



## Android - Cards de Referência

### Estrutura de um Projeto

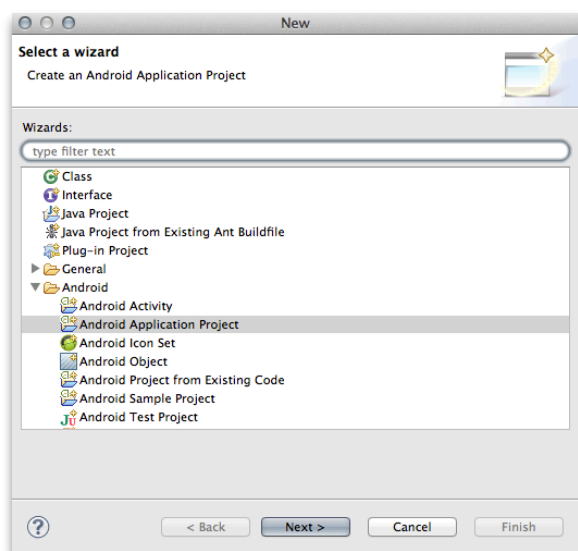
Bem vindo ao nosso segundo card sobre Android! Desta vez, vamos entender a estrutura de um projeto Android. Vamos iniciar a criação de um projeto e entender o passo a passo e o motivo de existir de cada pasta e arquivo.



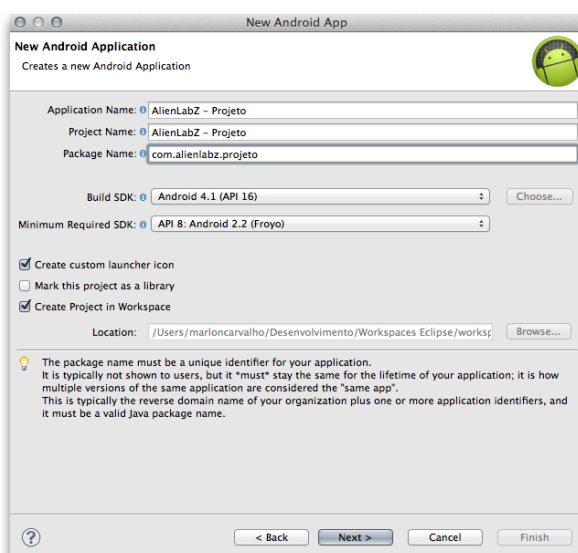
Por Marlon Silva Carvalho  
[marlon.carvalho@gmail.com](mailto:marlon.carvalho@gmail.com)  
 @marlonscarvalho

### CRIANDO UM PROJETO

Vamos criar nosso primeiro projeto para Android? Com o Eclipse aberto e já com o plugin ADT instalado, conforme fizemos no Card #1, vamos selecionar a opção **File > Other**. No Wizard que será aberto, procure a opção **Android Application Project**.



Você será guiado através de um Wizard que lhe fará algumas perguntas sobre como você quer seu projeto. Deste Wizard, inicialmente vamos prestar mais atenção para a tela a seguir.

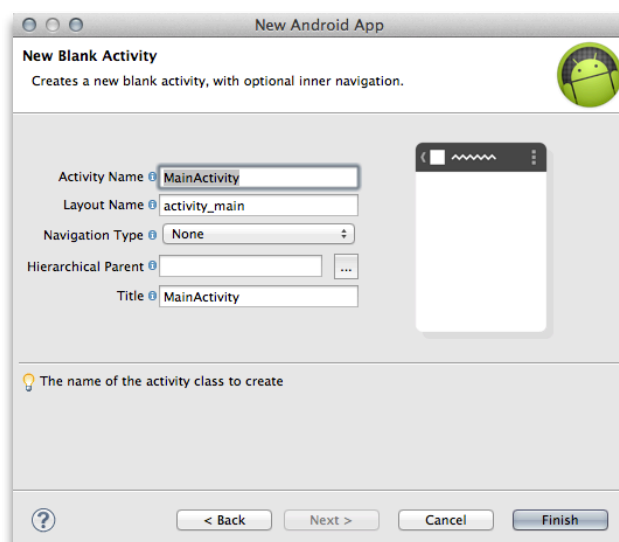


Precisamos dar um nome para nossa aplicação, através do campo **Application Name**. Depois, precisamos também nomear o nosso projeto, usando o campo **Project Name**. Estranhou? Parece que são dois campos com o mesmo objetivo, não é? Concordo. Mas são diferentes. O primeiro é o nome de sua aplicação mesmo, ou seja, o nome que será usado na publicação do Google Play. O segundo é apenas o nome do projeto no Eclipse.

Agora, você precisa informar o **Package Name**. Atenção! Esta informação é importante, pois serve como um identificador da sua aplicação. Este nome deve ser único, inclusive dentro do próprio Google Play. Suponha que você faça uma primeira versão do aplicativo e na segunda versão, altera este campo. O Google Play considerará como uma nova aplicação e não atualização da anterior. Entendeu?

Dos demais campos, observe o checkbox com o nome **Mark this project as library**. Deixe desmarcado. Ele serve apenas para o caso em que você deseja criar bibliotecas específicas para o Android. Um dia você chega lá! :-)

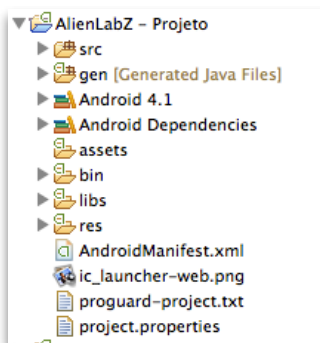
Clique em **Next**. Quer um ícone diferente? Informe nesta tela. Senão, clique em **Next** mais uma vez. Selecione agora a opção **Blank Activity** e avance de novo. Nesta última tela, deixe as informações que foram sugeridas para a criação da Activity.



São informações básicas de uma Activity, como o nome, o nome do arquivo em XML que define a estrutura da tela, o tipo de navegação, o pai e o título.

## ESTRUTURA DO PROJETO

Agora você já deve um projeto criado e com a estrutura padrão que todo projeto em Android deve ter. Vamos começar pelas pastas.



**src/** - o nome já diz tudo, aqui vão os códigos fonte em Java. Você terá aqui suas classes de Activity, suas regras de negócio e etc.

**gen/** - pasta gerada automaticamente e gerenciada pelo próprio plugin ADT. Sugestão? Nunca altere o conteúdo dela ou dos arquivos contidos nela.

**assets/** - uma pasta específica para que você coloque arquivos de recursos como sons e filmes. Coloque nesta pasta somente recursos que não precisam de internacionalização ou que não necessitem se adaptar conforme características do dispositivo, como tamanho de tela ou versão do sistema operacional.

**libs/** - coloque aqui as bibliotecas externas que você for usar. Enfim, aqui sempre vão os arquivos .jar de outros projetos.

**res/** - aqui vão os recursos do projeto, como imagens, layout, textos, cores, dimensões e por aí vai. Use esta pasta para todos os recursos em que você precisa de internacionalização e que necessitam se adaptar conforme características do dispositivo e do sistema operacional.

**AndroidManifest.xml** - é um arquivo especial em seu projeto. Ele traz informações importantes, como qual a Activity de inicialização, os serviços, as permissões, nome do pacote, nome da aplicação e muitas outras. Na verdade, este arquivo merece um card feito especialmente para ela. :-)

**proguard-project.txt** - você sabia que todo projeto Android tem seu código fonte ofuscado? É regra. É usado o ProGuard por padrão e este arquivo contém as informações básicas para se fazer a ofuscação. Caso queira alterar algo neste aspecto, este é o arquivo.

**project.properties** - tem algumas informações sobre o projeto, mas não se preocupe com ele, pois é usado somente pelo plugin ADT. Deixe ele em paz.

## A PASTA DE RECURSOS

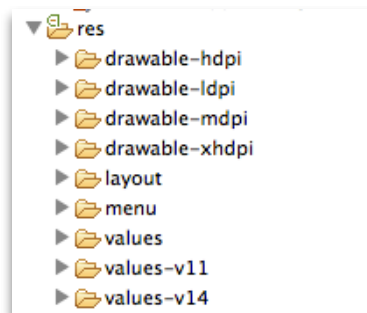
Vamos prestar um pouco mais de atenção agora para a pasta **/res**. Conforme já explicamos, ela contém os recursos do seu projeto que precisam se adaptar conforme características do dispositivo e do SO.

Por exemplo, você pode querer usar uma imagem diferente para cada tamanho de tela. Ou pode querer que seu aplicativo tenha os textos para duas línguas: inglês e português.

Você verá que estas pastas são compostas por um sufixo. Em alguns casos, temos um sufixo que indica a densidade de pixels, como no caso de **drawable-ldpi**. Em outros, temos um sufixo que identifica a versão do sistema operacional, como em **values-v11**.

Ainda temos situações em que temos pastas com o sufixo de um Locale. Por exemplo, **values-enUS**. E o que você acha de ainda termos sufixos para identificar o tamanho de tela? Para este caso, temos pastas como **drawable-xlarge**, onde você coloca imagens para tablets.

Ah! Quer tornar as coisas ainda mais interessantes? Você pode misturar isto tudo! Imagine uma pasta assim: **drawable-mdpi-xlarge-enUS-v11**. É possível? É, sim. Essa pasta terá imagens para serem usadas em dispositivos com densidade média de pixels e que são tablets. Ainda por cima, somente tablets com locale dos Estados Unidos e com, no mínimo, a versão 3.0 do Android. Muito louco, não? :-)



**drawable-\*/** - use estas pastas para colocar imagens. Você terá, por padrão, as pastas **drawable-ldpi**, **drawable-mdpi**, **drawable-hdpi** e **drawable-xhdpi**. Cada uma se refere a uma densidade de pixels

específica. Você também pode ter a pasta **drawable**, que é independente de DPI.

**layout-\*/** - aqui você terá as definições de layout de suas telas. No Android, você constrói suas telas, preferencialmente, usando arquivos em XML. Estes arquivos ficam aqui.

**menu-\*/** - já viu aqueles aplicativos que tem menus? Pois então, você define como serão e quais as opções do menu usando arquivos em XML que ficam nesta pasta.

**values-\*/** - você sabe que não é boa prática colocar textos diretamente no código Java, certo? Você coloca esses textos em arquivos contidos nesta pasta. O primeiro deles é o **strings.xml**.

**DICA**

Os arquivos de recurso devem ser sempre em minúsculas e não podem conter a maioria dos caracteres especiais. Apenas o \_ (sublinhado).

Mas você também pode encontrar outros tipos de arquivos aqui, como **color.xml**, **dimens.xml** e **styles.xml**. Os nomes deles já dizem tudo, não é? O primeiro é para você definir cores, o segundo dimensões e o último estilos mais gerais da aplicação.

Você também pode encontrar estes arquivos em pastas separadas. Em alguns casos, existe a pasta **/color/\*.xml**. Em outros, este arquivo fica na própria pasta **/values**.

**anim-\*/** - contém arquivos XML que definem animações pré-determinadas.

Quer ler o que o próprio Google diz sobre os recursos? Acesse o link <http://developer.android.com/guide/topics/resources/available-resources.html> que tem muita informação interessante.

**R.JAVA**

Agora que você tem um monte de recursos (imagens e afins) nas subpastas de **/res**, como eu uso eles no meu código Java? Não se preocupe. É criado automaticamente para você um arquivo contendo constantes que referenciam esses recursos.

Estas constantes estão no arquivo **R.java**, que se encontra naquela pasta **/gen**, que comentamos anteriormente. Este arquivo não é muito intuitivo. Mas esse não é o objetivo mesmo. Ele tem uma constante para cada arquivo de recurso.

Observe que este arquivo é composto de “subseções”. Por exemplo. **R.strings**, **R.drawable**, **R.color**, **R.menu** e assim em diante. Tudo para facilitar o acesso a recursos específicos a partir do código fonte.



## Sobre o Autor

### Marlon Silva Carvalho

É um programador de longa data. Já peregrinou por diversas linguagens e hoje se considera um programador agnóstico. Atualmente, está fascinado pelo mundo mobile e suas consequências para a humanidade. Está mais focado no desenvolvimento para Android, embora também goste de criar aplicativos para iOS.

Trabalha em projetos sobre mobilidade no SERPRO, tendo atuado no projeto Pessoa Física, para a Receita Federal do Brasil. Você pode encontrar ele no Twitter, no perfil @marlonscarvalho, em seu blog, no endereço <http://marlon.silvacarvalho.net> e através do e-mail [marlon.carvalho@gmail.com](mailto:marlon.carvalho@gmail.com).

## Sobre os Cards

Caso você já tenha visto os excelentes RefCards da DZone, deve ter percebido a semelhança. E é proposital. A ideia surgiu após acompanhar estes RefCards por um bom tempo. Contudo, eles são mais gerais. O objetivo destes Cards é tratar assuntos mais específicos. Trazendo informações relevantes sobre eles.

Caso tenha gostado, continue nos acompanhando! Tentaremos trazer sempre novos cards sobre os mais variados assuntos. Inicialmente, focaremos no Android.

Este trabalho usa a licença **Creative Commons Attribution-NonCommercial-ShareAlike 3.0**

<http://creativecommons.org/licenses/by-nc-sa/3.0/>



Basicamente, você pode copiar e modificar, desde que redistribua suas modificações. Também não é permitido usar este material ou suas derivações de forma comercial.