

ViewPager

Renault@lrde.epita.fr



View Pager



An horizontal swipe change the main view



Used for slideshows, or UI setup wizards



Also used for homescreen



Alternative to tabbed application when only few tabs are required



Based on fragments



Each view is a dedicated fragment



The parent view will display fragments and animations



You don't need to define your own animations

Define the Fragment (1/2)



Define the class that inherits from Fragment

```
<ScrollView
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/content"
    android:layout_width="match_parent"
    android:layout_height="match_parent" >

<TextView style="?android:textAppearanceMedium"
    android:padding="16dp"
    android:lineSpacingMultiplier="1.2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/lorem_ipsum" />
</ScrollView>
```

Define the Fragment (2/2)



Define the core of your fragment

```
public class ScreenSlidePageFragment extends Fragment {  
  
    @Override  
    public View onCreateView(LayoutInflater inflater,  
                            ViewGroup container,  
                            Bundle savedInstanceState) {  
        ViewGroup rootView = (ViewGroup) inflater.  
            inflate(R.layout.some_page, container, false);  
        return rootView;  
    }  
}
```



Setup the main activity



Use a ViewPager to display your pages

```
<android.support.v4.view.ViewPager
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/pager"
    android:layout_width="match_parent"
    android:layout_height="match_parent" />
```



You now have to :

- 📱 Create a pager adapter extending FragmentStatePagerAdapter
- 📱 setup the view pager with your own adapter

Defining a PagerAdapter



A simple adapter with 5 screens

Using different fragment can be done in getItem

```
private class ScreenSlidePagerAdapter
    extends FragmentStatePagerAdapter {

    public ScreenSlidePagerAdapter(FragmentManager fm) {
        super(fm);
    }

    @Override
    public Fragment getItem(int position) {
        return new ScreenSlidePageFragment();
    }

    @Override
    public int getCount() {
        return 5;
    }
}
```

Setup altogether

```
public class ScreenSlidePagerActivity extends FragmentActivity
{
    private ViewPager mPager;
    private PagerAdapter mPagerAdapter;

    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_screen_slide);

        mPager = (ViewPager) findViewById(R.id.pager);
        mPagerAdapter = new ScreenSlidePagerAdapter
            (getSupportFragmentManager());
        mPager.setAdapter(mPagerAdapter);
    }

    public void onBackPressed() {
        if (mPager.getCurrentItem() == 0)
            super.onBackPressed();
        else
            mPager.setCurrentItem(mPager.getCurrentItem() - 1);
    }
}
```

Animating transitions



Animations can be specified by implementing ViewPager.PageTransformer

- This method is called once for each visible page
- The previous page is also considered as visible for the animation
 - ▶ **Moving from page 2 to 3 will imply page 1, page 2 and page**
- transformPage(): is doing the job



Some predefined animations exits

- The page grows while getting close to the center



```
ViewPager mPager = (ViewPager) findViewById(R.id.pager);  
mPager.setPageTransformer(true, new ZoomOutPageTransformer());
```


Building your transitions



Implementation of ViewPager.PageTransformer is easy!

```
public class ParallaxPageTransformer
    implements ViewPager.PageTransformer {

    public void transformPage(View view, float position) {
        int pageWidth = view.getWidth();
        if (position < -1) { // [-Infinity,-1)
            // This page is way off-screen to the left.
            view.setAlpha(1);
        } else if (position <= 1) { // [-1,1]
            //Half the normal speed
            view.setTranslationX(-position * (pageWidth / 2));
        } else { // (1,+Infinity]
            // This page is way off-screen to the right.
            view.setAlpha(1);
        }
    }
}
```

Summary



ViewPager

- reduce the amount of displayed elements
- without removing actions



Easily customizable

- Specification of animations is simple
 - ▶ **use predefined ones**
 - ▶ **use your own**



Alternative to tabbed applications

- useful for content displaying
 - ▶ **Video**
 - ▶ **News**
 - ▶ **...**



