

Xerar Views nun Layout en tempo de execución. Layout ScrollView

Sumario

Introdución

Caso práctico

- XML do layout
- código Java

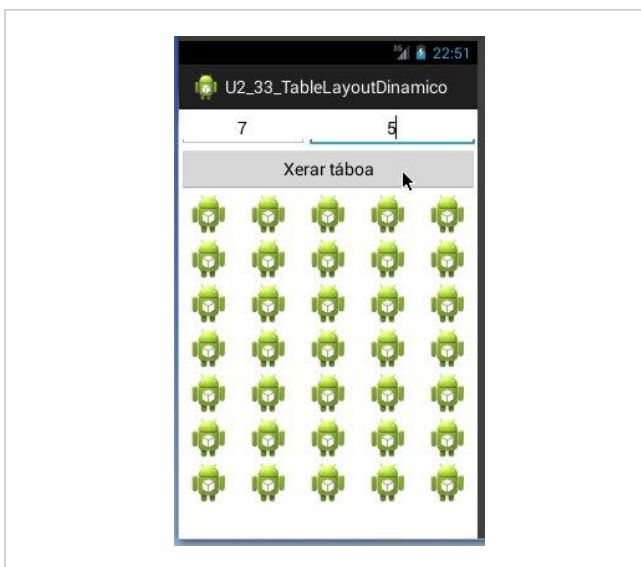
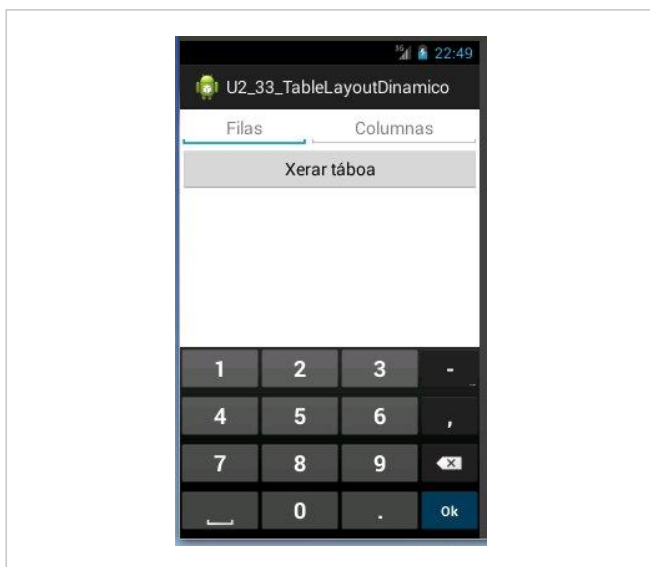
Introdución

- A continuación vanse crear os elementos dun TableLayout en tempo de execución.
- Os elementos dun TableLayout son as celas e cada unha vai ter un compoñente **ImageView**. Podería ser un botón, un texto, etc.
- Ademais a través do Layout **ScrollView** vaise permitir facer scroll.

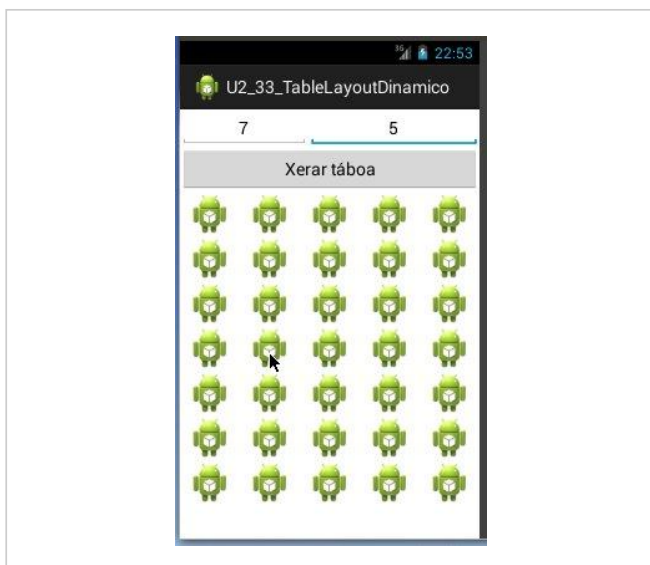
- Referencias
 - O control ScrollView: <http://developer.android.com/reference/android/widget/ScrollView.html>
 - O control TableLayout: <http://developer.android.com/reference/android/widget/TableLayout.html>

Caso práctico

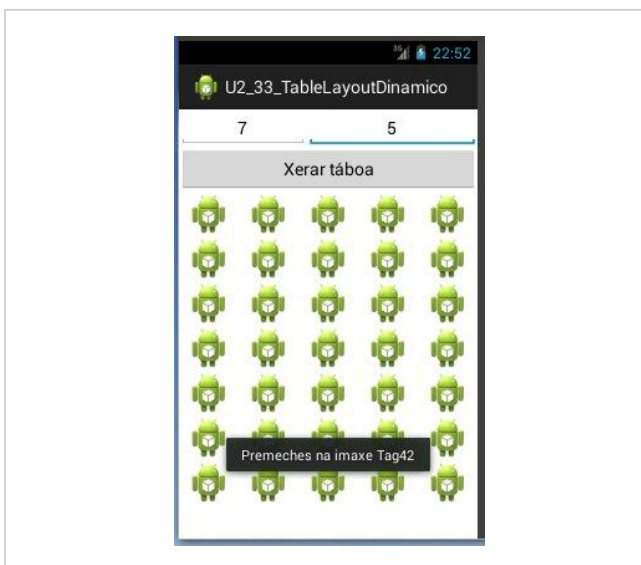
- Creamos unha novo proxecto: **U2_33_TableLayoutDinamico**
TableLayout Dinámico



Ao iniciar a aplicación introducimos o número de filas e columnas que desexamos que teña o TableLayout que está disposto debaixo do botón. Xeramos a táboa, e esta cárgase con ImageViews cuxa imaxe é a icona da aplicación.



Se prememos sobre unha imaxe.



Aparece un Toast amosando o seu Tag que indica a fila e columna na que está esa imaxe.

O XML do layout

- Imos centrarnos no novidoso:

```

1 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
2   xmlns:tools="http://schemas.android.com/tools"
3   android:layout_width="match_parent"
4   android:layout_height="match_parent"
5   android:orientation="vertical" >
6
7   <TableLayout
8     android:layout_width="wrap_content"
9     android:layout_height="wrap_content"
10    android:stretchColumns="*" >
11
12     <TableRow>
13
14       <EditText

```

```

15         android:id="@+id/et_filas"
16         android:layout_width="wrap_content"
17         android:layout_height="wrap_content"
18         android:hint="Filas"
19         android:gravity="center"
20         android:imeOptions="actionDone"
21         android:inputType="number" />
22
23     <EditText
24         android:id="@+id/et_columnas"
25         android:layout_width="wrap_content"
26         android:layout_height="wrap_content"
27         android:gravity="center"
28         android:hint="Columnas"
29         android:inputType="number" />
30 </TableRow>
31 </TableLayout>
32
33 <Button
34     android:layout_width="match_parent"
35     android:layout_height="wrap_content"
36     android:onClick="onXerarTaboaClick"
37     android:text="Xerar táboa" />
38
39 <ScrollView
40     android:layout_width="wrap_content"
41     android:layout_height="wrap_content" >
42
43     <TableLayout
44         android:id="@+id/taboa_dinamica"
45         android:layout_width="match_parent"
46         android:layout_height="match_parent"
47         android:stretchColumns="*" >
48     </TableLayout>
49 </ScrollView>
50
51 </LinearLayout>

```

- **ScrollView:** é un layout que permite activar o scroll vertical. Co cal, os elementos que haxa dentro poden crecer en vertical ao seu antollo, pois sempre se poderá realizar scroll.
 - Dentro dese Layout definiuse un TableLayout que non ten ningunha fila. Estas crearanse en tempo de execucion.
- **Liña 44:** observar que a táboa ten un ID para poder acceder a ela dende Java.

O código Java

```

1 package com.example.u2_33_tablelayoutdinamico;
2
3 import android.app.Activity;
4 import android.os.Bundle;
5 import android.view.Menu;
6 import android.view.View;
7 import android.view.View.OnClickListener;
8 import android.view.WindowManager;
9 import android.widget.EditText;
10 import android.widget.ImageView;
11 import android.widget.TableLayout;
12 import android.widget.TableRow;
13 import android.widget.Toast;
14
15 public class U2_33_TableLayoutDinamico extends Activity {
16
17     @Override
18     protected void onCreate(Bundle savedInstanceState) {
19         super.onCreate(savedInstanceState);
20         setContentView(R.layout.activity_u2_33_table_layout_dinamico);
21     }
22
23
24     public void onXerarTaboaClick(View v) {
25         TableLayout taboa = (TableLayout) findViewById(R.id.taboa_dinamica);
26         EditText filas = (EditText) findViewById(R.id.et_filas);
27         EditText columnas = (EditText) findViewById(R.id.et_columnas);
28
29         int f = Integer.parseInt(filas.getText().toString());
30         int c = Integer.parseInt(columnas.getText().toString());
31
32         taboa.removeAllViews();
33
34         for (int i = 1; i <= f; i++) {
35             TableRow fila = new TableRow(this);
36             taboa.addView(fila);
37

```

```

38     for (int j = 1; j <= c; j++) {
39         ImageView imaxe = new ImageView(this);
40         imaxe.setTag("Tag" + i + j);
41         imaxe.setImageResource(R.drawable.ic_launcher);
42         fila.addView(imaxe);
43         imaxe.setOnClickListener(new OnClickListener() {
44
45             @Override
46             public void onClick(View v) {
47                 // TODO Auto-generated method stub
48                 ImageView img = (ImageView) v;
49                 Toast.makeText(getApplicationContext(), "Premeches na imaxe " + img.getTag(), Toast.LENGTH_SHORT).show();
50             }
51         });
52     }
53 }
54 }
55 }
56
57 @Override
58 public boolean onCreateOptionsMenu(Menu menu) {
59     // Inflate the menu; this adds items to the action bar if it is present.
60     getMenuInflater().inflate(R.menu.u2_33_table_layout_dinamico, menu);
61     return true;
62 }
63
64 }

```

- **Liña 25:** no obxecto **taboa** apuntamos á táboa definida no layout.
- **Liña 29,30:** f e c indican cantas filas e columnas debe ter o **TableLayout**. Lembrar que **parseInt()** é un método estático da clase **Integer** en **Java**.
- **Liña 32:** eliminamos todas as vistas que poida ter a táboa.
- **Liña 35,36:** engadimos tantos **TableRow**s como filas nos indicou o usuario.
- **Liña 39,40,41:** creamos un obxecto **ImageView** co seu **Tag** e imaxe asociada.
- **Liña 42:** engadimos en cada fila tantas columnas como nos indicou o usuario a través de engadir, nese caso vistas do tipo **ImageView**.
- **Liña 43 a 51** para cada unha das imaxes chamamos ao seu **Listener** que estará a espera de que se faga click nesa mesma imaxe que se acaba de crear.
- **Liña 46 a 50** cando se preme nunha imaxe podemos coñecer, neste caso, cal é o seu **Tag**.

-- Ángel D. Fernández González e Carlos Carrión Álvarez -- (2015).

Obtenido de «https://manuais.iessanclemente.net/index.php?title=Xerar_Views_nun_Layout_en_tempo_de_execución._Layout_ScrollView&oldid=57134»

Esta página se editó por última vez el 28 jul 2015 a las 09:11.

El contenido está disponible bajo la licencia [Creative Commons: CC-BY-NC-SA](#), a menos que se indique lo contrario.