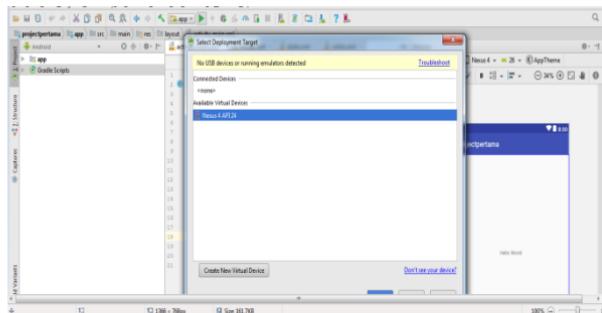


- Isi **Avd Name** > Klik **Show Advanced Settings** > Scroll kebawah. Klik **Finish**

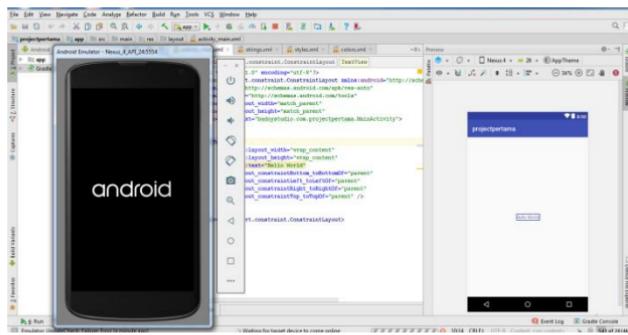
- Maka **AVD** sudah berhasil dibuat



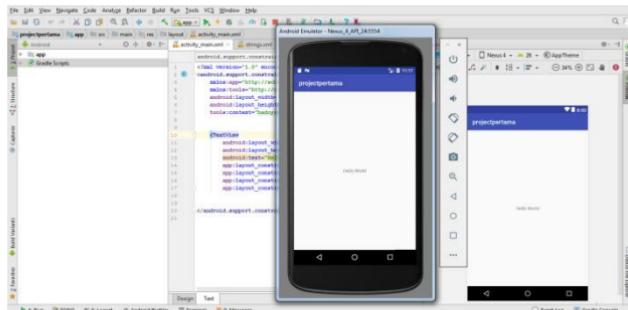
- Untuk merunning project silahkan klik icon run app



- Pilih **emulator yang sudah dibuat tadi (Nexus 4)**
- Tunggu beberapa saat hingga muncul tampilan seperti gambar dibawah ini



- Maka hasil akhir project yang kita running adalah sebagai berikut



Android Virtual Manager sudah berhasil dibuat dan project androidpun dapat berjalan dengan baik pada emulator bawaan android studio tersebut.

BAB 5

INSTALL EMULATOR GENYMOTION



Genymotion for personal use

Loading plugins....

Genymotion

Spesifikasi laptop yang pas-pasan untuk menjalankan android studio akan membuat proses running aplikasi menjadi lemot apabila menggunakan emulator (android virtual device) bawaan android studio.

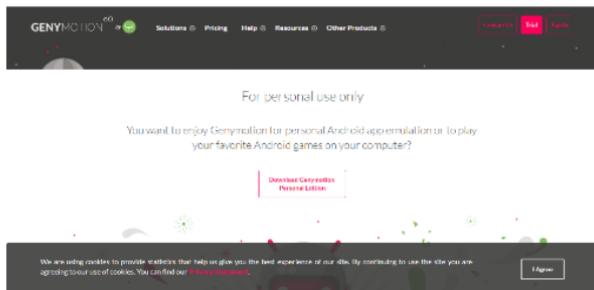


Maka Genymotion merupakan salah satu alternatif emulator yang bisa kita gunakan untuk menjalankan aplikasi android. Ditutorial ini kita akan belajar cara install genymotion pada windows 7 kemudian mengkonfigurasinya di Android Studio.

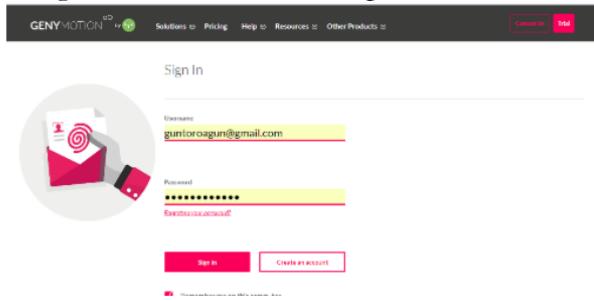
Install Genymotion

Langsung saja kita lakukan installasi genymotion :

- Langkah pertama adalah donwload aplikasi genymotion dari situs resminya. Klik tautan <http://genymotion.com>



- Jika sudah silahkan buat akun baru dengan cara mengisikan alamat email dan password.



- Login menggunakan akun yang sudah dibuat, proses selanjutnya adalah donwload aplikasi genymotion for

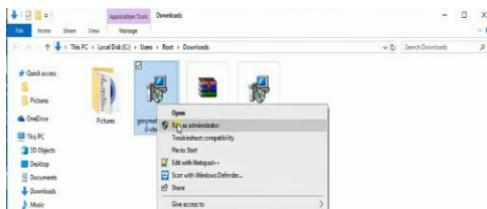
windows (untuk komputer yang menggunakan os windows).



Download Genymotion 2.12.0



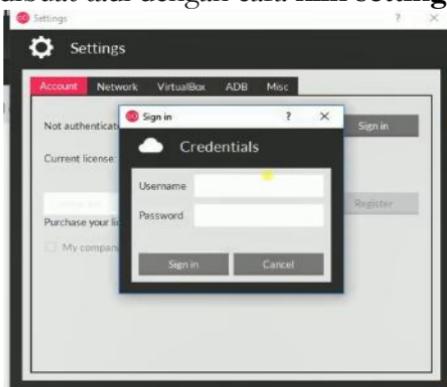
- Cari **file exe genymotion** yang sudah di download tadi. Buka **file exe**.



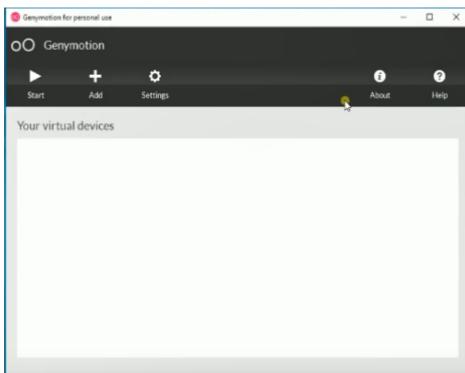
- Lakukan **installasi genymotion** sampai dengan selesai.



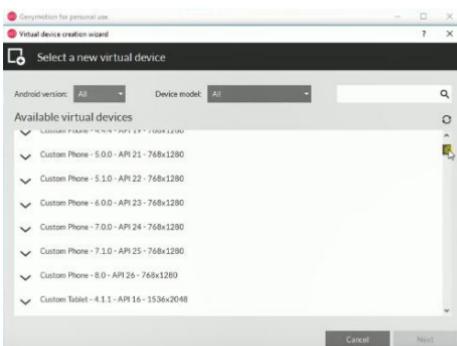
- **Tekan finish** maka proses Install sudah selesai. Selanjutnya Login menggunakan akun yang sudah dibuat tadi dengan cara **klik settings**.



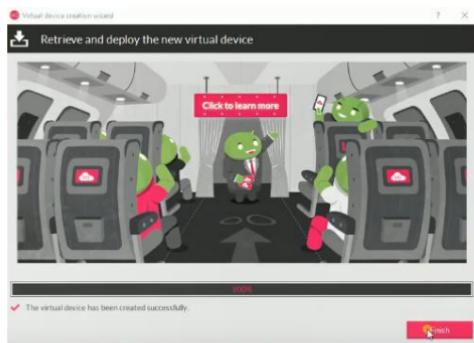
- **Pilih add** untuk membuat virtual device baru yang nantinya akan digunakan untuk merunning proect android kita.



- Pilih **Virtual Devicenya**



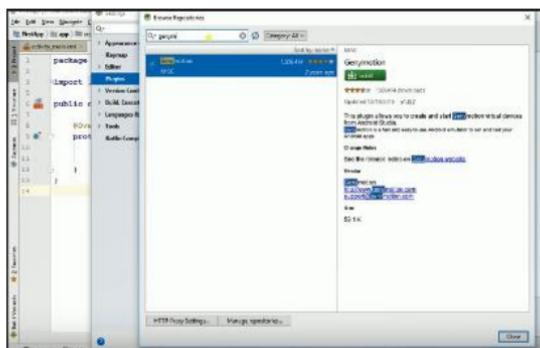
- Klik next sampai dengan virtual device terdownload.



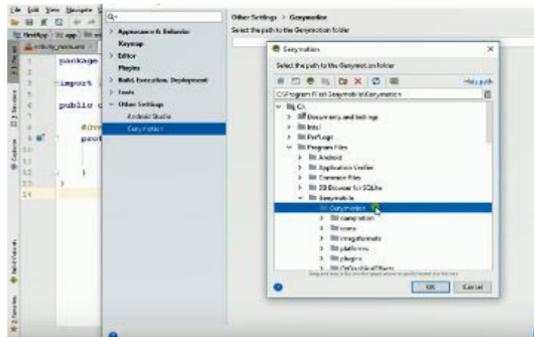
Konfigurasi Genymotion

Sekarang kita lakukan konfigurasi genymotion pada android studio. Caranya yaitu :

- Buka Android Studio >Klik file > setting > plugin > browse repositories > klik install



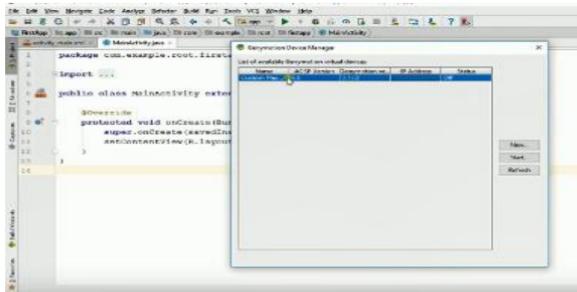
- Setelah itu klik **Restart android studio**
- Atur path (lokasi genymotion) dengan cara **Klik view > toolbar > Klik icon genymotion > Pilih path**



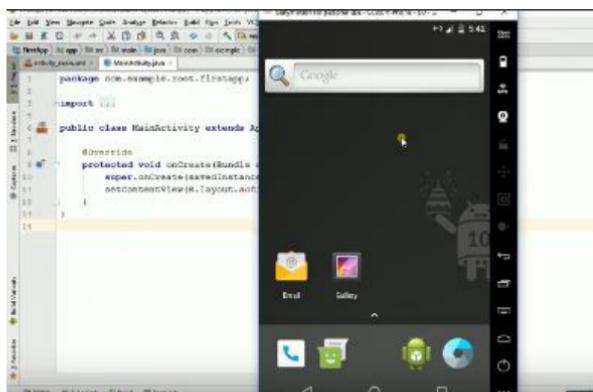
Running Project

Jalankan project android yang kita miliki menggunakan emulator genymotion.

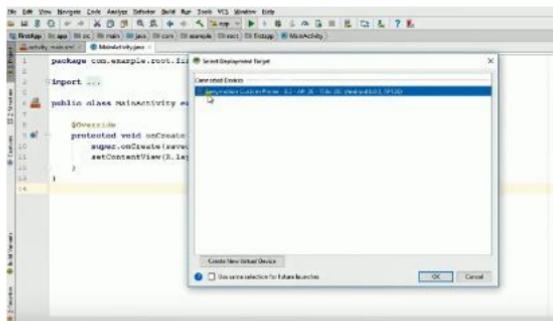
- Klik icon genymotion > pilih phone > start



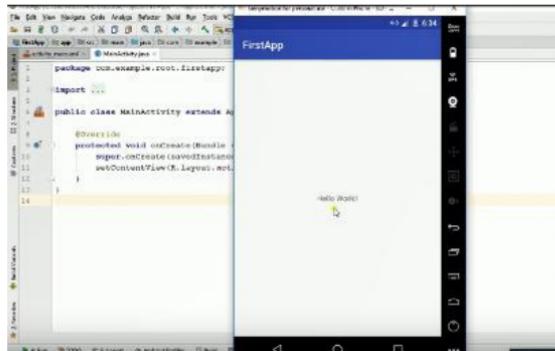
- Tunggu beberapa saat sampai emulator genymotion terbuka



- Jalankan project android dengan cara **klik icon running > pilih emulator genymotion > Ok**



- Maka hasil akhirnya seperti gambar dibawah ini.



The screenshot shows the Android Studio interface with the code editor open. The file is `MainActivity.java` under the package `com.example.root.firstapp`. The code contains the standard boilerplate for a new Activity:

```
1 package com.example.root.firstapp;
2
3 import android.os.Bundle;
4 import android.app.Activity;
5 import android.view.Menu;
6
7 public class MainActivity extends Activity {
8
9     @Override
10    protected void onCreate(Bundle savedInstanceState) {
11        super.onCreate(savedInstanceState);
12        setContentView(R.layout.activity_main);
13    }
14 }
```

To the right of the code editor is a preview pane showing a single line of text: "Hello World!". This indicates that the application's main screen is currently displaying the default "Hello World" message.

BAB 6

RUNNING PROJECT DI SMARTPHONE



Bisakah kita merunning aplikasi yang kita tanpa menggunakan emulator ? sangat bisa. Kali ini kita akan bahas cara merunning project atau aplikasi langsung di smartphone android yang kita miliki.

Mengaktifkan Mode Pengembang

Terlebih dahulu kita harus mengaktifkan mode pengembang pada smartphone android. Caranya :

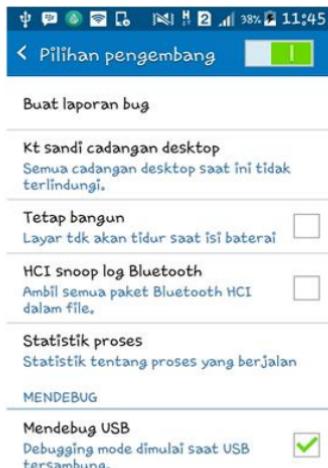
- **Klik pengaturan > Tentang Perangkat > ketuk 7 kali pada nomor versi (setelah itu biasanya ada notifikasi mode developer telah aktif).**



- Ketuk pilihan pengembang



- Buka mode pengembang lalu aktifkan > dan ceklis mendebug usb

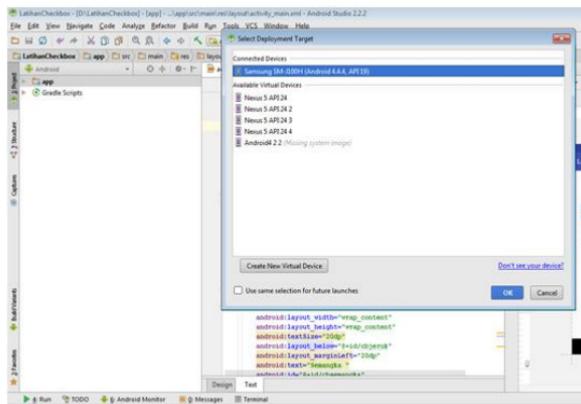


Running Project

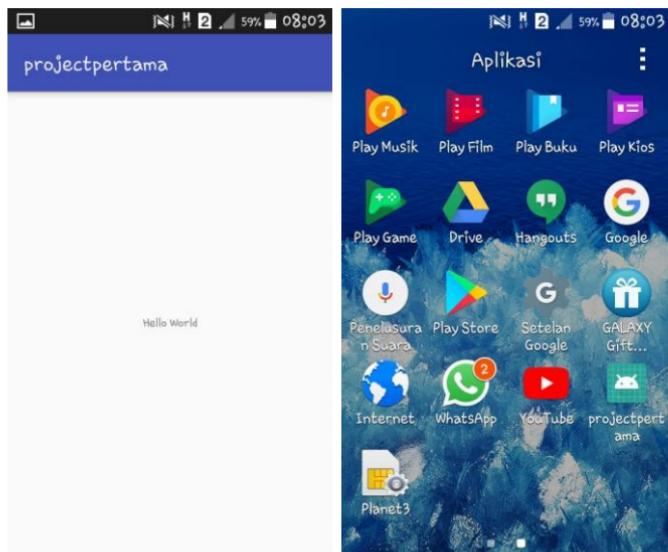
Jika sudah saatnya kita running aplikasi android yang sudah dibuat menggunakan android studio langsung pada smartphone android.

Adapun langkah-langkahnya adalah sebagai berikut :

- Hubungkan smartphone android dengan laptop menggunakan kabel usb
- Running aplikasi dengan **cara klik run pada android studio**, lalu pilih tipe smartphone android lalu **klik ok**



- Tunggu beberapa saat, aplikasi akan tampil seperti gambar dibawah ini :



Merunning aplikasi yang kita buat pada android studio menggunakan emulator genymotion atau langsung pada smartphone android kita merupakan alternative jika proses running menggunakan android virtual device dirasa sangat lambat.

BAB 7

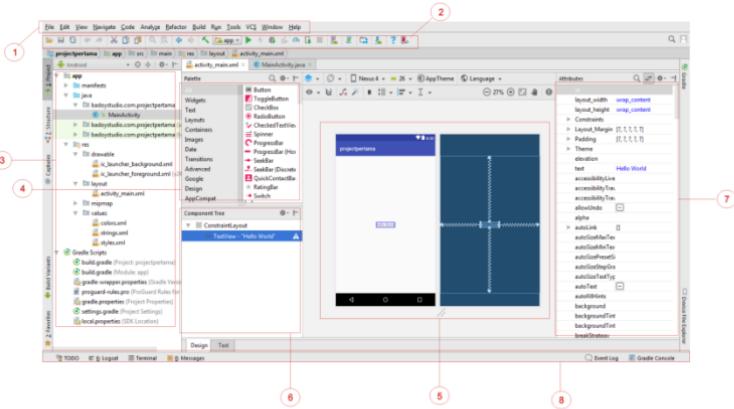
ANTARMUKA DAN STRUKTUR PROYEK ANDROID STUDIO



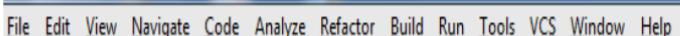
Antarmuka Android Studio

Agar lebih memudahkan saat membuat aplikasi android menggunakan android studio ada baiknya kita mempelajari antarmuka android studio.

Berikut ini merupakan bagian-bagian penting dari Antar Muka (Interface) Android Studio.



1. Menu Bar



Seperti pada aplikasi lain **Menu bar** merupakan bagian antar muka (interface) pengguna yang berisi perintah dan opsi yang dapat dipilih untuk mengeksekusi suatu perintah.

2. Tool Bar

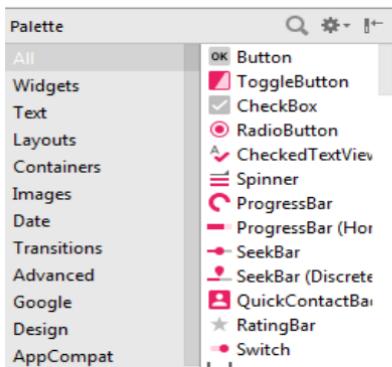


Dengan tool bar kita bisa mempercepat perintah pada sebuah aplikasi.

3. Struktur Proyek

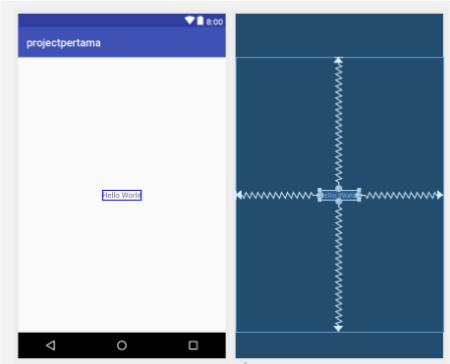
Selengkapnya dibahas pada bagian struktur proyek android.

4. Pallette



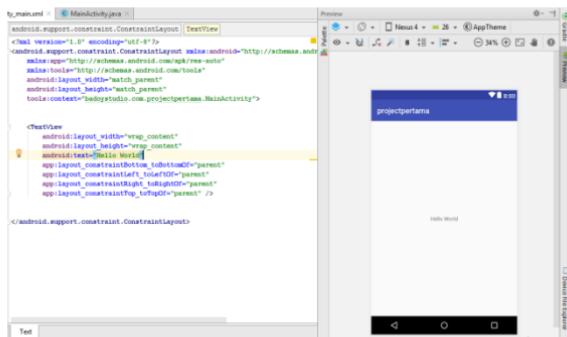
Di pallette tersedia semua tools untuk membuat aplikasi android dan enaknya lagi untuk menggunakannya cukup dengan **mendrag and drop ke design android**.

5. Design

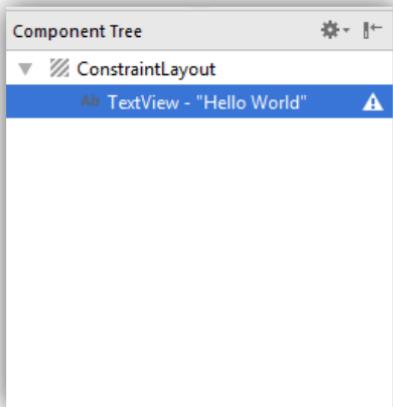


Tempat ini digunakan untuk mendesign layout aplikasi dengan cara **drag and drop** (tidak mengetikan script xml).

Untuk mengetikan script kode-kode xml dan java silahkan klik tab text.



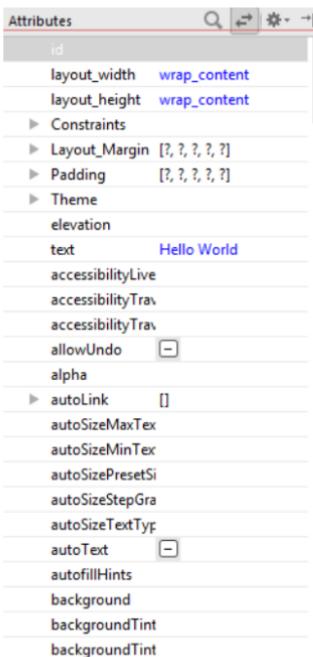
6. Component Tree



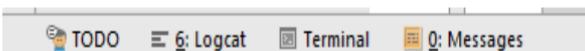
Setelah kita menggunakan tools yang ada di **pallete** maka akan ditampilkan pada **component tree** misal kita mendrag **TextView (widget untuk membuat teks)** ke design android. Maka nanti ditampilkan di **component tree TextView**.

7. Properties / Attributes

Disini akan ditampilkan pengaturan-pengaturan dari komponen yang digunakan untuk design aplikasi. Jika **TextView** maka akan ditampilkan untuk mengatur warna, size dan lainnya.



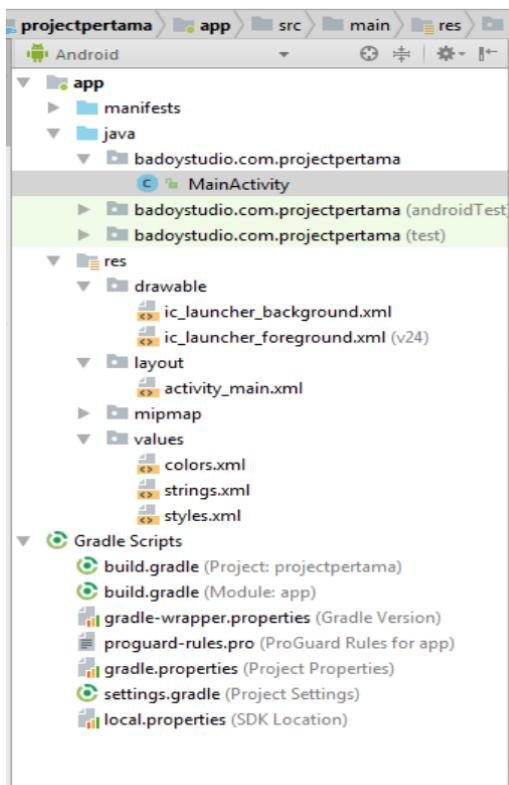
8. Status Bar



Menampilkan proses pada Android Studio. Proses Loading, Error, dan lainnya.

Struktur Proyek Android Studio

Semua komponen yang kita gunakan untuk membuat aplikasi android akan diletakan pada folder-folder proyek android studio.



Berikut struktur proyek yang ada pada android studio.

1. Manifest

Klik folder **Manifest** maka akan terdapat **AndroidManifest.xml**. File ini digunakan untuk melakukan beberapa pengaturan seperti :

- **Nama aplikasi**
- **Icon aplikasi**
- **Theme style**
- **User permission** (jika membuat aplikasi yang membutuhkan akses hardware smartphone ataupun koneksi internet).

Ketika kita buka file **AndroidManifest.xml**, maka kode-kode default yang terdapat pada file ini adalah :

```
<?xml version="1.0" encoding="utf-8"?>
<manifest
    xmlns:android="http://schemas.android.com/apk/res/android"
    package="badoystudio.com.projectpertama">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action
                    android:name="android.intent.action.MAIN" />
                <category
                    android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

2. Java

Di folder ini terdapat file **MainActivity.java** yang digunakan untuk memberikan perintah agar aplikasi melakukan sesuatu menggunakan bahasa pemrograman java.

Defaultnya **MainActivity.java** sebagai berikut :

```
package badoystudio.com.projectpertama;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;

public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

3. Res Drawable

Folder yang digunakan untuk memasukan gambar pendukung aplikasi yang kita buat baik berupa icon ataupun lainnya.

Cara memasukan gambar ke folder ini cukup mudah yaitu :

- Klik kanan pada folder **drawable**, pilih show in explorer, kemudian copykan gambar yang sudah kita siapkan di komputer pada folder **drawable**.

4. Res Layout

Disini terdapat file **activity_main.xml**. File ini digunakan untuk pengaturan layout untuk interface utama pada aplikasi android yang kita buat. Saat pertama kali dibuka biasnya file ini memiliki kode-kode default seperti :

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"

    tools:context="badoystudio.com.projectpertama.MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</android.support.constraint.ConstraintLayout>
```

5. Mipmap

Folder ini digunakan untuk memasukan gambar berupa icon. Icon default aplikasi yang kita buat juga diambil dari folder ini.

6. Values

- **Colors.xml**

File ini berisi kode-kode untuk pengaturan warna. Warna status bar, teks, ataupun lainnya. Defaultnya sebagai berikut.

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="colorPrimary">#3F51B5</color>
    <color name="colorPrimaryDark">#303F9F</color>
    <color name="colorAccent">#FF4081</color>
</resources>
```

- **Strings.xml**

Digunakan untuk pengaturan teks-teks aplikasi yang kita buat. Contoh teks yang digunakan untuk nama aplikasi seperti dibawah ini.

```
<resources>
    <string name="app_name">projectpertama</string>
</resources>
```

○ Styles.xml

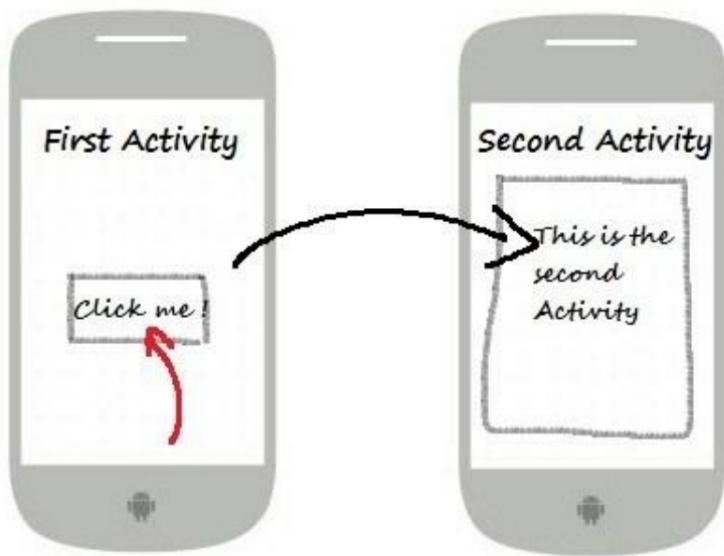
Digunakan untuk memberikan nama warna setelah kode-kode warna dimasukan atau disetting pada **color.xml**. File ini juga bisa disebut theme dasar dari aplikasi android.

```
<resources>
    <!-- Base application theme. -->
    <style name="AppTheme"
        parent="Theme.AppCompat.Light.DarkActionBar">
        <!-- Customize your theme here. -->
        <item
            name="colorPrimary">@color/colorPrimary</item>
        <item
            name="colorPrimaryDark">@color/colorPrimaryDark</item>
        <item
            name="colorAccent">@color/colorAccent</item>
    </style>
</resources>
```

Itulah interface dan struktur folder project pada Android Studio. Jika kita sudah memahaminya maka pengembangan aplikasi androidpun akan lebih mudah.

BAB 8

ACTIVITY DAN SIKLUS HIDUP



Activity

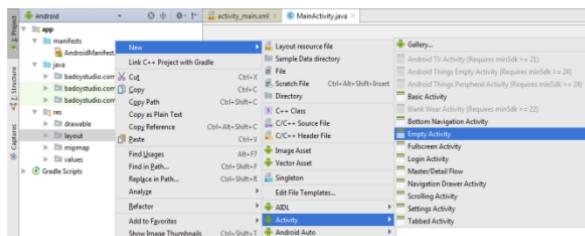
Activity (aktivitas) adalah sebuah komponen aplikasi yang menyediakan layar yang digunakan pengguna untuk berinteraksi guna melakukan sesuatu, misalnya memilih nomor ponsel, mengambil foto, mengirim email, atau menampilkan peta.

Tiap Activity (aktivitas) diberi sebuah jendela untuk menggambar antarmuka penggunanya. Satu aplikasi bisa memiliki lebih dari satu Activity, dan setiap Activity memiliki siklus hidup.

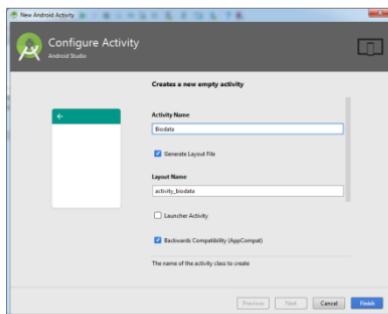
Membuat Activity Baru

Saat kita membuat activity baru maka nanti akan ada file java dan xml. Untuk membuat activity baru caranya sangat mudah yaitu :

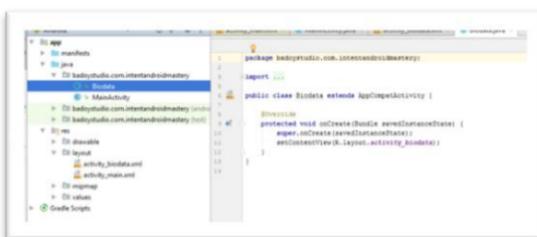
- **Klik kanan layout** yang ada pada folder res
- **pilih new lalu cari activity**
- kemudian **pilih empty activity**



- Lalu isikan **nama activity**. Contoh disini di isi dengan menggunakan nama Biodata. **Tekan finish**.



- Sekarang lihat maka sudah ada dua file baru yaitu **Biodata.java** dan **activity_biodata.xml**.



Untuk penggunaan activity secara maksimal akan dibahas di bab intent.

Siklus Hidup

Siklus hidup aplikasi Android diatur oleh sistem, berdasarkan kebutuhan pengguna, sumber daya yang tersedia, dan sebagainya.

Adapun siklus hidup android yaitu :

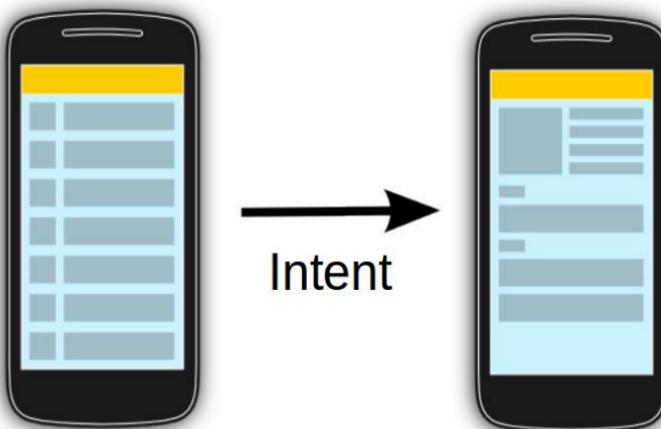
No	Method	Keterangan
1	onCreate()	Method ini dipanggil saat activity pertama kali dibuat
2	onStart()	Method ini dipanggil saat sebuah activity tampil ke pengguna
3	onRestart()	Method ini dipanggil ketika activity yang berjalan pada saat itu dihentikan sementara (paused) dan activity sebelumnya dijalankan kembali(resume). (Hasil dari method OnRestart())
4	onFreeze()	Method ini dipanggil ketika activity berada dalam

		keadaan freeze (tidak merespon) akibat sedang sibuk mengerjakan tugas tertentu pada sistem
5	onPause()	Method ini dipanggil ketika activity di hentikan sementara (pause) dan berikutnya ketika dijalankan kembali akan berada dalam posisi resume dan memanggil method OnResume()
6	onStop()	Method ini dipanggil ketika activity tidak lagi tampak kepada pengguna
7	onRestart()	Method ini dipanggil ketika acitivity dijalankan kembali setelah state Stop
8	onDestroy()	Method ini dipanggil sebelum activity dihancurkan (destroy) oleh sistem (baik

		secara manual maupun untuk kepentingan pelonggaran memori.
--	--	--

BAB 9

PENGERTIAN, FUNGSI, DAN TIPE INTENT



Intent

Dikutip dari website resmi developer android, intent merupakan objek yang menyediakan waktu proses yang mengikat komponen-komponen terpisah. Intent biasanya digunakan untuk berpindah ke activity lain pada aplikasi android.

Agak rumit dipahami memang mudahnya intent adalah cara untuk memberikan *Action* pada suatu komponen dalam satu aplikasi yang sama maupun aplikasi lain. Jika diweb kita mengenalnya dengan isitlah *Hyperlink*.

Fungsi Intent

- o Untuk berpindah halaman dari satu *Activity* ke *Activity* lain. Contohnya : Kita membuat aplikasi dengan 3 Activity. Activity utama berisi tombol informasi dan login. Ketika tombol informasi di klik akan tampil activity informasi begitupun dengan tombol login ketika di klik akan menampilkan activity login.

- Untuk transfer data dari satu **Activity** ke **Activity** lain. Contohnya : Pada saat kita mengisi data pada Activity login maka nanti akan ditampilkan informasinya pada Activity infomasi user.
- Untuk memanggil activity pada internal android seperti Melakukan Dial Number, Open Email ataupun lainnya.

Implicit Intent

Implicit Intent adalah intent yang digunakan untuk memanggil fungsi *Activity* yang sudah ada di fungsi internal android. Seperti membuka browser, melakukan panggilan dan lainnya.

Langsung saja kita praktekan cara penggunaan intent pada android studio.

- **Buat project baru** dengan nama “ belajar intent “ atau disesuaikan keinginan sendiri

- Lalu atur layout pada `activity_main.xml` menggunakan kode-kode dibawah ini

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/activity_main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="20dp"
    tools:context=".MainActivity">

    <Button
        android:text="TELEPON"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:id="@+id/btnpgl"
        android:onClick="panggil"
    />

    <Button
        android:text="BUKA BROWSER "
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/btnpgl"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="11dp"
        android:id="@+id/btnbuka"
        android:onClick="buka"/>

</RelativeLayout>
```

- Jika ada error biarkan saja. Karena kita belum menambahkan kodingan javanya pada `MainActivity.java`.

- Hasil dari layout yang kita buat menggunakan kode di atas menjadi seperti ini.



- **Buka MainActivity.java** perhatikan sebentar ada **nama packagenya**. Contoh dari nama package adalah :

```
package badoystudio.com.intentandroidmastery;
```

Nama package tersebut terbentuk secara otomatis dari nama project yang kita buat di awal. Karena saya menggunakan nama project intentandroidmastery maka hasilnya seperti itu.

Jika anda menggunakan nama project yang berbeda tidak masalah. Jadi untuk kode-kode javanya tempatkan dibawah nama package.

- Sekarang tambahkan kode-kode java berikut pada **MainActivity.java**. Jangan lupa sesuaikan nama packagenya dengan nama project yang anda buat tadi.

```
package badoystudio.com.intentandroidmastery;

import android.content.Intent;
import android.content.pm.PackageManager;
import android.net.Uri;
import android.support.v4.app.ActivityCompat;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;

public class MainActivity extends AppCompatActivity {

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

    public void panggil(View view) {
        String nomor = "09667347" ;
        Intent panggil = new Intent(Intent.
        ACTION_DIAL);
        panggil.setData(Uri.
        fromParts("tel",nomor,null));
        startActivity(panggil);
    }

    public void buka(View view) {
        String url = "https://badoystudio.com" ;
        Intent bukabrowser = new Intent(Intent.
        ACTION_VIEW);
        bukabrowser.setData(Uri.parse(url));
        startActivity(bukabrowser);
    }
}
```

- Penjelasan kode di atas :

- Untuk **melakukan panggilan** menggunakan kode-kode java :

```
public void panggil(View view) {  
    String nomor = "09667347" ;  
    Intent panggil = new Intent(Intent.  
ACTION_DIAL);  
    panggil.setData(Uri.  
fromParts("tel",nomor,null));  
    startActivity(panggil);  
}
```

- Untuk **membuka browser** menggunakan kode-kode java :

```
public void buka(View view) {  
    String url = "https://badoystudio.com" ;  
    Intent bukabrowser = new Intent(Intent.  
ACTION_VIEW);  
    bukabrowser.setData(Uri.parse(url));  
    startActivity(bukabrowser);  
}
```

- Karena kita akan menggunakan fasilitas call dan internet maka tambahkan user permission pada **AndroidManifest.xml** seperti berikut :

```
<?xml version="1.0" encoding="utf-8"?>
<manifest
    xmlns:android="http://schemas.android.com/apk/res/android"

    package="badoystudio.com.intentandroidmastery">

        <uses-permission
            android:name="android.permission.CALL_PHONE" />
        <uses-permission
            android:name="android.permission.INTERNET"/>

        <application
            android:allowBackup="true"
            android:icon="@mipmap/ic_launcher"
            android:label="@string/app_name"

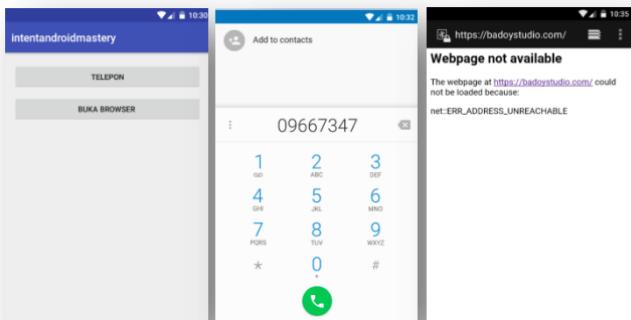
            android:roundIcon="@mipmap/ic_launcher_round"
            android:supportsRtl="true"
            android:theme="@style/AppTheme">
            <activity android:name=".MainActivity">
                <intent-filter>
                    <action
                        android:name="android.intent.action.MAIN" />

                    <category
                        android:name="android.intent.category.LAUNCHER"
                    />
                </intent-filter>
            </activity>
        </application>

    </manifest>
```

- Terakhir kita running aplikasi menggunakan android virtual device, genymotion, atau langsung melalui hape android yang anda miliki.

- Hasilnya adalah seperti gambar di bawah ini. Saat tombol telepon ditekan akan menampilkan gambar dial up, begitupun saat tombol buka browser ditekan default browser akan terbuka secara otomatis menuju halaman yang kita tuju.



Explicit Intent

Explicit intent digunakan untuk memanggil Activity lain yang masih dalam satu project ataupun berbeda.

Langsung saja kita praktikan bagaimana menggunakan **Explicit Intent** dengan menggunakan project Implicit intent.

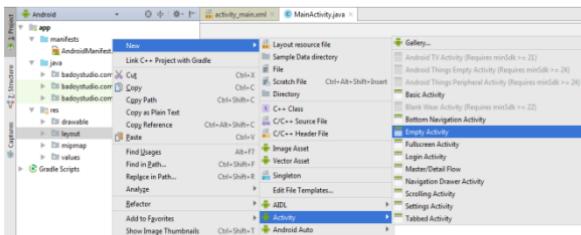
- Tambahkan satu button pada `activity_main.xml` dibawah button yang sudah dibuat pada implicit intent.

```
<Button  
    android:text="TENTANG APLIKASI"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_below="@+id/btnbuka"  
    android:layout_centerHorizontal="true"  
    android:layout_marginTop="22dp"  
    android:id="@+id/btninfo"  
    android:onClick="tentang"/>
```

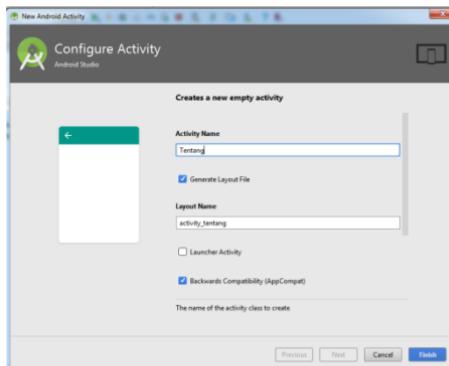
- Pastikan kode untuk menambahkan button di atas diletakan dibawah setelah kode button buka browser.

```
<Button  
    android:text="BUKA BROWSER "  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_below="@+id/btnpgl"  
    android:layout_centerHorizontal="true"  
    android:layout_marginTop="11dp"  
    android:id="@+id/btnbuka"  
    android:onClick="buka"/>
```

- Buat activity baru yang berisi informasi tentang aplikasi yang kita buat. Caranya : **klik folder java > klik kanan pada nama project > pilih New > Activity > Empty Activity**



- Beri nama tentang pada name activity dan **klik finish.**



- Tambahkan informasi aplikasi dengan cara **Klik res > Layout > activity_tentang.xml > tab text >** tambahkan kode-kode berikut :

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/activity_tentang"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".Tentang">

    <TextView
        android:text="ini adalah aplikasi belajar intent"
        android:textSize="20sp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="31dp"
        android:id="@+id/textView" />
</RelativeLayout>

```

- Pada **MainActivity.java** tambahkan intent untuk membuka halaman tentang yang tadi kita buat. Kodenya adalah :

```

public void tentang(View view) {
    Intent tentang= new Intent(MainActivity.this,
    Tentang.class);
    startActivity(tentang);
}

```

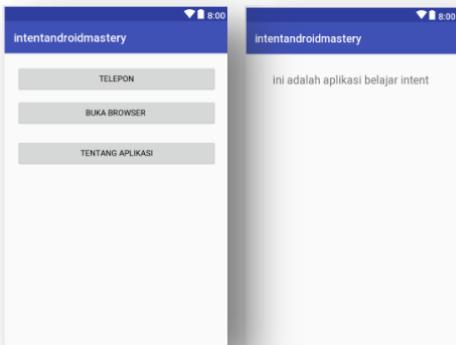
Tempatkan kode tersebut di atas dibawah kode

```

public void buka(View view) {
    String url = "https://badoystudio.com";
    Intent bukabrowser = new Intent(Intent.
ACTION_VIEW);
    bukabrowser.setData(Uri.parse(url));
    startActivity(bukabrowser);
}

```

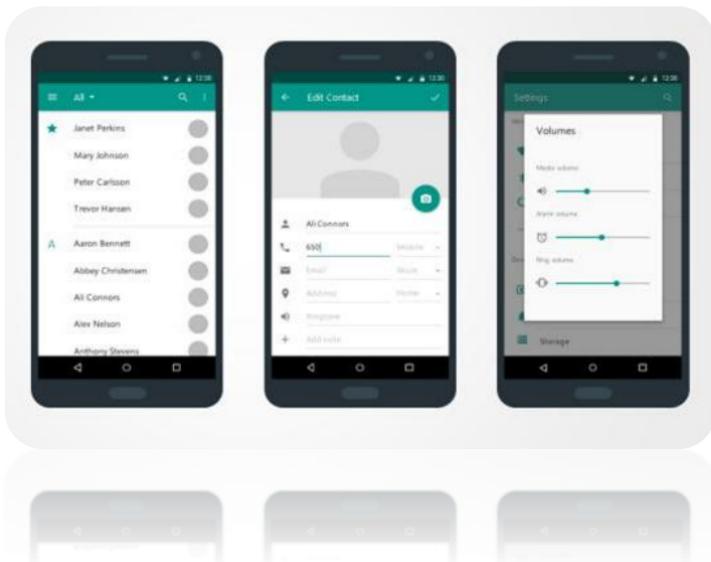
- Jalankan aplikasi maka akan tampil seperti gambar berikut ketika tombol tentang aplikasi di tekan.



Itulah penjelasan dan cara penggunaan intent. Cukup mudah bukan ? Silahkan dipraktekan.

BAB 10

KOMPONEN VIEW



User Interface

Menurut Wikipedia **User interface** (antar muka) merupakan bentuk tampilan grafis yang berhubungan langsung dengan pengguna.



User interface aplikasi android akan mempengaruhi tingkat kemudahan menggunakan aplikasi tersebut. Ketika kita menggunakan sebuah aplikasi biasanya yang pertama kali dilihat yaitu tampilan aplikasinya.

Baik itu tata letak tombol, gambar, teks dan lainnya. user interface yang menarik dan simple akan lebih disukai pengguna dibandingkan dengan user interface yang terlihat rumit.

Pembuatan User Interface pada aplikasi android dibagi menjadi dua yaitu :

1. Procedural

UI procedural yaitu menggunakan bahasa pemrograman java.

2. Declarative

UI declarative yaitu menggunakan bahasa xml.

Agar lebih mudah dipahami pembahasan user interface kali ini lebih menekankan menggunakan bahasa xml. Pada android terdapat dua jenis user interface yaitu :

✓ **View :**

Merupakan komponen yang ditampilkan pada layar yang biasanya bisa digunakan untuk berinteraksi dengan pengguna.

✓ **ViewGroups :**

Merupakan komponen yang berperan mengatur komponen-komponen view lainnya.

View

Di bab ini kita akan membahas beberapa komponen view yang sering digunakan dalam membuat user interface aplikasi android.

Komponen view yang akan dibahas adalah :

1. **TextView**
2. **Button**
3. **ImageView**
4. **EditText**
5. **Toast**

TextView

TextView digunakan untuk pengaturan font/ teks baik kalimat ataupun kata.

Dalam latihan ini kita akan melakukan beberapa pengaturan diantaranya : jenis font, ukuran font, ketebalan font, perataan dan memberikan warna pada font ataupun teks.

Langsung saja kita praktikan menggunakan TextView pada aplikasi android yang kita buat.

- Buat project baru dengan nama “**TextViewLatihan**”
- Atur target SDK dan lainnya tunggu sampai workspace android studio terbuka
- Untuk menggunakan TextView kita bisa mengaplikasikan atribut-atribut penting yang sering digunakan pada TextView diantaranya sebagai berikut :

1. Id

Atribut ini digunakan untuk memberikan nama pada TextView. Biasanya pada aplikasi ada beberapa teks yang ingin di tampilkan dalam satu activity.

Agar tidak terjadi error maka kita harus memberikan nama unik atau id pada masing-masing teks tersebut. Dengan menggunakan id bisa memudahkan kita saat mengatur layout untuk teks yang akan ditampilkan di aplikasi.

```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="TEKS 1"  
    android:id="@+id/teks1"/>  
  
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="TEKS 2"  
    android:id="@+id/teks2"  
    android:layout_below="@+id/teks1"/>
```

Maka hasilnya :



Tampilan teks 2 berada dibawah teks 1 karena menggunakan layout below pada teks yang ke 2.

2. Text

Untuk menampilkan teks pada aplikasi kita membutuhkan atribut ini.

```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="digunakan untuk menampilkan teks "  
    android:id="@+id/teks1"/>
```

Preview :



Kalimat ” digunakan untuk menampilkan teks ” berada pada atribut ” android:text ”.

3. Text Color

Warna teks sangat berpengaruh pada tampilan UI dan UX pada aplikasi android. Untuk memberikan warna teks kita menggunakan atribut ini.

```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="teks menggunakan warna primary  
color "  
    android:textColor="@color/colorPrimaryDark"  
    android:id="@+id/teks1"/>  
  
<TextView  
    android:paddingTop="20dp"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="teks menggunakan warna primary  
color "  
    android:textColor="@color/colorAccent"  
    android:layout_below="@+id/teks1"  
    android:id="@+id/teks2"/>
```

Preview :



Untuk warna sendiri kita bisa mengaturnya pada color.xml atau juga langsung dengan cara memasukan kode-kode warna seperti : #ffff dan lainnya.

4. Text Size

Atribut ini digunakan untuk memberikan ukuran pada teks yang ingin kita tampilkan pada aplikasi.

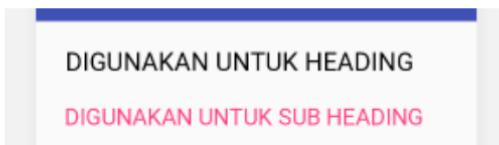
```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="MENGGUNAKAN UKURAN 14SP"  
    android:textSize="14sp"  
    android:id="@+id/teks1"/>  
  
<TextView  
    android:paddingTop="20dp"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="MENGGUNAKAN UKURAN 20SP"  
    android:textColor="@color/colorAccent"  
    android:layout_below="@+id/teks1"  
    android:id="@+id/teks2"  
    android:textSize="20sp"/>
```

5. Text Appearance

Ini digunakan untuk mengubah ukuran teks yang digunakan untuk **heading** atau **sub heading**.

```
<TextView  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:text=" DIGUNAKAN UNTUK HEADING "  
    android:typeface="serif"  
  
    android:textAppearance="?android:attr/textAppearanceLarge"  
    android:id="@+id/teks1"/>  
  
<TextView  
    android:paddingTop="20dp"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:text=" DIGUNAKAN UNTUK SUB HEADING "  
    android:typeface="monospace"  
    android:textColor="@color/colorAccent"  
    android:layout_below="@+id/teks1"  
    android:id="@+id/teks2"  
    android:textSize="20sp"  
    android:textAppearance="?android:attr/textAppearanceMedium" />
```

Preview :



6. Text Style

Kadang ada beberapa style yang ingin kita gunakan saat menampilkan teks yaitu bold dan italic.

```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="TANPA MENGGUNAKAN STYLE "  
    android:textSize="14sp"  
    android:id="@+id/teks1"/>  
  
<TextView  
    android:paddingTop="20dp"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="MENGGUNAKAN STYLE BOLD"  
    android:textStyle="bold"  
    android:textColor="@color/colorAccent"  
    android:layout_below="@+id/teks1"  
    android:id="@+id/teks2"  
    android:textSize="20sp"/>  
  
<TextView  
    android:paddingTop="20dp"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="MENGGUNAKAN STYLE ITALIC"  
    android:textStyle="italic"  
    android:textColor="@color/colorPrimary"  
    android:layout_below="@+id/teks2"  
    android:id="@+id/teks3"  
    android:textSize="20sp"/>
```



The screenshot shows a simple Android application interface. It consists of three vertically stacked TextViews. The top TextView has a white background and contains the text "TANPA MENGGUNAKAN STYLE" in a standard black font. The middle TextView has a white background and contains the text "MENGGUNAKAN STYLE BOLD" in a bold black font. The bottom TextView has a white background and contains the text "MENGGUNAKAN STYLE ITALIC" in an italicized black font. All three TextViews have a thin black border and are set against a light gray background.

TANPA MENGGUNAKAN STYLE

MENGGUNAKAN STYLE BOLD

MENGGUNAKAN STYLE ITALIC

7. Type Face

Dengan atribut ini kita bisa memilih tipe font bawaan yang disebutkan tadi diatas yaitu : serif, monospace, dan sans.

```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="TANPA MENGGUNAKAN FONT SERIF"  
    android:typeface="serif"  
    android:textSize="14sp"  
    android:id="@+id/teks1"/>  
  
<TextView  
    android:paddingTop="20dp"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="MENGGUNAKAN FONT MONOSPACE"  
    android:typeface="monospace"  
    android:textColor="@color/colorAccent"  
    android:layout_below="@+id/teks1"  
    android:id="@+id/teks2"  
    android:textSize="20sp"/>  
  
<TextView  
    android:paddingTop="20dp"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="MENGGUNAKAN FONT SANS"  
    android:typeface="sans"  
    android:textColor="@color/colorPrimary"  
    android:layout_below="@+id/teks2"  
    android:id="@+id/teks3"  
    android:textSize="20sp"/>
```

Preview :



8. Text Alignment

Digunakan untuk melakukan pengaturan posisi teks (perataan).

```
<TextView  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:text=" PERATAAN TENGAH (CENTER) "  
    android:typeface="serif"  
    android:textSize="20sp"  
    android:textAlignment="center"  
    android:id="@+id/teks1"/>  
  
<TextView  
    android:paddingTop="20dp"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:text="TEKS POSISI DEFAULT "  
    android:typeface="monospace"  
    android:textColor="@color/colorAccent"  
    android:layout_below="@+id/teks1"  
    android:id="@+id/teks2"  
    android:textSize="20sp"  
    />
```

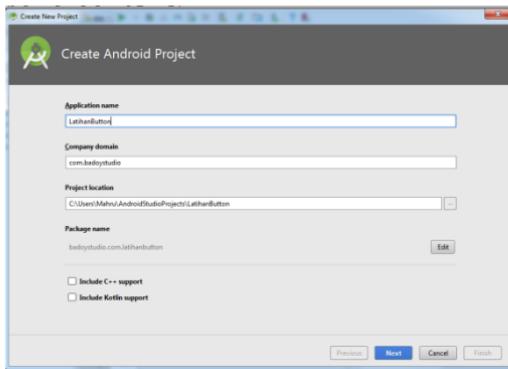
Preview :



Button

Button digunakan membuat dan mengatur tombol pada aplikasi android. Mari kita praktekan cara menggunakan button pada android studio.

- Buka Android Studio kemudian buat project baru dengan nama “**latihanbutton**”.



- Ketika workspace LihatButton sudah muncul seperti gambar diatas, buka file **activity_main.xml** lalu ketikan kode-kode dibawah ini.

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="20dp"
    tools:context=".MainActivity">
```

```
<Button  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:id="@+id/button1"  
    android:layout_marginBottom="20dp"  
    android:text="button menyesuaikan isi teks "  
    android:elevation="0dp" />  
  
<Button  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_marginBottom="20dp"  
    android:layout_below="@+id/button1"  
    android:id="@+id/button2"  
    android:text="button menyesuaikan lebar layout  
" />  
  
<Button  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_below="@+id/button2"  
    android:layout_marginBottom="30dp"  
    android:id="@+id/button3"  
    android:background="@color/colorPrimary"  
    android:text="button dengan warna " />  
  
</RelativeLayout>
```

Preview :



Button bisa juga menggunakan background dari gambar yang kita miliki caranya yaitu :

- **Copykan terlebih dahulu gambar** yang akan dijadikan background pada android studio. Langkah-langkahnya adalah : **Klik kanan pada folder drawable yang ada pada directory res.**
- **Copykan gambar** yang sudah kita siapkan untuk dijadikan background dari button **ke folder drawable**. Ganti nama gambar menjadi button1



Jika sudah silahkan ketikan kode-kode xml berikut dibawah kode botten pada activity_main.xml :

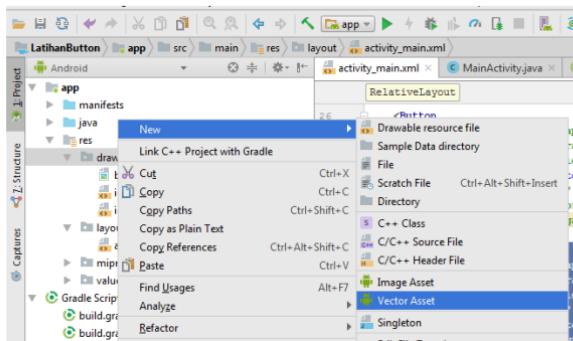
```
<Button  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_below="@+id/button3"  
    android:id="@+id/button4"  
    android:layout_marginBottom="30dp"  
    android:background="@drawable/button1"  
    android:text="button dengan background " />
```

Preview :

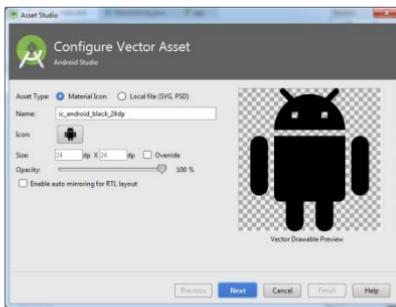


Untuk menambahkan icon di sebelah kanan ataupun kiri pada button adalah:

- Tambahkan asset dari android studio. Caranya : **Klik kanan pada drawable > New > Vector Asset**



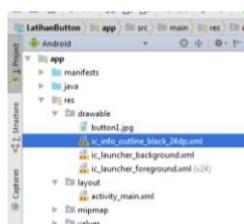
- **Klik icon**



- Pada kolom pencarian ketikan “info”. Lalu pilih icon “info outline “



- **Pilih ok lalu Next sampai dengan Finish.** Pada drawable akan muncul icon tadi.



- Kali ini tambahkan kode-kode xml berikut dibawah kode-kode button yang sudah diberikan image background pada activity_main.xml sebelumnya.

```
<Button  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_below="@+id/button4"  
    android:id="@+id/button5"  
    android:layout_alignParentRight="true"  
  
    android:drawableLeft="@drawable/ic_info_outline_black_24dp"  
    android:text="Tentang Aplikasi " />
```

Preview :

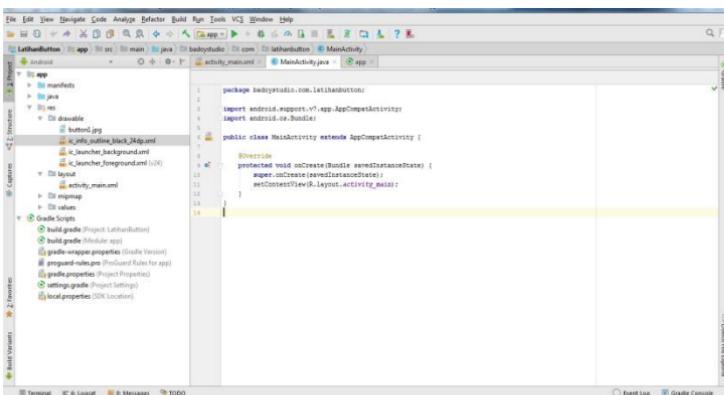


Toast

Supaya button yang diklik menimbulkan aksi (misalnya ketika button di klik akan memunculkan sebuah pesan berupa teks tertentu) maka kali ini kita akan menggunakan toast.

Toast berfungsi memberikan informasi berupa teks yang hanya tampil beberapa detik saja. kita lihat bagaimana toast bekerja yaitu sebagai berikut :

Buka tab **MainActivity.java**.



Kemudian ketik **kode-kode java** dibawah ini :

```
package badoystudio.com.latihanbutton;

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

public class MainActivity extends AppCompatActivity {

    Button btn1, btn2, btn3;

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

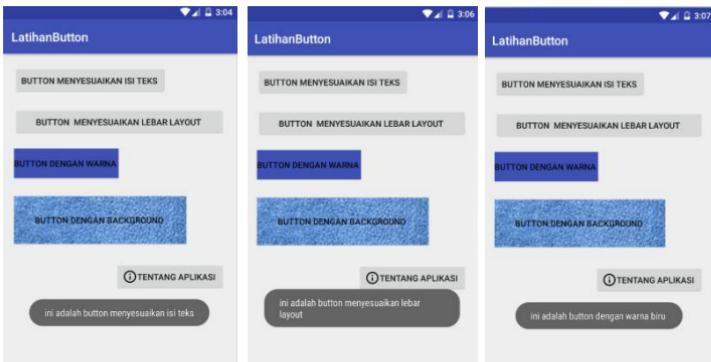
        btn1=(Button)findViewById(R.id.button1);
        btn2=(Button)findViewById(R.id.button2);
        btn3=(Button)findViewById(R.id.button3);

        btn1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Toast.makeText(getApplicationContext(),
                        "ini adalah button menyesuaikan isi teks",
                Toast.LENGTH_LONG).show();
            }
        });

        btn2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Toast.makeText(getApplicationContext(),
                        "ini adalah button menyesuaikan lebar
layout", Toast.LENGTH_LONG).show();
            }
        });

        btn3.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Toast.makeText(getApplicationContext(),
                        "ini adalah button dengan warna biru ",
                Toast.LENGTH_LONG).show();
            }
        });
    }
}
```

ketika aplikasi dijalankan pada emulator maka button1-button3 akan memunculkan informasi toast ketika di klik.



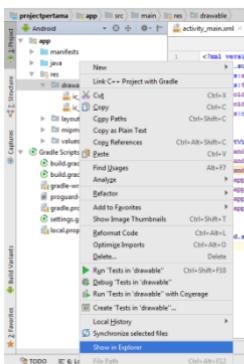
ImageView

ImageView merupakan komponen view yang bisa kita gunakan untuk menambahkan / menampilkan gambar pada aplikasi android yang kita buat.

Penggunaan ImageView

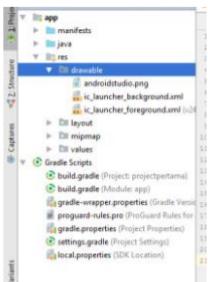
Berikut ini merupakan langkah mudah untuk menambahkan gambar pada android studio.

- Klik kanan pada folder **drawable** yang terdapat pada folder **res**



- Pilih **show in explorer**. Copykan gambar yang akan kita gunakan pada aplikasi android ke folder **drawable**.

- Di latihan ini saya memasukan gambar dengan nama **androidstudio.png** ke folder **drawable**
- Jika berhasil gambar akan tampil pada struktur project android di folder **drawable**



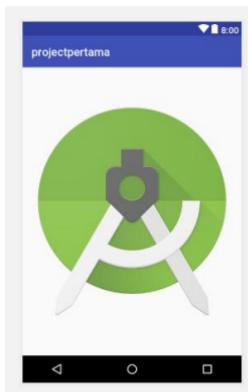
- Selanjutnya buka **activity_main.xml**, lalu tambahkan kode-kode dibawah ini.

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

    <ImageView
        android:id="@+id/imageView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/androidstudio" />

</LinearLayout>
```

- Preview dari kode-kode xml yang sudah dimasukan di atas yaitu :



Atribut ImageView

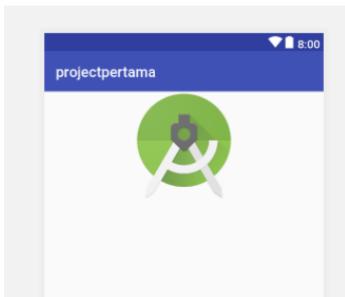
Beberapa atribut pada **ImageView** yang sering digunakan adalah :

No	Atribut	Fungsi
1.	<ImageView	Kode pembuka untuk menampilkan gambar
2.	android:id	Memberikan id pada gambar
3.	android:src	Menampilkan gambar dari source (sumber gambar) contoh diatas diambil dari folder drawable

4.	layout_width dan layout_height	Digunakan untuk mengatur ukuran gambar. Ukuran gambar bisa kita gunakan pixel sebagai satuan. Contoh 300px.
5.	layout_gravity	mengatur posisi gambar. Apakah ingin ditempatkan di tengah atau tidak pada layout.

```
<ImageView  
    android:id="@+id/imageView1"  
    android:layout_width="300px"  
    android:layout_height="300px"  
    android:layout_gravity="center"  
    android:src="@drawable/androidstudio" />
```

Hasil dari kode-kode di atas adalah :



Edit Text

Dengan menggunakan EditText kita bisa membuat sebuah tampilan form untuk menginputkan data.

Jenis teks yang biasanya diinputkan dalam sebuah form yaitu :

- Teks biasa
- Angka/ Nomor
- Nomor Telepon
- Passowrd
- Email
- dan Tautan sebuah web (url web)

Penggunaan Edit Text

Buka **activity_main.xml** kemudian masukan kode-kode xml dibawah ini.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="15dp">

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Nama Peserta"
        android:inputType="text" />

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Nomor Peserta"
        android:inputType="number" />

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Nomor Telepon"
        android:inputType="phone" />

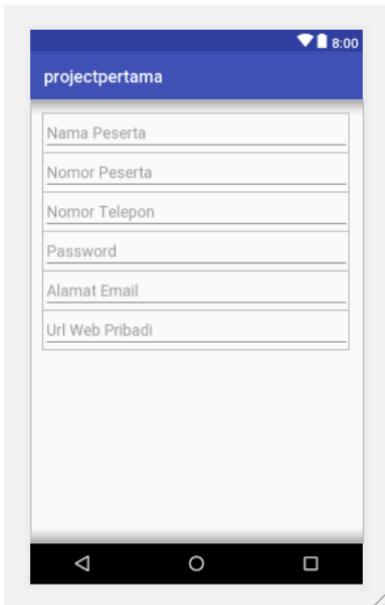
    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Password"
        android:inputType="textPassword" />

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Alamat Email"
        android:inputType="textEmailAddress" />

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Url Web Pribadi"
        android:inputType="textUri" />

</LinearLayout>
```

Maka previewnya adalah :



Atribut Edit Text

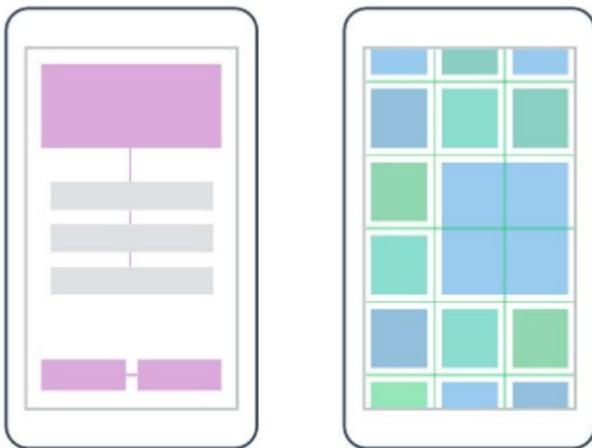
Beberapa atribut **EditText** yang digunakan pada contoh di atas adalah :

No	Atribut	Fungsi
1.	layout_width dan height	digunakan untuk mengatur panjang dan lebar tampilan form input
2.	hint	kalau diweb biasanya kita

		mengenal placeholder. Ini digunakan untuk memberikan informasi form jenis teks apa yang akan kita inputkan
3.	inputType	Digunakan untuk mengisi jenis input teks pada form : <ul style="list-style-type: none">○ texture = Untuk url web○ text = Untuk teks biasa○ number = Inputan berupa nomor/ angka○ phone = Inputan berupa nomor telepon○ textPassword = Inputan berupa password○ textEmailAddress = Inputan berupa alamat email

BAB 11

KOMPONEN VIEW GROUP



Layout Android

Ada beberapa layout yang bisa kita gunakan untuk membangun User interface (UI) yang menarik pengguna, diantaranya yaitu :

- Linear Layout
- Relative Layout
- Absolute Layout
- Frame Layout

LinearLayout

Linear Layout merupakan layout yang berfungsi untuk menampilkan komponen-komponen aplikasi contohnya : teks, tombol, gambar dan lainnya dengan cara **vertikal** ataupun**horizontal**.

Kesederhanaan Linear Layout memudahkan penggunaan namun membatasi fleksibilitasnya. Untuk aplikasi yang memiliki design tidak terlalu kompleks maka linear layout sangat cocok untuk digunakan.

Beberapa tag pada xml yang digunakan pada Linear Layout yaitu :

1. Tag LinearLayout

Pada saat kita membuat project baru menggunakan android studio defaul layout yang digunakan adalah RelativeLayout.

Maka kita harus merubahnya menggunakan Tag LinearLayout yaitu tag pembuka dari Linear Layout, juga harus dilengkap dengan layout_width dan height jika tidak maka akan terjadi error. Untuk orientationnya sendiri defaultnya horizontal.

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

2. Tag Orientation

Kita dapat mengatur orientasi untuk ditampilkan secara horisontal atau vertikal, sehingga item muncul dalam satu kolom atau satu baris. Lengkapi kode diatas dengan menambahkan tag :

```
    android:orientation="horizontal"
```

sehingga menjadi :

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

</LinearLayout>
```

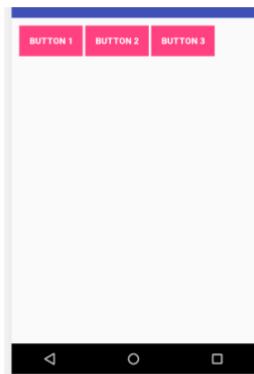
Orientation Horizontal

Contoh pengguna tag orientation horizontal yaitu :

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:padding="10dp"
        android:orientation="horizontal" >
```

```
<Button  
    android:layout_width="100dp"  
    android:layout_height="wrap_content"  
    android:text="Button 1"  
    android:background="@color/colorAccent"  
    android:layout_margin="2dp"  
    android:textColor="#fff"  
    android:textStyle="bold"/>  
<Button  
    android:layout_width="100dp"  
    android:layout_height="wrap_content"  
    android:text="Button 2"  
    android:background="@color/colorAccent"  
    android:layout_margin="2dp"  
    android:textColor="#fff"  
    android:textStyle="bold"/>  
<Button  
    android:layout_width="100dp"  
    android:layout_height="wrap_content"  
    android:text="Button 3"  
    android:background="@color/colorAccent"  
    android:layout_margin="2dp"  
    android:textColor="#fff"  
    android:textStyle="bold" />  
  
</LinearLayout>
```

Maka hasilnya seperti gambar dibawah ini :



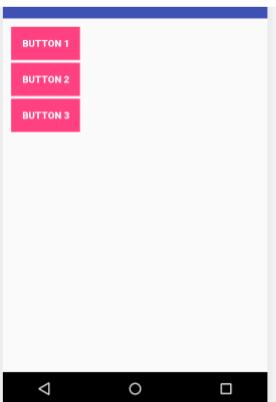
Orientation Vertical

Contoh pengguna tag orientation vertical yaitu :

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/
    android"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:padding="10dp"
        android:orientation="vertical" >

    <Button
        android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:text="Button 1"
        android:background="@color/colorAccent"
        android:layout_margin="2dp"
        android:textColor="#fff"
        android:textStyle="bold"/>
    <Button
        android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:text="Button 2"
        android:background="@color/colorAccent"
        android:layout_margin="2dp"
        android:textColor="#fff"
        android:textStyle="bold"/>
    <Button
        android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:text="Button 3"
        android:background="@color/colorAccent"
        android:layout_margin="2dp"
        android:textColor="#fff"
        android:textStyle="bold"/>
</LinearLayout>
```

Preview :



3. Tag Layout Gravity

Tag gravity digunakan untuk mengatur tata letak komponen yang ada pada linear layout. Contohnya :

```
    android:layout_gravity="center "
```

Tag tersebut digunakan untuk mengatur tata letak komponen agar berada di tengah.

Penerapannya yaitu :

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:orientation="vertical" >

    <Button
        android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:text="Button 1"
        android:background="@color/colorAccent"
        android:layout_margin="2dp"
        android:textColor="#fff"
        android:textStyle="bold"/>
    <Button
        android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:text="Button 2"
        android:background="@color/colorAccent"
        android:layout_margin="2dp"
        android:textColor="#fff"
        android:textStyle="bold"/>
    <Button
        android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:text="Button 3"
        android:background="@color/colorAccent"
        android:layout_margin="2dp"
        android:textColor="#fff"
        android:textStyle="bold"/>
</LinearLayout>
```

Preview :



Yang lainnya bisa kamu coba sendiri **layout_gravity** : **center, right, left, dan bottom.**

4. Tag Weight

Weight digunakan untuk memberikan bobot pada komponen layout. Contohnya sebagai berikut :

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal" >
    <TextView
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:text="teks 1"
        android:background="@color/colorAccent"
        android:layout_weight="5"/>
```

```
<TextView
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:text="teks 2"

    android:background="@color/colorPrimary"
    android:layout_weight="5"/>

<TextView
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:text="teks 3"

    android:background="@color/colorPrimaryDark"
    android:layout_weight="5"/>

</LinearLayout>
```

Maka hasilnya akan seperti gambar dibawah ini :

Weight pada orientation horizontal.



Relative Layout

Relative Layout merupakan layout yang berfungsi untuk mengatur tata letak komponen atau widget aplikasi android dengan cara relative (secara bebas) tidak hanya vertikal atau horisontal saja seperti pada Linear Layout.

Karena alasan tersebut Relative layout menjadi layout yang paling banyak digunakan setelah Linear Layout di Android.

Tag- tag Relative Layout

Adapun beberapa **Tag xml** penting yang sering digunakan pada **Relative Layout** yaitu :

Tag	Keterangan
android:layout_alignParentBottom	Digunakan untuk menata letak komponen (widget) di tepi paling bawah layar.
android:layout_alignParentTop	Digunakan untuk menata letak komponen (widget) di tepi paling atas layar.

android:layout_alignParentLeft	Digunakan untuk menata letak komponen (widget) di tepi paling kiri layar.
android:layout_alignParentRight	Digunakan untuk menata letak komponen (widget) di tepi paling kanan layar.
android:layout_below	Digunakan untuk memposisikan komponen (widget) di bawah komponen (widget) yang lainnya.
android:layout_above	Digunakan untuk memposisikan komponen (widget) di atas komponen (widget) yang lainnya.

Contoh Penggunaan

Buat project baru lalu ketikan kode-kode berikut pada Activity_main.xml.

```
<?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:padding="10dp"
    android:background="#ef0000" >
```

```
<EditText  
    android:id="@+id/name"  
    android:layout_width="200dp"  
    android:layout_height="wrap_content"  
    android:background="#f8eff8f3"  
    android:hint="Nama "  
    android:padding="16dp" />  
  
<EditText  
    android:id="@+id/email"  
    android:layout_width="180dp"  
    android:layout_height="wrap_content"  
    android:layout_marginLeft="5dp"  
    android:layout_toRightOf="@+id/name"  
    android:background="#f8eff8f3"  
    android:hint="Alamat "  
    android:padding="16dp" />  
  
<EditText  
    android:id="@+id/subject"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_below="@+id/name"  
    android:layout_marginTop="16dp"  
    android:background="#f8eff8f3"  
    android:hint="Subjek"  
    android:padding="16dp" />  
  
<EditText  
    android:id="@+id/message"  
    android:layout_width="match_parent"  
    android:layout_height="200dp"  
    android:layout_below="@+id/subject"  
    android:layout_marginTop="16dp"  
    android:background="#f8eff8f3"  
    android:gravity="top"  
    android:hint="Pesan"  
    android:padding="16dp" />  
  
<Button  
    android:id="@+id/submit"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_below="@+id/message"  
    android:layout_centerHorizontal="true"  
    android:layout_marginTop="16dp"  
    android:padding="16dp"  
    android:text="KIRIM" />  
/>
```

Preview :



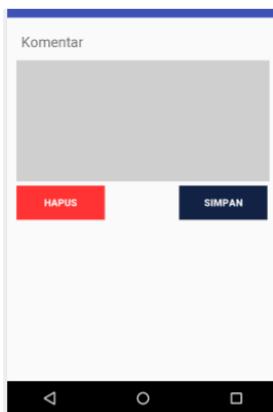
Contoh lainnya :

```
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:padding="10dp"
    android:layout_height="match_parent" >

    <TextView
        android:id="@+id/lblComments"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Komentar"
        android:layout_alignParentTop="true"
        android:layout_alignParentLeft="true"
        android:textSize="20dp"
        android:padding="10dp" />
    <EditText
        android:id="@+id/txtComments"
        android:layout_width="match_parent"
        android:layout_height="170dp"
        android:textSize="18sp"
        android:layout_alignLeft="@+id/lblComments"
        android:layout_below="@+id/lblComments"
        android:layout_centerHorizontal="true"
        android:background="#32222"
        android:layout_margin="3dp" />
```

```
<Button  
    android:id="@+id/btnSave"  
    android:layout_width="125dp"  
    android:layout_height="wrap_content"  
    android:text="Simpan"  
    android:layout_below="@+id/txtComments"  
    android:layout_alignRight="@+id/txtComments"  
    android:background="#f124"  
    android:textColor="#fff"  
    android:layout_margin="3dp"  
    android:textStyle="bold"/>  
  
<Button  
    android:id="@+id/button1"  
    android:layout_width="125dp"  
    android:layout_height="wrap_content"  
    android:layout_alignBaseline="@+id/btnSave"  
    android:layout_alignBottom="@+id/btnSave"  
    android:layout_alignParentLeft="true"  
    android:text="Hapus"  
    android:background="#ff33"  
    android:textColor="#fff"  
    android:layout_margin="3dp"  
    android:textStyle="bold"/>  
  
</RelativeLayout>
```

Hasil dari tag diatas yaitu seperti gambar dibawah ini :



Untuk memahami lebih lanjut tentang penggunaan **Relative Layout** kita harus sering-sering latihan agar terbiasa dan menjadi mudah ketika membuat UI menggunakan **Relative Layout**.

ListView

ListView merupakan tampilan yang mengelompokkan beberapa item dan menampilkannya dalam daftar vertikal. Item yang ditampilkan secara otomatis dimasukkan ke daftar menggunakan Adapter yang menarik konten dari seperti basis data atau array. Contohnya adalah : daftar kontak nomor handphone dibuat menggunakan listview.

Adapter digunakan untuk menjembatani antara komponen UI dan sumber data yang mengisi data ke dalam Komponen UI.

Penggunaan ListView

Pada contoh penggunaan **ListView** kali ini kita akan menampilkan data array dari kodingan java. Artinya data

yang kita tampilkan berupa data statis bukan dinamis (dari database).

- Buat project baru dengan nama “ **contohlistview** ”.
- Jika workspace android studio sudah siap digunakan, buka **activity_main.xml** lalu tambahkan kode-kode dibawah ini.

```
<?xml version="1.0" encoding="utf-8"?>
<ListView
    xmlns:android="http://schemas.android.com/apk/res/android"

    xmlns:tools="http://schemas.android.com/tools"
        android:id="@+id/listView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:padding="30dp"
        tools:context=".MainActivity">

</ListView>
```

- Selanjutnya tambahkan **array** pada **strings.xml**. Sehingga kode-kode xml yang ada pada **strings.xml** menjadi seperti dibawah ini.

```
<resources>
    <string
        name="app_name">contohlistview</string>
    <string-array name="smartphone_array">
        <item>Nokia</item>
        <item>Nexus</item>
        <item>Iphone</item>
        <item>Samsung</item>
        <item>Asus</item>
        <item>Xiaomi</item>
        <item>Oppo</item>
        <item>Vivo</item>

    </string-array>
</resources>
```

- Selanjutnya buka **MainActivity.java** dan tambahkan kodingan berikut.

```
package badoystudio.com.contohlistview;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.ListView;

import android.widget.Toast;

public class MainActivity extends AppCompatActivity
implements AdapterView.OnItemClickListener{
    ListView listView;
    ArrayAdapter<CharSequence> adapter;
```

```

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

    listView =
    (ListView) findViewById(R.id.listView);
    adapter =
    ArrayAdapter.createFromResource(this,R.array.smartphone_
array, android.R.layout.simple_list_item_1);
    //adapter= new ArrayAdapter<String>(this,
    android.R.layout.simple_list_item_1);
    listView.setAdapter(adapter);
    listView.setOnItemClickListener(this);
}

@Override
public void onItemClick(AdapterView<?> parent, View
view, int position, long id) {
    Toast.makeText(this, adapter.getItem(position),
    Toast.LENGTH_SHORT).show();
}
}
}

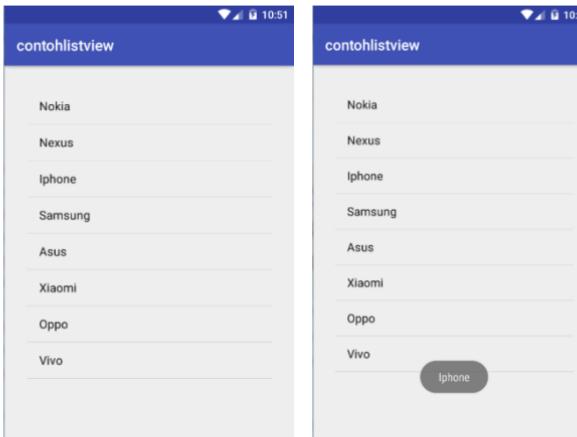
```

- Penjelasan singkat kode-kode java di atas :

No	Kode	Fungsi
	<pre> ArrayAdapter.crea teFromResource(th is,R.array.smartp hone_array,androi d.R.layout.simple _list_item_1); </pre>	<p>digunakan untuk mengambil nilai array pada strings.xml yang bernama smartphone_array. Kemudian ditampilkan dengan tampilan simple_list_item1</p>

	onItemClick	Ketika data pada listview dipilih akan ditampilkan toast.
--	-------------	---

- Silahkan running project, jika tidak ada error akan tampil seperti gambar dibawah ini.



GridView

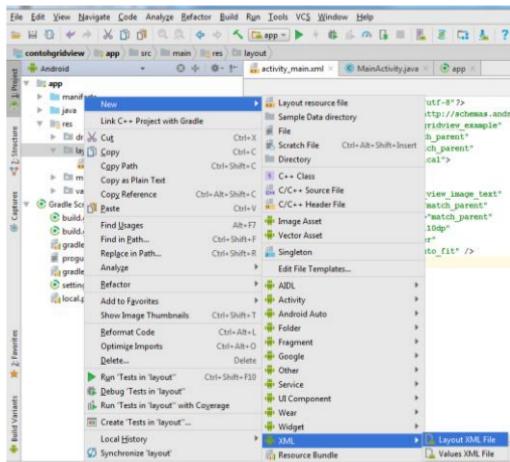
GridView adalah salah satu layout yang paling sering digunakan. Layout GridView digunakan untuk menampilkan data dalam grid bergulir dua dimensi.

Layout ini biasanya digunakan pada untuk membuat aplikasi seperti pembaca berita, penampil gambar, aplikasi musik, dan lain sebagainya. GridView adalah subkelas dari AdapterView, yang mengambil data dari sumber eksternal.

Baik langsung saja kita latihan membuat aplikasi menggunakan layout gridview.

- Buka android studio lalu buat project baru dengan nama “**contohgridview**”
- Setelah workspace android studio terbuka dan siap digunakan.
- Dalam latihan ini kita tidak akan menggunakan **activity_main.xml** dan **MainActivity.java**

- Buat layout baru dengan cara klik kanan pada res > Layout. Pilih New > XML > Layout XML File. Isikan “gridview_image_text” sebagai nama layout.



- Masukan kode-kode dibawah ini pada gridview_image_text.xml.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/android_gridview_example"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <GridView
        android:id="@+id/grid_view_image_text"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:columnWidth="110dp"
        android:gravity="center"
        android:numColumns="auto_fit" />

</LinearLayout>
```

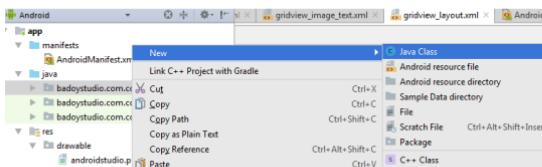
- Tambahkan kembali layout baru dengan nama **gridview_layout.xml**. Caranya sama dengan membuat layout sebelumnya
- Masukan kode-kode xml dibawah ini.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/android_custom_gridview_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center_horizontal"
    android:orientation="vertical">

    <ImageView
        android:id="@+id/android_gridview_image"
        android:layout_width="50dp"
        android:layout_height="50dp"
        android:layout_marginTop="15dp" />

    <TextView
        android:id="@+id/android_gridview_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="5dp"
        android:textSize="12sp" />
</LinearLayout>
```

- Buat class java baru dengan cara klik kanan pada nama folder nama project android yang kita buat pada folder java. Pilih New > Java Class



- Beri nama file “ **CustomGridViewActivity** ”

```
package badoystudio.com.contohgridview;

import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.BaseAdapter;
import android.widget.ImageView;
import android.widget.TextView;

public class CustomGridViewActivity extends BaseAdapter {

    private Context mContext;
    private final String[] gridViewString;
    private final int[] gridViewImageId;

    public CustomGridViewActivity(Context context,
        String[] gridViewString, int[] gridViewImageId) {
        mContext = context;
        this.gridViewImageId = gridViewImageId;
        this.gridViewString = gridViewString;
    }

    @Override
    public int getCount() {
        return gridViewString.length;
    }

    @Override
    public Object getItem(int i) {
        return null;
    }

    @Override
    public long getItemId(int i) {
        return 0;
    }
}
```

```
@Override
public View getView(int i, View convertView, ViewGroup
parent) {
    View gridViewAndroid;
    LayoutInflater inflater = (LayoutInflater) mContext
    .getSystemService(Context.LAYOUT_INFLATER_SERVICE);

    if (convertView == null) {

        gridViewAndroid = new View(mContext);
        gridViewAndroid =
        inflater.inflate(R.layout.gridview_layout, null);
        TextView textViewAndroid = (TextView)
        gridViewAndroid.findViewById(R.id.android_gridview_text);
        ;
        ImageView imageViewAndroid = (ImageView)
        gridViewAndroid.findViewById(R.id.android_gridview_image);
        ;
        textViewAndroid.setText(gridViewString[i]);

        imageViewAndroid.setImageResource(gridViewImageId[i]);
    } else {
        gridViewAndroid = (View) convertView;
    }

    return gridViewAndroid;
}
```

- Satu lagi buat class java dengan nama file “**GridViewImageTextActivity**“

- Masukan kode-kode berikut.

```
package badoystudio.com.contohgridview;

import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.AdapterView;
import android.widget.GridView;
import android.widget.Toast;

public class GridViewImageTextActivity extends
AppCompatActivity {

    GridView androidGridView;

    String[] gridViewString = {
        "Android Studio", "Kotlin", "React Native",
        "Android Studio", "Kotlin", "React Native",
        "Android Studio", "Kotlin", "React Native",

    } ;
    int[] gridViewImageId = {
        R.drawable.androidstudio,
        R.drawable.kotlin, R.drawable.react,
        R.drawable.androidstudio,
        R.drawable.kotlin, R.drawable.react,
        R.drawable.androidstudio,
        R.drawable.kotlin, R.drawable.react,
    };

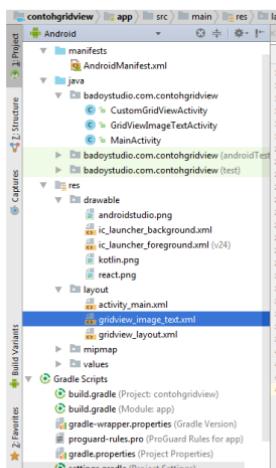
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.gridview_image_text);

        CustomGridViewActivity adapterViewAndroid = new
        CustomGridViewActivity(GridViewImageTextActivity.this,
        gridViewString, gridViewImageId);

        androidGridView=(GridView)findViewById(R.id.grid_view_im
        age_text);
        androidGridView.setAdapter(adapterViewAndroid);
        androidGridView.setOnItemClickListener(new
        AdapterView.OnItemClickListener() {
```

```
        @Override
        public void onItemClick(AdapterView<?> parent,
View view,
                           int i, long id) {
    Toast.makeText(GridViewImageTextActivity.this, "GridView
Item: " + gridViewString[i], Toast.LENGTH_LONG).show();
}
});
```

- Jika ada error biarkan saja. Hal itu terjadi karena kita belum memasukan gambar pendukung
 - Sekarang masukan gambar pendukung yang ada di folder “gambar gridview” ke folder **drawable**.
 - Kurang lebih susunan folder dan file yang kita buat seperti dibawah ini.



- Sebelum di running buka folder **Manifest** lalu edit file **AndroidManifests.xml** menjadi seperti dibawah ini.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest
    xmlns:android="http://schemas.android.com/apk/res/android"
        package="badoystudio.com.contohgridview">

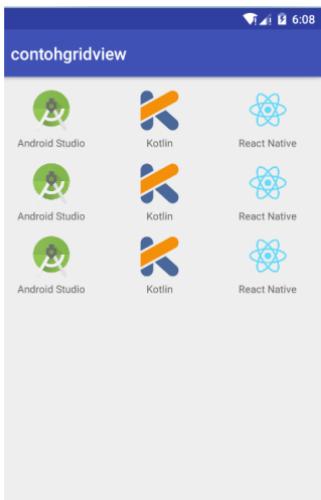
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity
            android:name=".GridViewImageTextActivity">
            <intent-filter>
                <action
                    android:name="android.intent.action.MAIN" />

                <category
                    android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

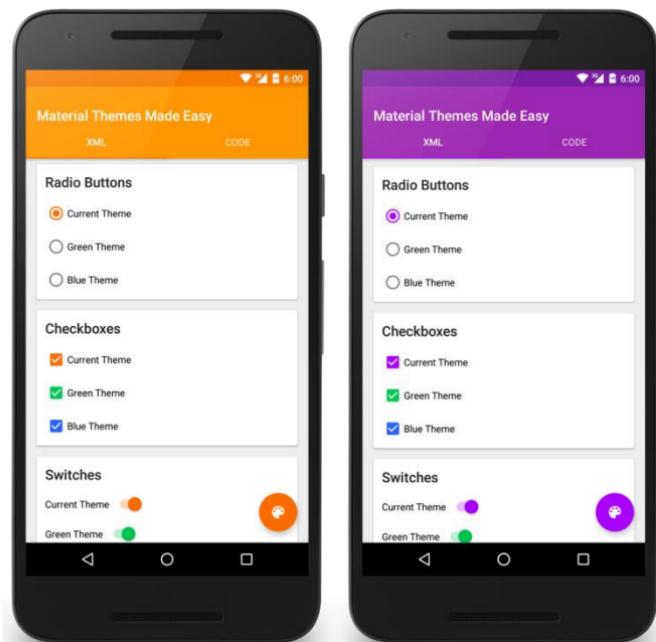
Di **AndroidManifests.xml** ini kita mengatur yang tadinya class **MainActivity.java** adalah file yang pertama kali dijalankan sekarang diubah menjadi **GridViewImageTextActivity**.

- Terakhir silahkan running aplikasi android “contohgridview”. Maka hasilnya :



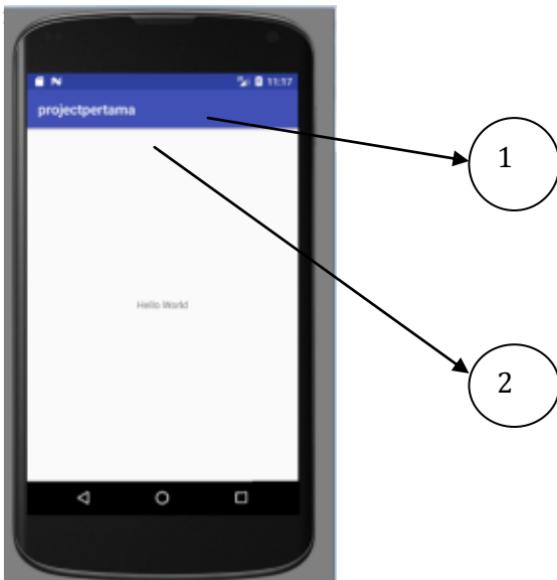
BAB 13

STYLE DAN TEMA APLIKASI ANDROID



Colors.xml

Warna default yang digunakan pada aplikasi android yaitu

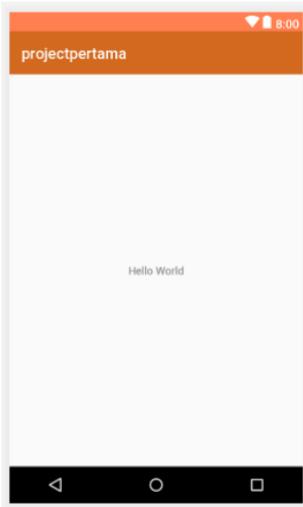


1. ColorPrimaryDark dengan kode hexadecimal : **#303F9F**
2. ColorPrimary dengan kode hexadecimal : **#3F51B5**

Bisakah kita mengubah kedua warna tersebut dengan keinginan kita ? tentu saja bisa. Kita bisa mengubahnya pada file **colors.xml** yang terdapat pada folder **values**.

Coba kita praktekan mengubah kedua warna tersebut lalu lihat hasilnya.

- Buka file **colors.xml**
- Edit kode hexadecimal **ColorPrimaryDark** menggunakan #FF7F50 (warna coral)
- Edit juga kode hexadecimal **ColorPrimary** menggunakan #D2691E (warna chocolate)
- Sekarang buka preview aplikasi dan lihat hasilnya pada **activity_main.xml**



Action Bar



Action Bar nama lainnya adalah App Bar. Pada action bar biasanya terdapat beberapa komponen :

- Nama Aplikasi
- Icon Aplikasi
- Icon Kembali
- Navigasi

Dalam aplikasi android kita bisa memilih apakah mau menggunakan action bar atau tidak. Defaultnya ketika kita membuat aplikasi android menggunakan android studio action bar sudah terpasang.

Jika kita tidak mau menggunakan action bar pada aplikasi android yang kita buat caranya cukup mudah, silahkan ikuti langkah-langkah berikut ini :

- Buka file **styles.xml** pada folder **values**.

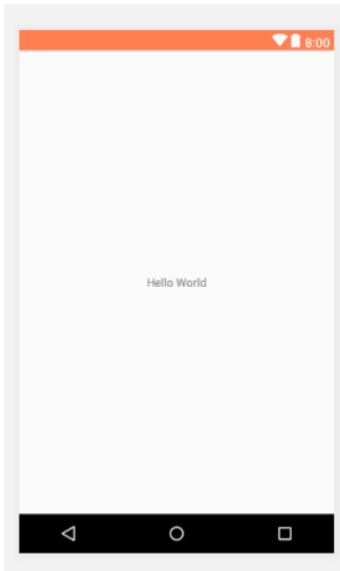
- Ganti baris kode :

```
<style name="AppTheme"
    parent="Theme.AppCompat.Light.DarkActionBar">
```

menjadi :

```
<style name="AppTheme"
    parent="Theme.AppCompat.Light.NoActionBar">
```

- Sekarang coba lihat preview aplikasinya.



Nama Aplikasi Android

Nama aplikasi android yang kita buat akan sesuai dengan nama yang kita tuliskan saat membuat project baru. Nah jika namanya ingin kita ubah tidak maka kita tidak harus membuat ulang project android tersebut.

Untuk mengubah nama aplikasi android caranya sangatlah mudah.

- Pertama buka file **strings.xml** pada folder **values**.
- Ubah nama aplikasi pada kode berikut :

```
<resources>
    <string name="app_name">Ganti
Nama</string>

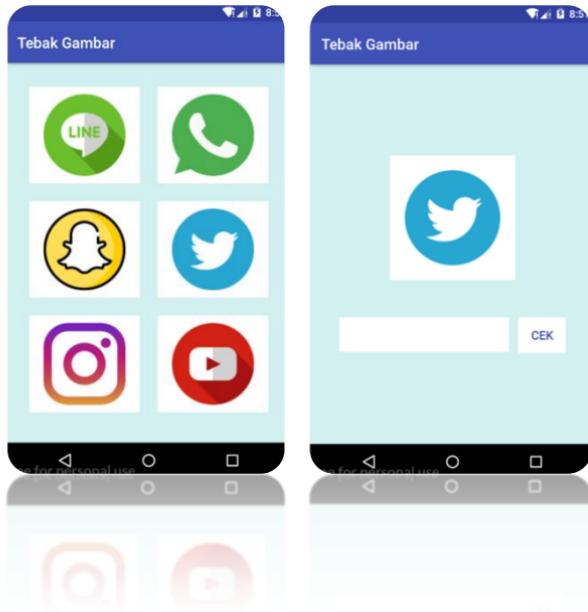
</resources>
```

- Lihat previewnya maka nama aplikasi sudah berubah



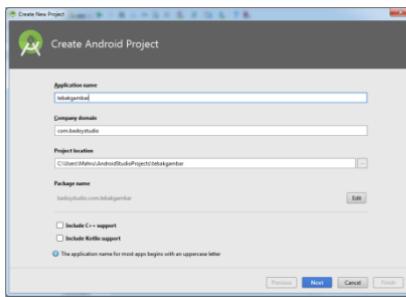
BAB 14

MEMBUAT APLIKASI GAMTE TEBAK GAMBAR



Di bab ini kita akan membuat aplikasi android game tebak gambar sederhana. Adapun langkah-langkahnya sebagai berikut :

1. Buka android studio kemudian buat project baru dengan nama “ **tebakgambar**”⁸



2. Tentukan target device, activity, sampai dengan finish dan workspace android studio terbuka.
3. Sebelum membuat layout masukan gambar-gambar pendukung aplikasi (asset) **ke folder drawable**.⁹

⁸ Silahkan pelajari cara membuat project baru pada bab 3

⁹ Foldernya : pendukungtebakgambar

4. Jika sudah sekarang saatnya kita membuat layout dari aplikasi game tebak gambar. Buka **activity_main.xml** lalu tambahkan kode-kode dibawah ini.

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_gravity="center"
    android:background="#22FF"
    android:padding="20dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:orientation="horizontal">

            <ImageView
                android:padding="10dp"
                android:id="@+id/imageView_line"
                android:layout_width="match_parent"
                android:layout_height="130dp"
                android:layout_weight="1"
                android:layout_margin="12dp"
                android:src="@drawable/line"
                android:background="#fff"/>

            <ImageView
                android:id="@+id/imageView_whatsapp"
                android:layout_width="match_parent"
                android:layout_height="130dp"
                android:layout_weight="1"
                android:layout_margin="12dp"
                android:src="@drawable/whatsapp"
                android:padding="10dp"
                android:background="#fff"/>

        </LinearLayout>
    </LinearLayout>
</ScrollView>
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal">

    <ImageView
        android:id="@+id/imageView_snapchat"
        android:layout_width="match_parent"
        android:layout_height="130dp"
        android:src="@drawable/snapchat"
        android:layout_weight="1"
        android:layout_margin="12dp"
        android:padding="10dp"
        android:background="#fff"/>

    <ImageView
        android:id="@+id/imageView_twitter"
        android:layout_width="match_parent"
        android:layout_height="130dp"
        android:src="@drawable/twitter"
        android:layout_weight="1"
        android:layout_margin="12dp"
        android:padding="10dp"
        android:background="#fff"/>
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal">

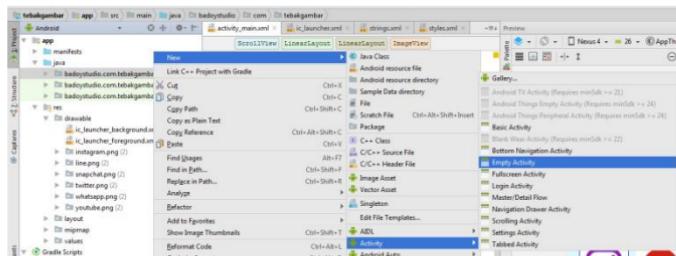
    <ImageView
        android:id="@+id/imageView_instagram"
        android:layout_width="match_parent"
        android:layout_height="130dp"
        android:src="@drawable/instagram"
        android:layout_weight="1"
        android:layout_margin="12dp"
        android:padding="10dp"
        android:background="#fff"/>

    <ImageView
        android:id="@+id/imageView_youtube"
        android:layout_width="match_parent"
        android:layout_height="130dp"
        android:src="@drawable/youtube"
        android:layout_weight="1"
        android:layout_margin="12dp"
        android:padding="10dp"
        android:background="#fff"/>
</LinearLayout>
</ScrollView>
```

5. Preview dari kode-kode diatas yaitu :



6. Tambahkan activity baru dengan nama “ tebak ”. Cara menambahkannya yaitu : **klik kanan pada folder project android yang terdapat pada direktori java > pilih New > Activity > Empty Activity**



7. Beri nama “ TebakActivity ”. Klik finish.

8. Klik folder layout lalu buka file **activity_tebak.xml** lalu masukan kode-kode dibawah ini.

```
<?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:gravity="center"
    android:background="#22FF"
    android:padding="40dp"
    tools:context=".TebakActivity"
    android:orientation="vertical">

    <ImageView
        android:id="@+id/imageView_tebak"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/line"
        android:padding="20dp"
        android:background="@android:color/white"/>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="50dp"
        android:orientation="horizontal"
        android:gravity="bottom">

        <EditText
            android:id="@+id/editText_jawab"
            android:layout_width="match_parent"
            android:layout_height="47dp"
            android:background="@android:color/white"
            android:layout_weight="1"/>

        <Button
            android:id="@+id/buttonCek"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="CEK"
            android:textAllCaps="true"
            android:layout_weight="3"
            android:textColor="@color/colorPrimary"
            android:background="@android:color/white"
            android:layout_marginLeft="12dp"
            android:textSize="16sp"/>

    </LinearLayout>
```

9. Buka file **MainActivity.java** dan tambahkan kode-kode berikut ini.

```
package badoystudio.com.tebakgambar;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.ImageView;

public class MainActivity extends AppCompatActivity {

    ImageView imageView_line, imageView_whatsapp,
    imageView_youtube, imageView_instagram,
    imageView_snapchat, imageView_twitter;

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

        setInisialisasi();
        onClickMantab();
    }

    private void onClickMantab() {

        imageView_line.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent d = new
Intent(MainActivity.this, TebakActivity.class);
                d.putExtra("nama_icon", "Line");
                startActivity(d);
            }
        });
    }

    imageView_snapchat.setOnClickListener(new
View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent c = new
Intent(MainActivity.this, TebakActivity.class);
            c.putExtra("nama_icon", "snapchat");
            startActivity(c);
        }
    });
}
```

```
imageView_twitter.setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent a = new Intent(MainActivity.this,
TebakActivity.class);
        a.putExtra("nama_icon", "twitter");
        startActivity(a);
    }
});

imageView_youtube.setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent al = new
Intent(MainActivity.this, TebakActivity.class);
        al.putExtra("nama_icon", "youtube");
        startActivity(al);
    }
});

imageView_instagram.setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent al = new
Intent(MainActivity.this, TebakActivity.class);
        al.putExtra("nama_icon", "Instagram");
        startActivity(al);
    }
});

imageView_whatsapp.setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent al = new
Intent(MainActivity.this, TebakActivity.class);
        al.putExtra("nama_icon", "whatsapp");
        startActivity(al);
    }
});

private void setInisialisasi() {
    imageView_line=
    (ImageView) findViewById(R.id.imageView_line);
    imageView_snapchat =
    (ImageView) findViewById(R.id.imageView_snapchat);
    imageView_instagram =
    (ImageView) findViewById(R.id.imageView_instagram);
    imageView_twitter =
    (ImageView) findViewById(R.id.imageView_twitter);
    imageView_whatsapp =
    (ImageView) findViewById(R.id.imageView_whatsapp);
    imageView_youtube=
    (ImageView) findViewById(R.id.imageView_youtube);
}
```

10. Selanjutnya tambahkan kode-kode java berikut pada file **TebakActivity.java**.

```
package badoystudio.com.tebakgambar;

import android.content.Intent;
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.ImageView;
import android.widget.Toast;

public class TebakActivity extends AppCompatActivity {

    ImageView imageView_tebak;
    EditText editText_jawabb;
    Button button_cek;

    String jawaban;

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

        setInitialisasi();
        cekIntent();
        onClickJos();
    }

    private void onClickJos() {
        button_cek.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View view) {
                if
(editText_jawabb.getText().toString().equals(jawaban)) {
                    Toast.makeText(TebakActivity.this,"Jawaban anda benar !
",Toast.LENGTH_SHORT).show();
                } else {
                    Toast.makeText(TebakActivity.this,"Jawaban anda salah "
,Toast.LENGTH_SHORT).show();
                }
            }
        });
    }
}
```

```
private void cekIntent() {
    Intent cek = getIntent();
    String nama_icon =
cek.getStringExtra("nama_icon");

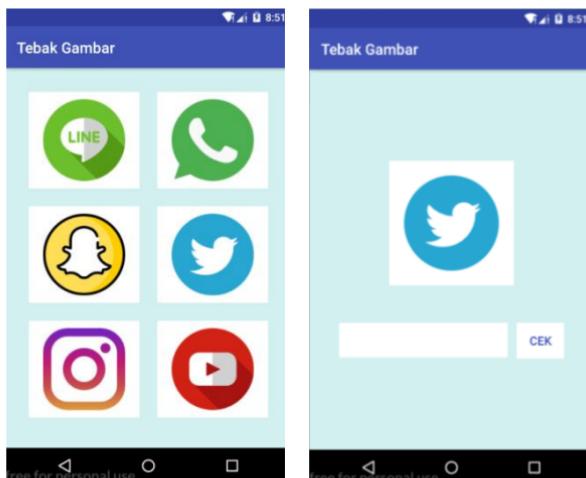
    if (nama_icon.equals("line")){
        imageView_tebak.setImageResource(R.drawable.line);
        jawaban = "line";

    } else if (nama_icon.equals("instagram")){
        imageView_tebak.setImageResource(R.drawable.instagram);
        jawaban = "instagram";
    } else if (nama_icon.equals("youtube")){
        imageView_tebak.setImageResource(R.drawable.youtube);
        jawaban = "youtube";
    } else if (nama_icon.equals("whatsapp")){
        imageView_tebak.setImageResource(R.drawable.whatsapp);
        jawaban = "whatsapp";
    } else if (nama_icon.equals("snapchat")){
        imageView_tebak.setImageResource(R.drawable.snapchat);
        jawaban = "snapchat";
    } else {

        imageView_tebak.setImageResource(R.drawable.twitter);
        jawaban = "twitter";
    }
}

private void setInitalisasi() {
    imageView_tebak =
(ImageView) findViewById(R.id.imageView_tebak);
    editText_jawab =
(EditText) findViewById(R.id.editText_jawab);
    button_cek =
(Button) findViewById(R.id.buttonCek);
}
```

11. Jika sudah tidak ada error silahkan running project android tebak gambar yang sudah kita buat. Maka hasilnya adalah



BAB 15

BUILD APK APLIKASI GAME TEBAK GAMBAR



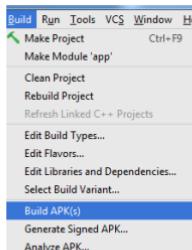
Aplikasi game tebak gambar sederhana sudah dibuat. Kini saatnya kita membuildnya agar bisa kita upload ke Google Playstore dan diinstallkan di smartphone android.

Ada dua cara untuk membuild APK yaitu :

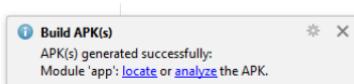
Cara pertama :

Cara pertama ini bisa kita gunakan jika kita hanya ingin mentester aplikasi android di smartphone yang kita miliki (tidak untuk di publish di playstore).

- Pilih menu **Build** lalu pilih **Build APK**



- Tunggu beberapa saat akan muncul pilihan **Apk(s) generated successfully.**



- Pilih locate secara otomatis windows explorer akan terbuka dan file apk dari aplikasi tebak gambar sudah kita dapatkan dengan nama file “**apk-debug.apk**”.

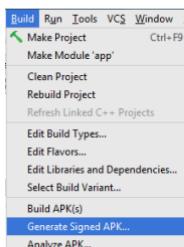


- File tersebut sudah bisa kita installkan di smartphone android yang kita miliki.

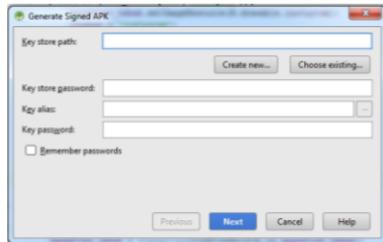
Cara kedua :

Untuk mempublish apk kita ke playstore kita harus menggunakan cara yang kedua. Yaitu membuat apk dengan generate signed apk.

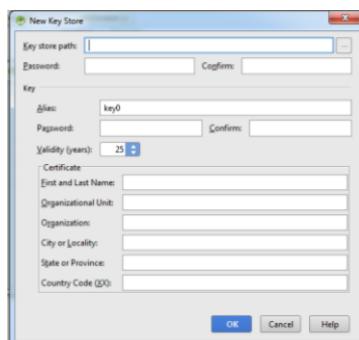
- Pilih menu **Build** lalu pilih **Generated sign APK**



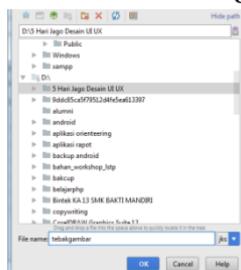
- Maka akan muncul dialog seperti berikut :



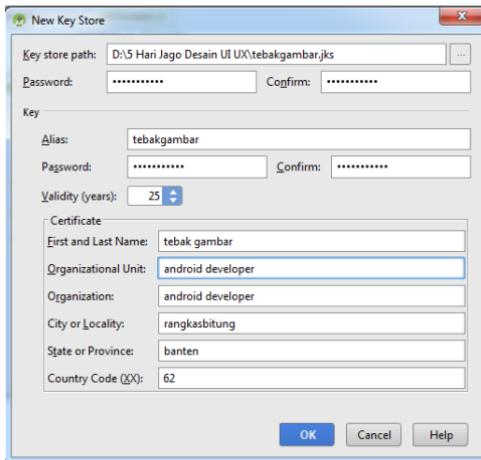
- Klik **Create new ...** maka akan muncul jendela key store.



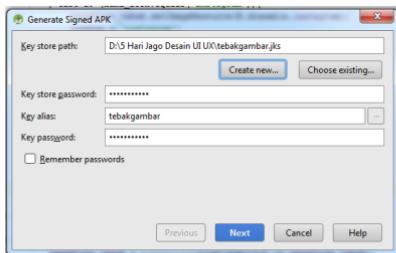
- Klik tombol ... lalu pilih folder yang akan kita gunakan untuk menyimpan file keystore > isikan nama file “tebakgambar” dan klik ok.



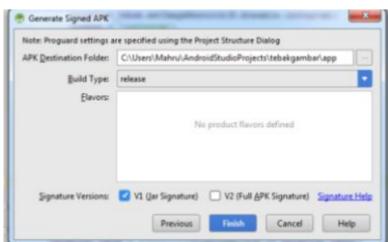
- **Isi password** (inputkan minimal 4 karakter lalu confirm). **Beri nama aplikasi pada alias.** **Validitynya biarkan terisi 25.**
- Kemudian isi **first name, last name, organisasi unit, kota atau lokasi.** Kurang lebih seperti gambar dibawah ini. Jika sudah **klik ok.**



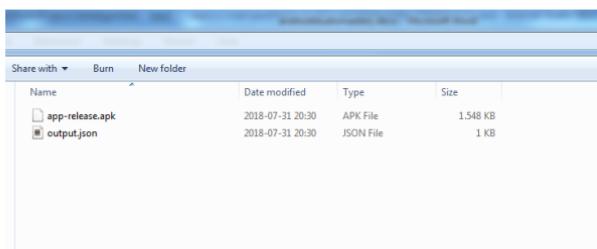
- **Klik next** pada jendela berikut



- Tentukan lokasi untuk file apk yang akan di rilis. **Pilih v1 (jar signature)** pada **signature version**. Lalu **klik finish**.



- Tunggu beberapa saat sampai file apk kita di rilis oleh android studio. **Klik locate** jika sudah sukses lalu **pilih folder rilis**. Akan ada file dengan nama **apk-release.apk**.



BAB 16

DAFTAR AKUN GOOGLE DEVELOPER (GOOGLE CONSOLE)

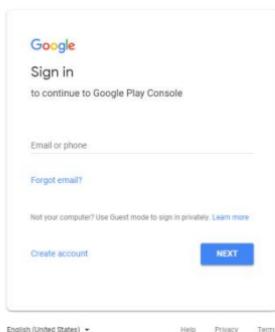


Daftar Akun Google Developer

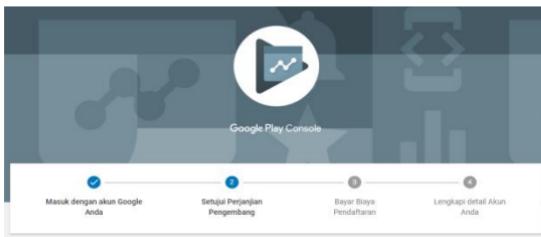
Setelah membuild aplikasi android yang kita buat menggunakan android studio menjadi *file APK*. Tentu saja kita ingin aplikasi buatan kita dapat di download dan digunakan oleh pengguna yang sudah kita tentukan.

Sebelum mengupload uplikasi ke google playstore kita harus memiliki akun google developer (google console terlebih dahulu). Berikut ini langkah-langkah daftar akun google developer :

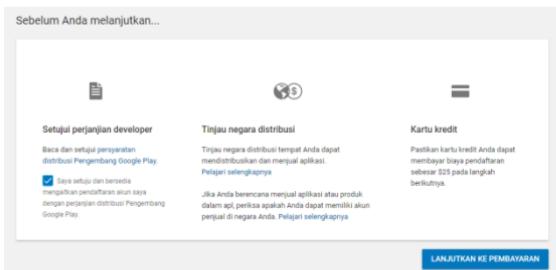
- Pertama-tama silahkan kunjungi link dibawah ini :
<https://play.google.com/apps/publish/signup/>
- Pastikan kamu sudah memiliki akun gmail. Jika belum silahkan buat terlebih dahulu
- Masuk menggunakan akun gmail yang sudah kamu



- Tampilan halaman untuk mendaftar yaitu seperti gambar dibawah ini.



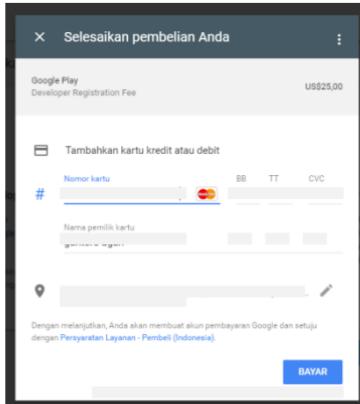
- Scroll ke bawah lalu centang setuju dan bersedia, lalu tekan tombol lanjutkan ke pembayaran



- Biaya yang dibutuhkan untuk daftar akun google developer ini adalah \$25 jika dirupiahkan dengan kurs Rp. 14.000 yaitu sekitar Rp. 350.000 dan harus menggunakan kartu kredit. Namun jangan khawatir jika belum punya kartu kredit kita bisa menggunakan

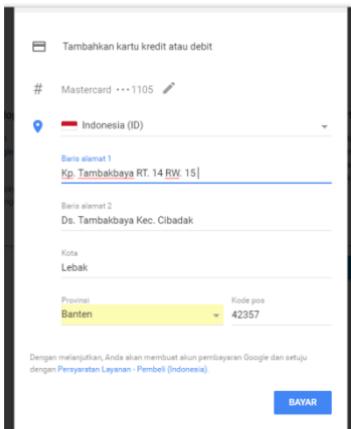
VCC¹⁰. Silahkan *googling* dengan kata kunci “ jasa vcc google developer ”.

- Langkah selanjutnya isikan nama dan kartu kredit atau vcc yang sudah kamu miliki.



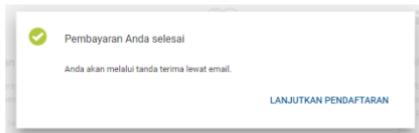
- Tambahkan juga nomor telepon aktif, dan alamat tempat tinggal saat ini. Lalu klik bayar.

¹⁰ VCC (Virtual Credit Card) atau Kartu Kredit Virtual. Isinya yaitu nomor kartu, nama, tanggal kadaluarsa dan lainnya.



- Gambar dibawah ini akan tampil jika pembayaran telah sukses. Selain itu akan ada notifikasi sukses pembayaran dari google payment ke emailmu¹¹. Biaya yang dibayar untuk pendaftaran akun ini berlaku untuk selamanya (asalkan akun tidak di banned oleh google). Jadi kita bisa upload aplikasi sebanyak-banyaknya hanya dengan menggunakan satu akun google developer.

¹¹ Dibeberapa kasus proses pembayaran bisa memakan waktu 24 jam.



- Klik lanjutkan pendaftaran lalu isikan nama pengembang, alamat email, situs web jika ada (bisa juga menggunakan blog gratisan, dan nomor telepon aktif pada form profil pengembang.

Anda hampir selesai...

Cukup selesaikan detail berikut. Jika perlu, Anda dapat mengubah informasi ini nanti di setelan akun Anda.

Profil Pengembang	Bidang yang ditandai dengan * harus diisi sebelum disimpan.
Nama pengembang *	<small>0/50 Nama developer ditampilkan kepada pengguna di bawah nama aplikasi Anda. Perubahan pada nama developer akan ditinjau oleh Google dan prosesnya dapat memakan waktu hingga 7 hari.</small>
Alamat email *	
Situs web	
Nomor Telepon *	<small>Sertakan tanda plus, kode negara, dan kode area. Misalnya, +1-800-555-0199. Mengaga kami meminta nomor ponsel Anda?</small>

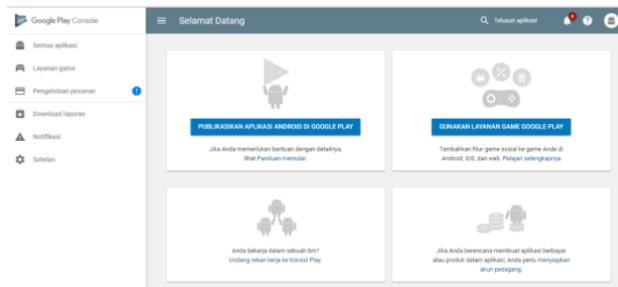
- Centang kedua preferensi email dibawah ini untuk mendapatkan update terbaru mengenai google console. Lalu klik tombol selesai pendaftaran

Preferensi email
<input type="checkbox"/> Saya ingin mendapatkan pengumuman fitur baru dan kiat untuk membantu menyempurnakan aplikasi.
<input type="checkbox"/> Saya ingin memberikan masukan untuk membantu menyempurnakan Google Play Developer Console.

Dengan mengakses Konsol Play, sesuai dengan Kebijakan Privasi Google dan Perjanjian Distribusi Developer, Anda setuju untuk berbagi informasi dengan Google tentang penggunaan Konsol Play. Data ini akan digunakan untuk pengembangan di masa mendatang dan personalisasi Konsol. Anda dapat menonaktifkan bagian data penggunaan di Setelan > Preferensi. Pelajari selengkapnya

SELESAIKAN PENDAFTARAN

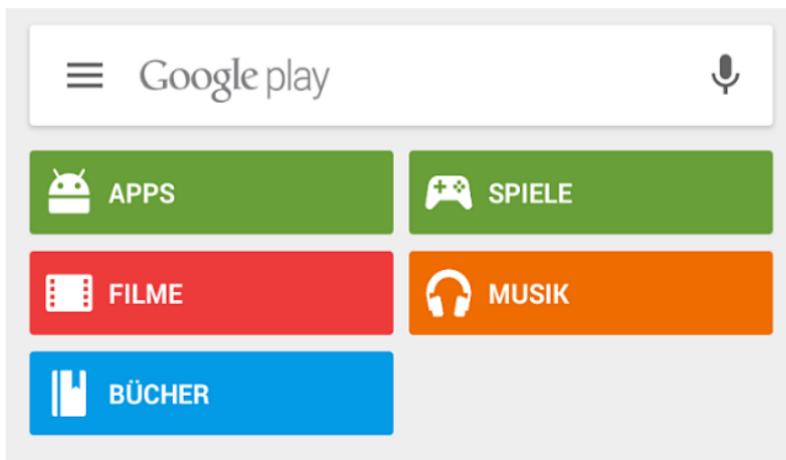
- Jika berhasil akan ditampilkan halaman utama dari Google play console.



Sampai disini kamu sudah memiliki akun google developer (Google play console).

BAB 16

RILIS APLIKASI KE GOOGLE PLAY STORE

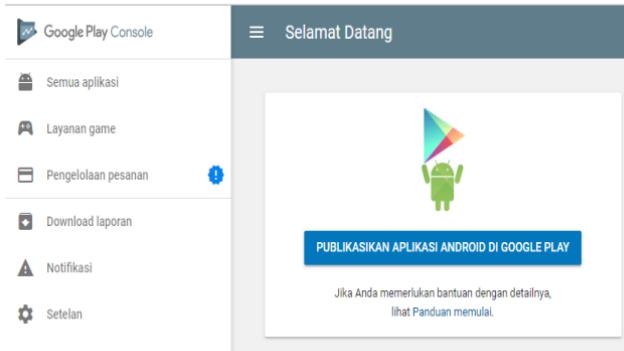


Rilis Aplikasi

Ini saatnya kita merilis aplikasi ke Google Playstore.

Caranya sangat mudah yaitu :

- **Masuk ke akun Google Play Developer Console** menggunakan akun gmail yang sudah didaftarkan seperti pada bab sebelumnya.
- Pada halaman utama **pilih menu publikasikan** aplikasi android di Google Play



- **Pilih bahasa** yang akan digunakan dan nama aplikasi android yang akan kita upload ke Google Play pada jendela buat aplikasi

Buat aplikasi

Bahasa default *

Indonesia – id

Judul *

Aplikasi Tebak Gambar |

22/50

BATAL BUAT

- Selanjutnya isikan deskripsi singkat dan lengkapnya, scroll ke bawah

Detail produk

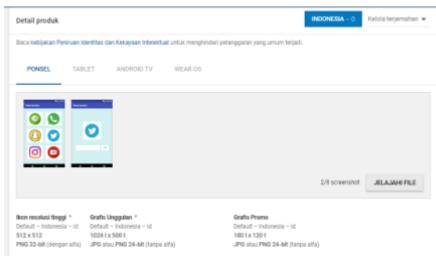
INDONESIA – Kelola terjemahan ▾

Bidang yang ditandai dengan * harus diisi sebelum dipublikasikan.

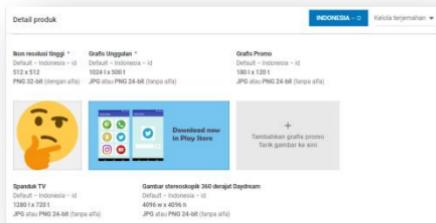
Judul *	Indonesia – id	Aplikasi Tebak Gambar	21/50
Deskripsi singkat *	Indonesia – id	Applikasi tebak gambar - Tebak icon	35/80
Deskripsi lengkap *	Indonesia – id	Applikasi tebak gambar sederhana ini berisi kumpulan icon, dimana nanti user harus bermanu dengan menebak nama dari icon yang ditampilkan.	

SIMPAN DRAFT

- Tambahkan screenshot aplikasi android yang sudah kita buat. Minimal dua screenshot. Saran saya masukan lebih dari satu agar lebih menarik user.



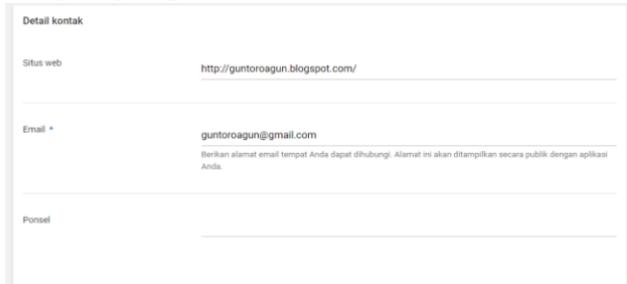
- **Tambahkan icon aplikasi, gambar unggulan, dan gambar untuk promosi jika ada.**



- **Selanjutnya tambahkan video promosi (link video youtube) jika ada. Pilih jenis aplikasi, kategori, dan rating konten.**

The screenshot shows the 'Detail produk' section with a 'Video Promo' field containing 'Video YouTube' and a 'Masukan URL' placeholder. Below this is a 'Kategorisasi' section with dropdown menus for 'Jenis aplikasi' (set to 'Game') and 'Kategori' (set to 'Pendidikan'). At the bottom, there's a note about rating content and a 'Rating konten' field.

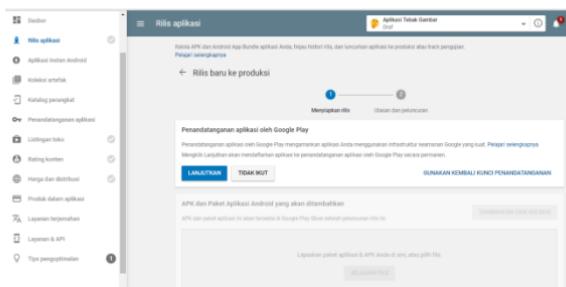
- Isi situs web (jika ada), alamat email, dan nomor telepon yang aktif.



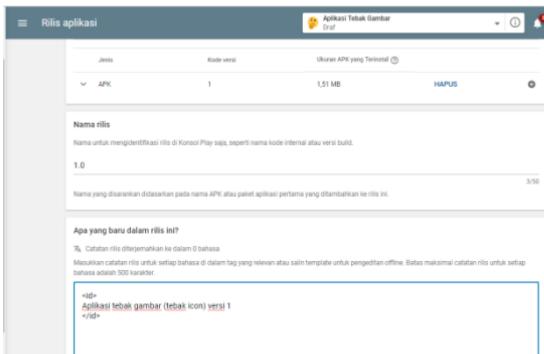
- Scroll lagi ke bawah lalu **isikan website untuk kebijakan privasi**. Kemudian **klik simpan draf**.



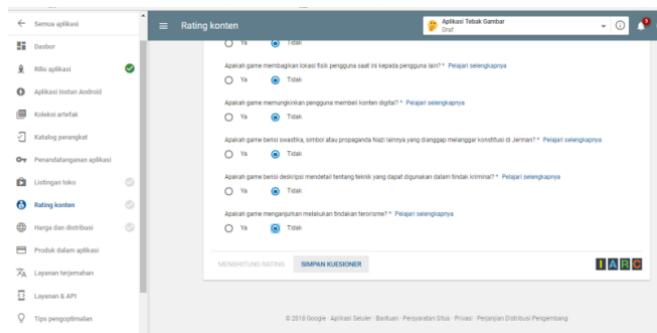
- Selanjutnya pilih menu **rilis aplikasi**



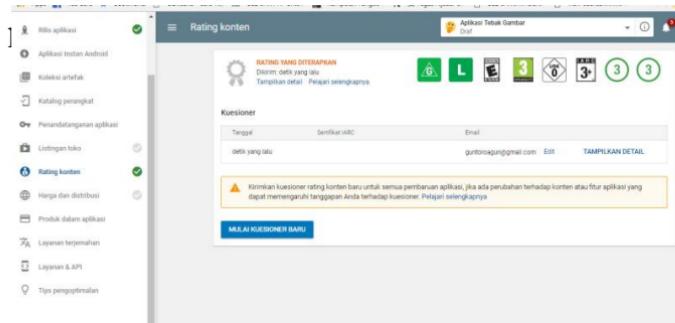
- Klik lanjutkan > terima > masukan file apk-release.apk. Jika berhasil akan tampil seperti dibawah ini. Kemudian tambahkan catatan. Klik Simpan



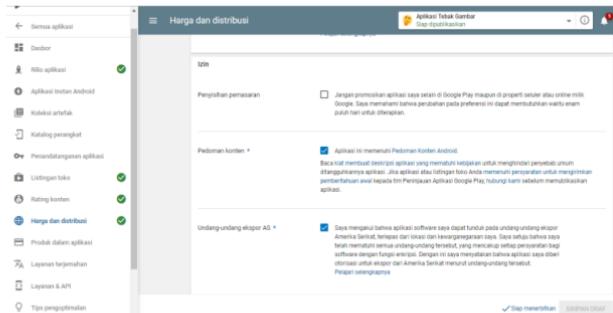
- Klik menu rating konten. Isikan email dan kategori aplikasi kita. Isi kuisioner lalu klik simpan.



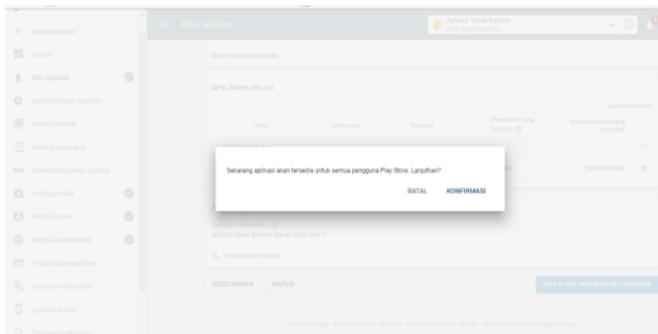
- Klik menghitung rating kemudian terapkan rating. Sehingga rating konten menjadi ceklis



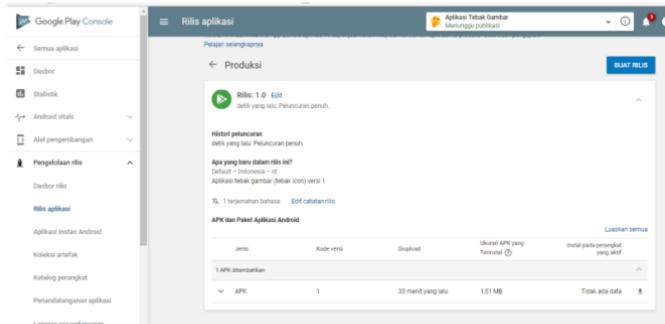
- Klik tab / menu harga dan distribusi. Isi Negara yang tersedia, ditujukan untuk pengguna anak. Scroll ke bawah centang aktifkan pada managed google play.
- Ceklis juga pada pedoman konten dan undang-undang ekspor AS. Lalu klik simpan



- Jika tab/ menu rilis aplikasi, rating konten, listing toko, dan harga distribusi sudah centang warna hijau. Maka aplikasi sudah bisa kita publish ke playstore
- Sekarang klik menu rilis aplikasi, pilih tinjau lalu klik mulai peluncuran produksi. Tekan



- Hasilnya aplikasi tebak gambar sudah di rilis ke play store

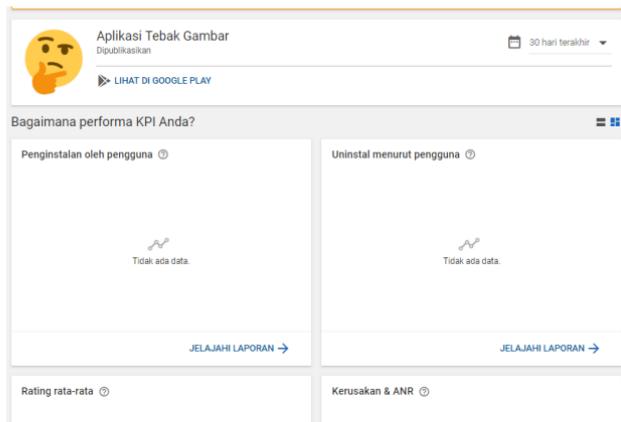


- Setelah itu aplikasi akan di review oleh pihak google selama 8-12 jam jika baru pertama kali upload ke google playstore. Selanjutnya akan memakan waktu kurang lebih 5 jam.
- Jika sudah berhasil status akan berubah menjadi dipublikasikan

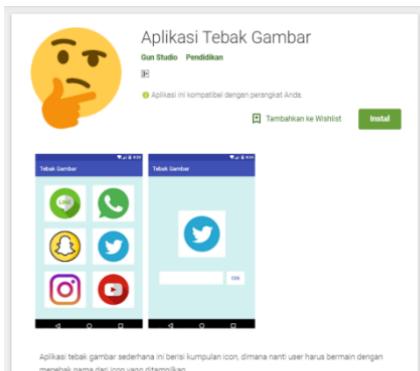
▲ Nama aplikasi	Instal aktif ⓘ	Rating rata2 / Total #	Terakhir diperbarui	Status
 Aplikasi Tebak Gambar badystudio.com.tebakgambar	-	★ -	31 Jul 2018	Dipublikasikan

Halaman 1 dari 1

- Klik jendela tersebut lalu akan muncul dashboard analityc dari aplikasi yang kita upload.



- **Klik lihat google play** untuk melihat aplikasi kita di play store.



Penulis



Perkenalkan saya Gunter. Saya merupakan pendiri Badoystudio.com yaitu web seputar pembelajaran membuat aplikasi android dan web kedepannya.

Hanya orang biasa yang kesehariannya mengajar di SMK, Blogger, dan trainer di London School Technopreneur.

Karena jiwa saya bebas tidak ingin banyak di atur oleh orang lain maka saya juga memilih menjadi freelancer dalam bidang web dan mobile development.

Untuk menghubungi saya perihal pekerjaan, sekedar menyapa, atau ingin memberikan kritik dan saran bisa melalui :

email : gunteragun@gmail.com

whatsapp : 085717474692