# SMActivity

A Activity é responsável pela implementação do menu partilhado pelas Activities ‘UserStatusActivity’ e ‘TimelineActivity’ (**S**hared**M**enu**Activity**).

Esta activity extende a superclasse Activity e implementa apenas três métodos: **onCreate**, **onCreateOptionsMenu** e **onOptionsItemSelected**.

No método **onCreate** é chamado o mesmo método da superclasse e iniciada a variável de instância **MyApplication app**, *protected*, para uso nas classes derivadas.

O método **onCreateOptionsMenu** é responsável por fazer o *inflate* do menu, passando o ficheiro .xml que contém o layout do menu, menu.xml .

Este ficheiro .xml contém 4 itens:

* Carregar as preferências (UserPreferencesActivity);
* Iniciar a timeline (TimelineActivity);
* Escrever um novo tweet (UserStatusActivity);
* Fazer a actualização da timeline;

No método **onOptionsItemSelected** é onde se atribui uma acção aos diversos itens que estão no menu, ou seja, iniciar as Activities consoante seja premido o item respectivo. A acção de actualização da timeline é feita na própria Activity, que ao extender SMActivity, irá fazer *override* deste método, chamando este (método da superclasse), e adicionando uma acção a este item.

# DetailActivity

Esta Activity extende SMActivity e é bastante simples, contendo apenas dois métodos: **onCreate** e **init**, sendo o último chamado dentro do primeiro.

O primeiro método, analogamente a todas as Activities, começa por chamar o onCreate da superclasse. De seguida carrega o ficheiro detail.xml correspondente ao layout da Activity.

Este ficheiro contém:

* Um ImageView, onde será mostrada a foto do utilizador;
* Três TextView, encarregues por mostrar o nome do utilizador, a data e a mensagem em causa.

Por último é chamado o método **init**.

Este método é responsável por iniciar as Views do layout, e começa primeiro por se criar, através do método **getParcelable,** um objecto do tipo DetailsModel, que contém a informação da informação a ser mostrada. De seguida, com o método findViewById, afectam-se as variáveis de instância correspondentes às Views do layout, ou seja, um ImageView e três TextView. Após serem iniciadas as variáveis, são afectadas com os métodos, **setImageDrawable** e **setText** com os valores extraídos do objecto da classe DetailsModel.