

Combinación de layouts

Sumario

- 1 Introducción
- 2 Caso práctico
 - ◆ 2.1 Combinación de layouts Relativos e Liniais
 - ◆ 2.2 XML asociado ao Layout

Introdución

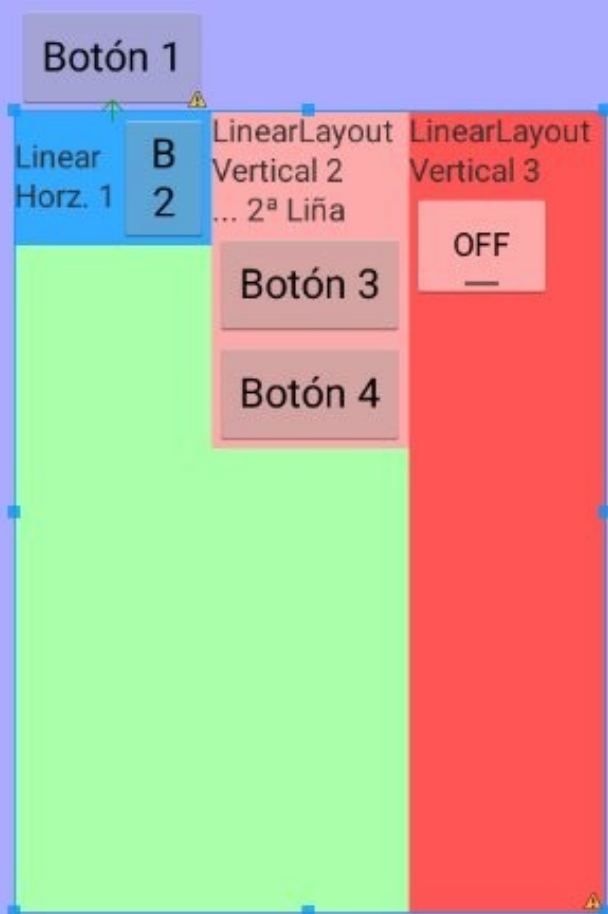
- Os layouts pódense combinar e aniñar uns dentro doutros.

Caso práctico

- Creamos un novo proxecto: U2_07_Combinados
 - ◆ **NOTA Edición 2015:** Existe un programa U2_07_ con outro nome da edición 2013, que faremos na sección de controis dentro desta unidade. Para non renumerar todo o que vén a continuación para a edición 2015, teremos 2 programas U2_07: U2_07_Combinados e U2_07_TextView (o faremos despois).

Combinación de layouts Relativos e Liniais

- A imaxe amosa a distribución que se quere ter:



- O **color malva** é un `LayoutRelative`, que ten un recheo de 30dp.
- Este layout contén:
 - ◆ Un botón: Botón 1
 - ◆ Un `LinearLayout` de **cor verde** (o que está cos bordes de cor azul cor verde é os cadradiños).
 - ◆ Este `LinearLayout` contén 3 `Linears Layouts` en disposición horizontal e cun peso igual para cada un deles para distribuír o ancho do que lle deixa o pai.
 - ◊ A altura deste `linearlayout` (cor verde) é a que lle deixa o pai.
 - ◊ Os tres layouts organízanse do seguinte modo:
 - O primeiro `LinearLayout` (**Cor azul**) ten unha disposición horizontal (as views dentro del están unha á dereita da outra)
 - A altura deste layout adáptase á altura dos elementos que contén.
 - O segundo `LinearLayout` (**Cor rosa**) ten nunha disposición vertical (as views dentro del están unha debaixo da outra).
 - A altura deste layout adáptase á altura dos elementos que contén.
 - O terceiro `LinearLayout` (**Cor vermello**) ten nunha disposición vertical (as views dentro del están unha debaixo da outra).
 - A altura deste layout adáptase á altura total do pai (`Layout` de cor verde).
- Este exemplo traballa só con `Relative` e `Linear` layouts, pero poderíase traballar con calquera outro layout dos vistos e non vistos neste curso.

XML asociado ao Layout

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#AAF"
    android:padding="30dp"
    tools:context="com.example.u2_07_combinados.U2_07_Combinados" >
```

```
<Button
    android:id="@+id/Boton1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Botón 1" />
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_below="@id/Boton1"
    android:background="#AFA"
    android:orientation="horizontal" >
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:background="#3AF"
    android:orientation="horizontal" >
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Linear\nHorz. 1" />
```

```
<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="B2" />
```

```
</LinearLayout>
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:background="#FAA"
```

```

        android:orientation="vertical" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="LinearLayout Vertical 2 \n... 2ª Liña" />

        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Botón 3" />

        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Botón 4" />
    </LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:background="#F55"
        android:orientation="vertical" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="LinearLayout Vertical 3" />

        <ToggleButton
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>
</LinearLayout>

</RelativeLayout>

```

- Explicación, liñas:

- ♦ **1 a 81**: onde se abre e se pecha o Relative Layout. Contén un botón e un linear layout que se abre na liña 15 e pécase na 79.
- ♦ **10**: Dáselle un ID ao botón.
- ♦ **15 a 81**: Ábrese un linear layout con disposición dos seus elementos (3 LinearLayouts) en horizontal (liña 20).
- ♦ **18**: Indica que o LinearLayout anterior se sitúe debaixo do botón.
- ♦ **22 a 38**: O primeiro LinearLayout que dispón os seus elementos en horizontal (Liña 27).
- ♦ **24, 42 e 65**: Observar como se indica a altura de cada un dos tres layouts. Nos dous primeiros casos que se adapte á altura dos elementos que contén e no terceiro á altura do pai.
- ♦ **25, 43 e 66**: Indícase o peso co que se distribúe cada un dos layouts (1/3 para cada un).
- ♦ **32 e 50**: Observar como se introduce un código de control (como en Java) para o salto de liña \n.

- O resto das liñas marcadas, a bo seguro que o participante no curso é quen de interpretalas.