

```
1 package edu.seaaddicts.brockbutler.animation;
2
3 import android.view.View;
4 import android.view.animation.Animation;
5 import android.view.animation.Transformation;
6 import android.widget.LinearLayout.LayoutParams;
7
8 public class ExpandAnimation extends Animation {
9     public static final int ANIMATE_LONG = 500;
10    public static final int ANIMATE_SHORT = 250;
11
12    private View mAnimatedView;
13    private LayoutParams mViewLayoutParams;
14    private int mMarginStart, mMarginEnd;
15    private boolean mIsVisibleAfter = false;
16    private boolean mWasEndedAlready = false;
17
18    /**
19     * Initialize the animation
20     * @param view The layout we want to animate
21     * @param duration The duration of the animation, in ms
22     */
23    public ExpandAnimation(View view, int duration) {
24
25        setDuration(duration);
26        mAnimatedView = view;
27        mViewLayoutParams = (LayoutParams) view.getLayoutParams();
28
29        // decide to show or hide the view
30        mIsVisibleAfter = (view.getVisibility() == View.VISIBLE);
31
32        mMarginStart = mViewLayoutParams.bottomMargin;
33        mMarginEnd = (mMarginStart == 0 ? (0 - view.getHeight()) : 0);
34
35        view.setVisibility(View.VISIBLE);
36    }
37
38    @Override
39    protected void applyTransformation(float interpolatedTime, Transformation t) {
40        super.applyTransformation(interpolatedTime, t);
41
42        if (interpolatedTime < 1.0f) {
43
44            // Calculating the new bottom margin, and setting it
45            mViewLayoutParams.bottomMargin = mMarginStart
46                + (int) ((mMarginEnd - mMarginStart) * interpolatedTime);
47
48            // Invalidating the layout, making us seeing the changes we made
49            mAnimatedView.requestLayout();
50
51            // Making sure we didn't run the ending before (it happens!)
52        } else if (!mWasEndedAlready) {
53            mViewLayoutParams.bottomMargin = mMarginEnd;
54            mAnimatedView.requestLayout();
55
56            if (mIsVisibleAfter) {
57                mAnimatedView.setVisibility(View.GONE);
58            }
59            mWasEndedAlready = true;
60        }
61    }
62 }
63
```