|  |  |  |
| --- | --- | --- |
|  | | **Departamento de Engenharia Informática**  **TeSP em Programação de Sistemas de Informação**  **Acesso Móvel a Sistemas de Informação**  2016/2017 |
|  |  |  |
|  |  |  |
| **Docentes** | Sérgio Lopes, sergio.lopes@ipleiria.pt | |
|  | David Safadinho, david.safadinho@ipleiria.pt | |

Ficha de Exercícios N/03

ListView e GridView

# Geral

Listas e grelhas são dois elementos de interação com utilizador bastante usados em dispositivos móveis. De tal forma que a plataforma *Android* possui dois layouts/grupos dedicados a este tipo de interação.

Uma *ListView* permite a disposição de elementos numa lista com deslocamento vertical, sendo responsável pela gestão e controlo dos elementos visíveis, dos que ficam fora da área visível e da reutilização dos objetos criados. Do mesmo modo, uma *GridView* permite a apresentação de elementos com toda a gestão a cargo do componente, sendo os elementos dispostos numa grelha bidimensional (linhas e colunas).

## *Adapters*

*ListView* e *GridView* são dois (entre outros) componentes que requerem a utilização de *Adapters* para apresentarem os dados. Um *adapter* (adaptador) é responsável por fazer a ponte entre os dados a serem apresentados e a forma como estes dados são apresentados. Isto é, se olharmos para a API de uma *ListView*, não há forma de configurarmos os componentes visuais; não temos como dizer se queres apresentar imagens, texto, qual o alinhamento e tamanho da imagem, etc.; para isso precisamos de criar uma vista nova que a *ListView* vai repetir, mas essa vista não tem dados, apenas os componentes visuais; assim, precisamos também de um adaptador que configure a subvista que a *ListView* usa.

# Exercício 1 – ListView

Crie um projeto com as seguintes características:

* Nome do projeto: List+GridLayout
* SDK: API 19
* Modelo: Empty Ativity
* Nome da atividade: ListaAlunos
* Nome do layout: lista\_alunos

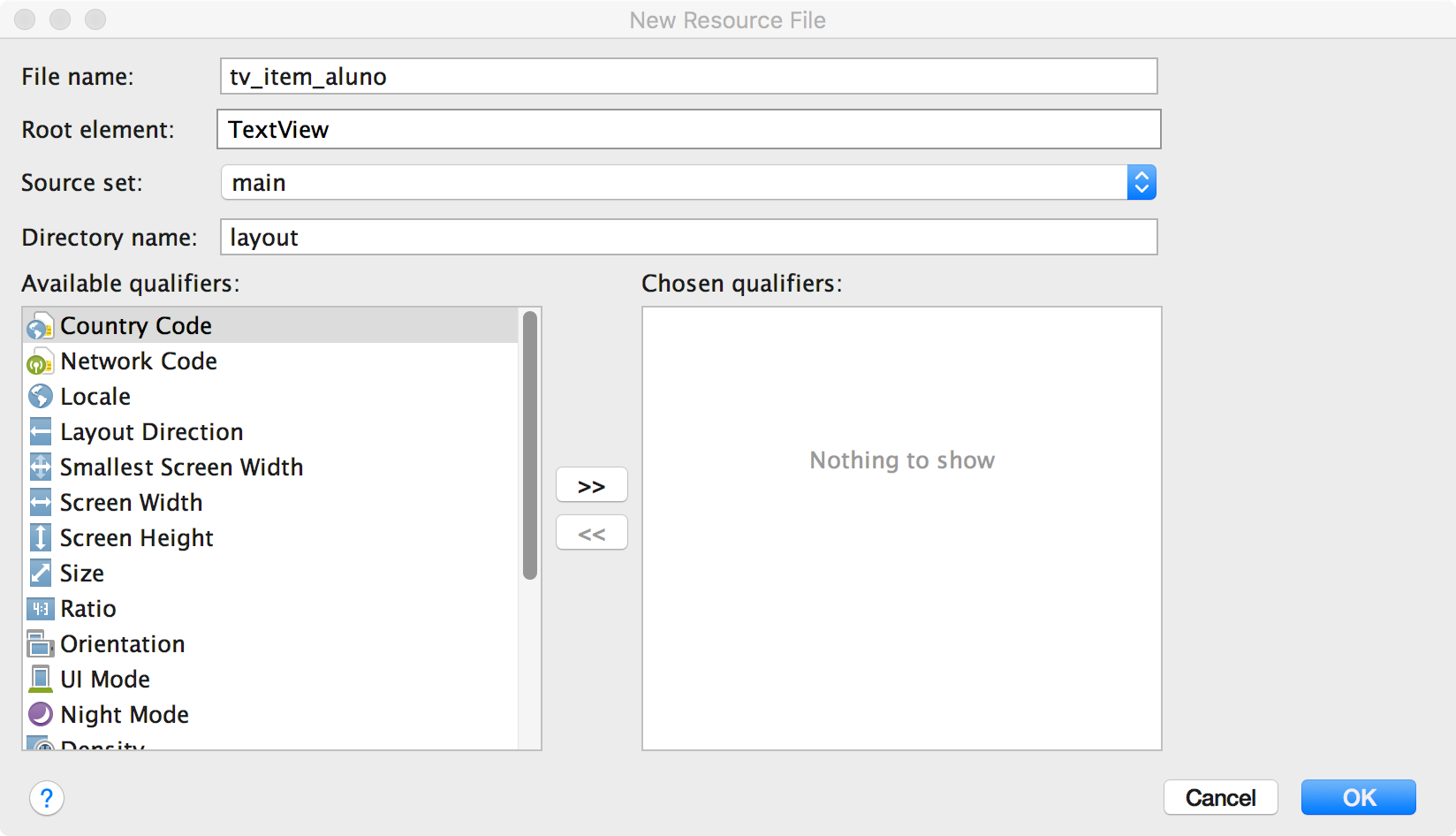
Depois de criar o projeto, altere o layout base para um **LinearLayout** e apague a **TextView** que foi criada automaticamente. Adicione uma *ListView* que ocupe o espaço todo visível (**match\_parent** em altura e largura).

Neste momento temos a nossa lista, mas ela não faz nada de especial. Para apresentar alguma informação precisamos de:

1. Dados a apresentar, vamos usar uma lista de nomes de pessoas (alunos);
2. Adaptador que entenda a nossa lista de nomes e que saiba como apresentar esta informação
   1. Um *layout* para representar os itens repetidos da lista;
3. Configurar o adaptador e a lista;

Adicione um novo layout clicando com o botão direito na pasta “**app/res/layout**”, escolhendo a opção “**New > Layout resource file**” e configurando as opções de acordo com a imagem seguinte.

Figura 1 – Interface para criação de ficheiro de layout.



Este *layout* tem apenas uma *TextView* e é este componente que será usado pela *ListView* (repetido as vezes que for necessário) para construir a lista de elementos. Mas ainda não temos quaisquer dados a apresentar.

No método **onCreate** da atividade crie um novo adaptador (**ArrayAdapter**), uma lista de **String** e introduza alguns valores exemplo:

**alunos** = **new** ArrayList<>(24);

**alunos**.add(**"João"**);

**alunos**.add(**"Sérgio"**);

**alunos**.add(**"Mário"**);

**alunos**.add(**"Luís"**);

ArrayAdapter<String> adapter = **new** ArrayAdapter<String>(**this**, R.layout.***tv\_item\_aluno***, **alunos**);

Configure o adaptador na lista através do método **setAdapter**.

# Exercício 2 – GridView

* Adicione uma segunda atividade ao projeto “**New > Activity > Empty Activity**” e altere o *layout* para um **LinearLayout**. Tal como a *ListView*, a *GridView* requer a utilização de um adaptador e neste caso iremos criar o nosso próprio adaptador em vez de usarmos um adaptador já fornecido pelo sistema *Android*.
* Extraia as imagens presentes no arquivo ZIP (no *Moodle*) e coloque-as na pasta “**app/res/drawable**”. Estas imagens serão usadas para criar a nossa grelha, são os nossos dados a mostrar.
* Crie uma nova classe, no package onde se encontram as classes das atividades, com os seguintes parâmetros:
* Nome: StaticFotoAdapter
* Subclasse de: BaseAdapter
* Implemente os métodos todos em falta usando o assistente (lâmpada vermelha à esquerda da linha/erro)

**public class** StaticFotoAdapter **extends** BaseAdapter {

**private** Context **mContext**;

**private** Integer[] **fotos**;

**public** StaticFotoAdapter(Context c) {

**mContext** = c;

}

**public int** getCount() {

**return** (**fotos** != **null** ? **fotos**.**length**: 0);

}

**public** Object getItem(**int** position) {

**return null**;

}

**public long** getItemId(**int** position) {

**return** 0;

}

**public void** setFotosArray(Integer[] fotos) {

**this**.**fotos** = fotos;

}

**public** View getView(**int** position, View convertView, ViewGroup parent) {

ImageView imageView;

*// verificar se estamos a reciclar a view ou não.*

**if** (convertView == **null**) {

imageView = **new** ImageView(**mContext**);

*// configurar parâmetros de layout, tal como se faria usando XML*

imageView.setLayoutParams(**new** GridView.LayoutParams(85, 85));

imageView.setScaleType(ImageView.ScaleType.***CENTER\_CROP***);

imageView.setPadding(8, 8, 8, 8);

} **else** {

imageView = (ImageView) convertView;

}

imageView.setImageResource(**fotos**[position]);

**return** imageView;

}

}