

Pertemuan 2

User Interface

XML Layout

A. Materi

- User interface, XML Layout

B. Kompetensi Dasar

- Mengenal macam layout

C. Tujuan

- Mahasiswa mengenal dan bisa menerapkan layout pada aplikasi android

D. Dasar Teori

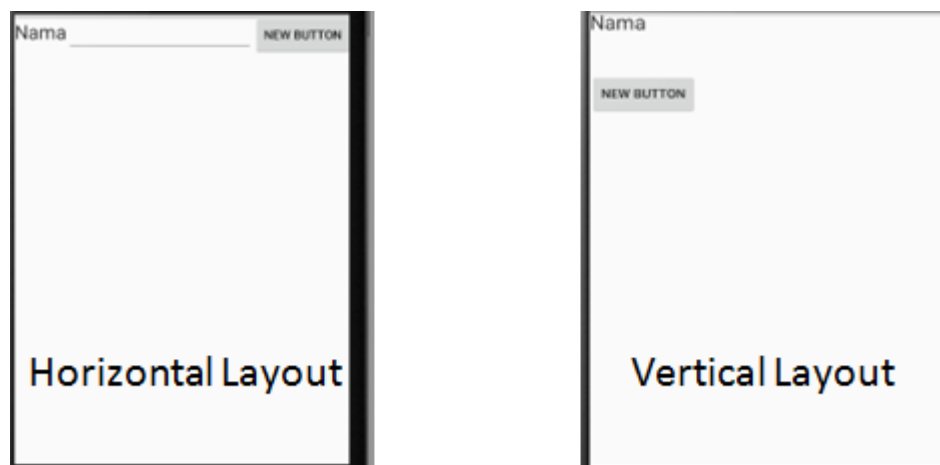
1. Membuat *Layout*

Layout mendefinisikan tampilan *visual* dari *User Interface*. Untuk membuat aplikasi yang akan diakses oleh user tentunya harus memiliki *user interface*. Dengan *user interface* user bisa berkomunikasi dengan aplikasi. Contohnya ketika *user* menginputkan datanya pada *EditText* kemudian menekan *button*, selanjutnya data yang diinputkan oleh *user* akan diproses oleh *system*.

EditText dan *button* ini disebut dengan *widget*. Sedangkan *widget* yang diletakkan di *user interface* harus menggantung pada *layout*. Pada modul ini kita akan belajar mengenai pembuatan *layout*. Di Android ada beberapa jenis *layout* diantaranya :

a. **LinearLayout**

LinearLayout adalah sebuah *ViewGroup* yang meletakkan elemen *widget* secara *vertical* atau *horizontal*. Pada **Gambar 2.1.** terlihat tampilan *user interface* yang menggunakan *LinearLayout*



Gambar 2.1. *User Interface* menggunakan *LinearLayout*

Pada horizontal layout komponen *widget* terletak secara *horizontal* atau sejajar. Sedangkan pada *VerticalLayout* komponen *widget* terletak secara *vertical*.

Untuk membuat LinearLayout ikuti langkah berikut :

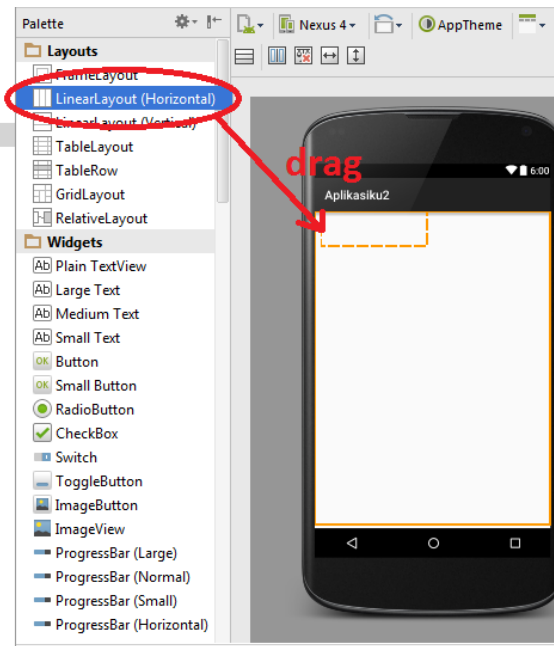
Buka android studio -> app ->res->layout->activity_main.xml pilih tampilan dalam bentuk design. Terlihat bahwa hp android menampilkan tulisan “hello World!”. Hapus tulisan tersebut. Dari tampilan design pindah ke tampilan text, sehingga akan terlihat list code dalam bentuk xml. Hal ini karena user interface android memakai XML. Ubah code yang ada menjadi berikut :

```
<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:orientation="horizontal"
android:id="@+id/idln"></LinearLayout>
```

Pada LinearLayout bisa diletakkan elemen-elemen widget. Berikut elemen LinearLayout yang telah diletakkan widget didalamnya

```
<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:orientation="horizontal"
android:id="@+id/idln"><TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:text="Large Text"
    android:id="@+id/textView"
    android:layout_alignParentTop="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />
<EditText
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/editText"
    android:layout_above="@+id/button"
    android:layout_toRightOf="@+id/button"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true" />
<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="New Button"
    android:id="@+id/button"
    android:layout_below="@+id/textView"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />
</LinearLayout>
```

Bisa pula membuat LinearLayout dengan mendrag Layout ke UI design terlihat seperti pada **Gambar 2.2.**



Gambar 2.2. Membuat LinearLayout

b. RelativeLayout

Buat *Layout* baru dengan cara klik kanan folder layout -> new -> layout resource file kemudian tuliskan nama layout yang baru, awali dengan huruf kecil -> klik ok

Berikut langkah untuk membuat *RelativeLayout* :

- Ganti tampilan menjadi design
- Drag *RelativeLayout* ke UI desain
- Drag *widget* dengan letak sesuai keinginanmu

Atau bisa dengan cara menambahkan coding pada layout. Caranya :

Ganti tampilan menjadi teks. Tambahkan *script* dibawah ini pada file .xml

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

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:text="Large Text"
        android:id="@+id/textView"
        android:layout_alignParentTop="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true" />

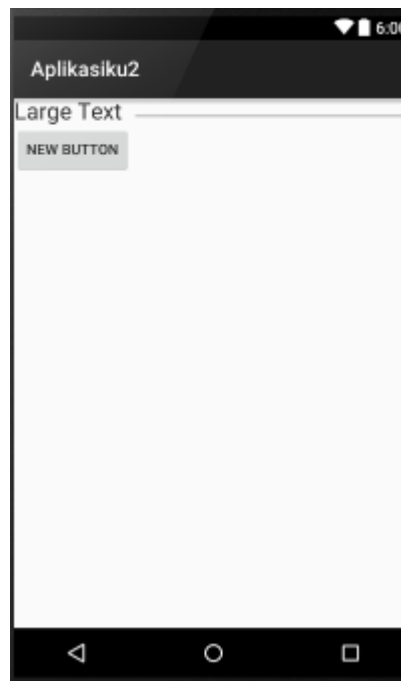
    <EditText
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/editText"
        android:layout_above="@+id/button"
```

```

        android:layout_toRightOf="@+id/button"
        android:layout_alignParentRight="true"
        android:layout_alignParentEnd="true" />
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="New Button"
        android:id="@+id/button"
        android:layout_below="@+id/textView"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true" />
</RelativeLayout>

```

Hasil terlihat seperti pada **Gambar 2.3**.



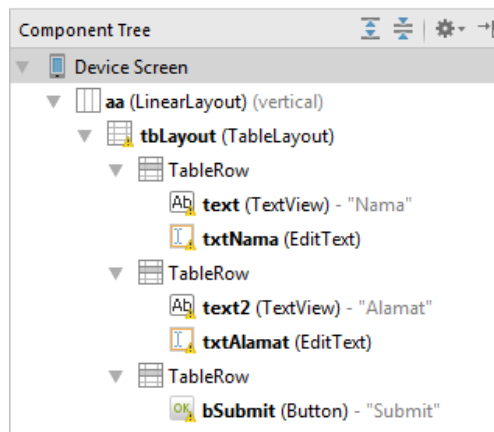
Gambar 2.3. Tampilan UI dengan *Framelayout*

c. **TableLayout**

TableLayout adalah sebuah *ViewGroup* yang meletakkan komponen pada baris dan kolom.

Untuk membuat komponen yang diletakkan pada *TableLayout* bisa dengan cara *drag*. Berikut cara membuat dengan tampilan desain :

- Drag *TableLayout* ke UI desain
- Drag *TableRow* ke UI desain
- Pindah ke mode text. Tambahkan **android:stretchColumns="1"** didalam komponen *TableLayout* seperti dibawah ini
- **<TableLayout**
- **android:stretchColumns="1">**
- Tambahkan *widget* ke komponen *TableRow*. Hingga *ComponentTree* terlihat seperti **Gambar 2.4**.

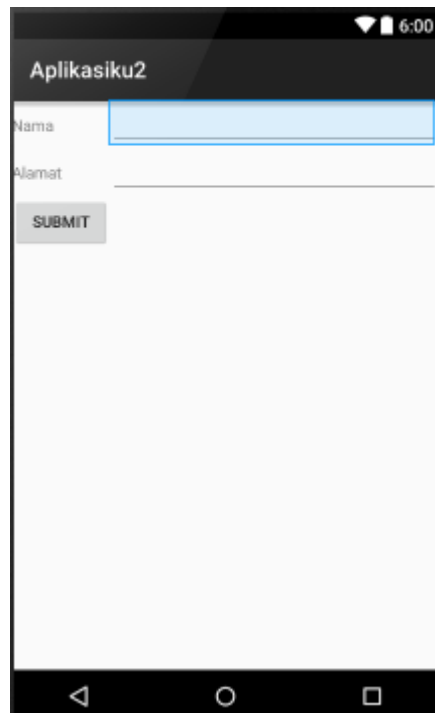


Gambar 2.4. Component Tree dari project dengan TableLayout

Atau bisa dengan meng-copikan script dibawah ini ke file .xml

```
<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:orientation="vertical"
android:id="@+id/aa">
<TableLayout
android:layout_width="match_parent"
android:layout_height="match_parent"
android:stretchColumns="1"
android:orientation="horizontal"
android:id="@+id/tbLayout">
<TableRow>
<TextView
android:text="Nama"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:id="@+id/text" />
<EditText
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:id="@+id/txtNama" />
</TableRow>
<TableRow>
<TextView
android:text="Alamat"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:id="@+id/text2" />
<EditText
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:id="@+id/txtAlamat" />
</TableRow>
<TableRow>
<Button
android:text="Submit"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:id="@+id/bSubmit" />
</TableRow>
</TableLayout>
</LinearLayout>
```

Hasil terlihat seperti pada **Gambar 2.5**.



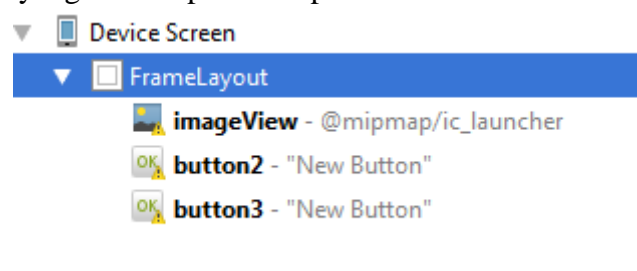
Gambar 2.5. Tampilan UI dengan Tablelayout

d. **FrameLayout**

FrameLayout adalah FrameLayout adalah frame yang dirancang untuk menyisipkan widget ke dalam widget lain

Untuk membuat komponen yang diletakkan pada FrameLayout bisa dengan cara drag. Berikut cara membuat dengan tampilan design :

- Drag FrameLayout ke UI design
- Drag komponen widget yang ingin diletakkan pada FrameLayout. Komponen yang diletakkan bisa seperti yang terlihat pada component tree **Gambar 2.6**.



Gambar 2.6. Component Tree dari project dengan FrameLayout

Atau bisa dengan cara, meng-copikan script dibawah ini ke file .xml

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

```

<ImageView
    android:layout_width="305dp"
    android:layout_height="305dp"
    android:id="@+id/imageView"
    android:layout_gravity="left|top"
    android:src="@mipmap/ic_launcher" />

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="New Button"
    android:id="@+id/button2"
    android:layout_gravity="left|top" />

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="New Button"
    android:id="@+id/button3"
    android:layout_gravity="right|top" />
</FrameLayout>

```

Hasil terlihat seperti pada **Gambar 2.7**.



Gambar 2.7. Tampilan UI dengan FrameLayout

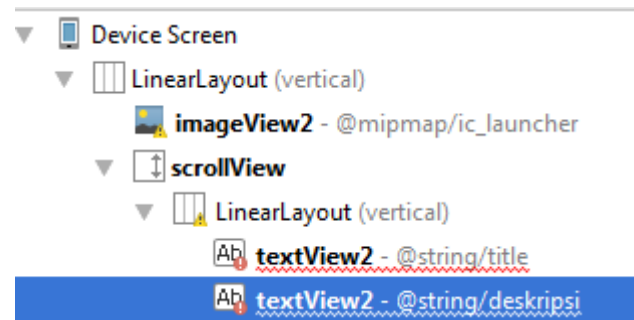
e. ScrollView

ScrollView adalah tipe dari FrameLayout yang memungkinkan user mescroll sebuah list dari view yang membutuhkan banyak space lebih dari tampilan fisik

Untuk membuat komponen yang diletakan pada ScrollView bisa dengan cara drag. Beikut cara membuat dengan tampilan design :

- Drag ScrollView ke UI design
- Drag LinearLayout ke dalam ScrollView
- Drag komponen widget ke dalam LinearLayout Komponen yang diletakkan bisa seperti yang terlihat pada component tree **Gambar 2.8**.

-



Gambar 2.8. Component Tree dari project dengan ScrollView

- Supaya textView berisi banyak tulisan seperti **Gambar 2.10.** bisa menggunakan nilai String yang disimpan di strings.xml
- Caranya buka values -> klik strings.xml

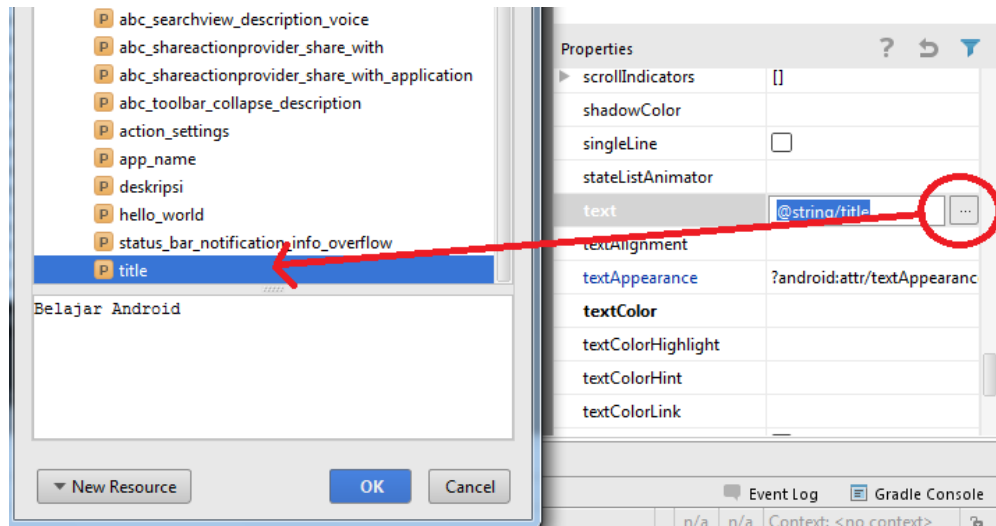
Di file ini menyimpan String atau tulisan yang bisa digunakan untuk membuat project seperti dibawah ini

```
<resources>
    <string name="app_name">Aplikasiku2</string>

    <string name="hello_world">Hello world!</string>
    <string name="title">Belajar Android</string>
    <string name="deskripsi">In this tutorial we will show you,
how to use Android ScrollView component and create a simple
example using various ScrollView properties.

Most Android application will likely to have the contents
that's doesn't fit the screen. Think a bit about displaying the
news details, the contents are dynamic and can grow beyond your
screen size. If we design our screen layout using standard
layout managers like LinearLayout, RelativeLayout, FrameLayout
or TableLayout; when the content grows, and data goes beyond
screen size, and user won't be able scroll and view the
content.</string>
    <string name="action_settings">Settings</string>
</resources>
```

Name merupakan key dari nilai String tersebut. Supaya bisa menggunakannya kembali ke file layout .xml, tentukan widget yang akan menggunakan String tersebut. Misal pada textView diatas, klik textView tersebut. Pada properties lakukan seperti pada **Gambar 2.9.**



Gambar 2.9. Memberi nilai text pada widget

- Lakukan hal yang sama pada textView kedua namun pilih *deskripsi*
- Bila bingung, copikan koding dibawah ini ke file .xml anda

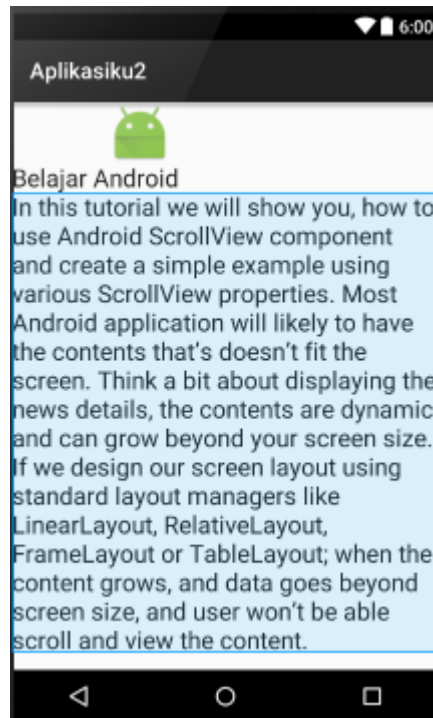
```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:weightSum="1">
    <ImageView
        android:layout_width="230dp"
        android:layout_height="wrap_content"
        android:id="@+id/imageView2"
        android:src="@mipmap/ic_launcher"
        android:layout_weight="0.39" />
    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/scrollView">
        <LinearLayout
            android:orientation="vertical"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="0.39">
            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"

                android:textAppearance="?android:attr/textAppearanceLarge"
                android:text="@string/title"
                android:id="@+id/textView2"
                android:layout_weight="0.39" />
            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"

                android:textAppearance="?android:attr/textAppearanceLarge"
                android:text="@string/deskripsi"
                android:id="@+id/textView2"
                android:layout_weight="0.39" />
        </LinearLayout>
    </ScrollView>
</LinearLayout>
```

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

- Hasil bisa dilihat pada **Gambar 2.10**.



Gambar 2.10. Tampilan UI dengan *ScrollView*

E. PRAKTIKUM

1. Lakukan langkah-langkah membuat *Linear Layout*, *TableLayout*, *RelativeLayout*, *FrameLayout* dan *ScrollView* Layout seperti pada dasar teori!
2. Buatlah layout untuk menampilkan form berikut :

Nama :
Alamat :
Email :

Submit

Reset

Gunakan *Relative Layout* untuk meletakkan widget serta gunakan *Linear Layout* untuk meletakkan dua buah button! Sehingga *Linear Layout* terletak didalam *Relative Layout*