

# Mobile Application Development

## MyRent ViewPager

Waterford Institute of Technology

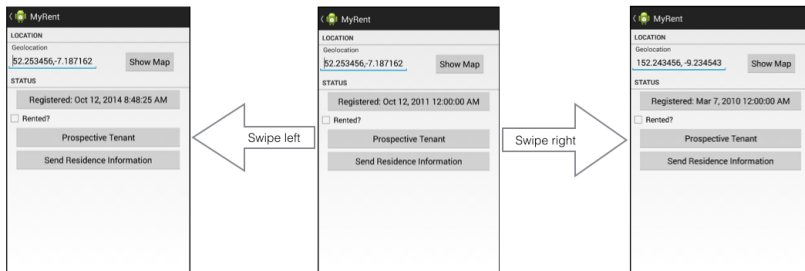
October 10, 2016

John Fitzgerald

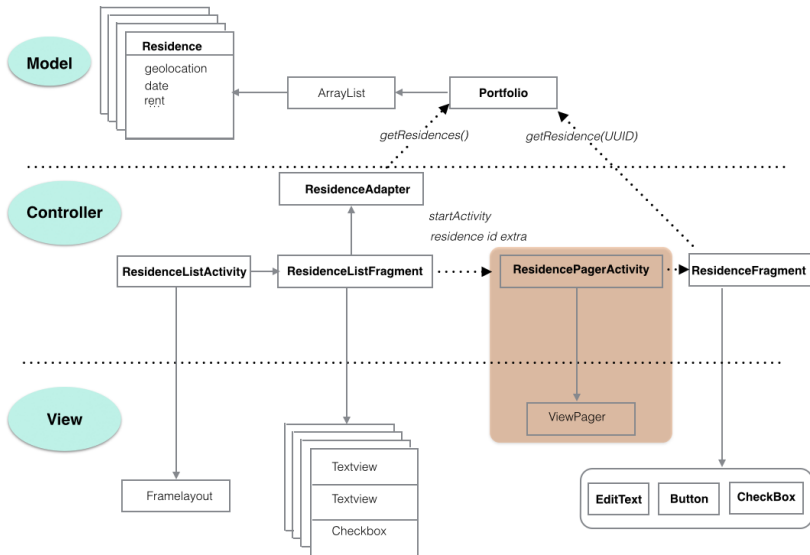
# ViewPager

## Layout Manager

- User allowed swipe views or pages left and right
- Convenient to use with fragments



# ViewPager



# ResidencePagerActivity

## Modify Manifest

- ResidencePagerActivity replaces ResidenceActivity
- Rename not mandatory - simply for self-documenting reason
- Nominate parent activity in manifest

```
<activity
  android:name=".activities.ResidencePagerActivity"
  android:label="@string/app_name" >
  <meta-data android:name="android.support.PARENT_ACTIVITY"
    android:value=".activities.ResidenceListActivity"/>
</activity>
```

# ResidencePagerActivity

## Class skeleton

### Android ViewPager

- Layout manager allowing left and right swipe through data

```
public class ResidencePagerActivity extends AppCompatActivity {  
    private ViewPager viewPager;  
    @Override  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
    }  
}
```

# ResidencePagerActivity

ViewPager object becomes activity content view

```
public class ResidencePagerActivity extends AppCompatActivity {  
    private ViewPager viewPager;  
    @Override  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
  
        viewPager = new ViewPager(this);  
        viewPager.setId(R.id.viewPager);  
        setContentView(viewPager);  
    }  
}
```

# ResidencePagerActivity

Resource id required for ViewPager

```
<resources>  
  <item type="id" name="viewPager" />  
</resources>
```

# ResidencePagerActivity

Initialize local reference to residence list

```
private void setResidenceList() {  
    MyRentApp app = (MyRentApp) getApplication();  
    portfolio = app.portfolio;  
    residences = portfolio.residences;  
}
```



# Abstract class

A brief digression

## Abstract class

- Optionally may include abstract methods.
- Like interfaces, cannot instantiate.
- Subclasses must implement abstract methods.

# Abstract class

A brief digression

```
abstract class FragmentStatePagerAdapter {  
    int pos;  
    Fragment fragment;  
  
    public void onCreate() {  
        // TODO Complete implementation  
    }  
  
    abstract int getCount();  
    abstract Fragment getItem(int pos);  
}
```

# Abstract class

## A brief digression

```
public class PagerAdapter extends FragmentStatePagerAdapter {  
  
    @Override  
    int getCount() {  
        // TODO Auto-generated method stub  
        return 0;  
    }  
  
    @Override  
    Fragment getItem(int pos) {  
        // TODO Auto-generated method stub  
        return null;  
    }  
}
```

# PagerAdapter

## FragmentManagerPagerAdapter

```
class PagerAdapter extends FragmentPagerAdapter {
    ...
    @Override
    public int getCount() {
        return residences.size();
    }
    @Override
    public Fragment getItem(int pos) {
        Residence res = residences.get(pos);
        Bundle args = new Bundle();
        args.putSerializable(ResidenceFragment.EXTRA_RESIDENCE_ID, res.
            id);
        ResidenceFragment fragment = new ResidenceFragment();
        fragment.setArguments(args);
        return fragment;
    }
}
```

# PagerAdapter

## FragmentStatePagerAdapter

- Uses a Fragment to manage each page
- Manages save & restore fragment state
- Frugal with memory usage

# ResidencePagerActivity

Ensure correct residence displayed

```
void setCurrentItem() {  
    Long resId = (Long) getIntent()  
        .getSerializableExtra(ResidenceFragment.  
        EXTRA_RESIDENCE_ID);  
    for (int i = 0; i < residences.size(); i++) {  
        if (residences.get(i).id.equals(resId)) {  
            viewPager.setCurrentItem(i);  
            break;  
        }  
    }  
}
```

# ResidencePagerActivity

## onCreate

```
@Override
public void onCreate(Bundle savedInstanceState)
{
    super.onCreate(savedInstanceState);
    viewPager = new ViewPager(this);
    viewPager.setId(R.id.viewPager);
    setContentView(viewPager);
    setResidenceList();
    pagerAdapter = new PagerAdapter(getSupportFragmentManager(),
        residences);
    viewPager.setAdapter(pagerAdapter);
    setCurrentItem();
}
```

# ResidencePagerActivity

## Listener interface

Optional feature used here to change title in action bar

```
ViewPager.OnPageChangeListener
```

```
@Override
```

```
public void onPageScrolled(int position, float positionOffset, int  
positionOffsetPixels)
```

```
@Override
```

```
public void onPageSelected(int position)
```

```
@Override
```

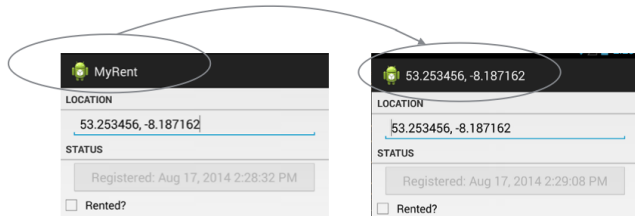
```
public void onPageScrollStateChanged(int state)
```



# ResidencePagerActivity

## Listener interface

```
public void onPageScrolled(int position, float positionOffset, int positionOffsetPixels) {  
  
    Residence residence = residences.get(position);  
    if (residence.geolocation != null) {  
        setTitle(residence.geolocation);  
    }  
}
```



# ViewPager.OnPageChangeListener

Explained using demo class

```
public class ViewPager {  
  
    // Nested interface containing single abstract method  
    interface OnPageChangeListener {  
        void onPageSelected(int pos);  
    }  
}
```

# ViewPager.OnPageChangeListener

Explained using demo class

```
public class ResidencePagerActivity implements ViewPager.  
    OnPageChangeListener  
{  
  
    // Must implement OnPageChangeListener interface method  
    @Override  
    public void onPageSelected(int pos) {  
        // TODO Auto-generated method stub  
    }  
  
}
```

## Referenced Material

1. Android Documentation: ViewPager

<http://developer.android.com/reference/android/support/v4/view/ViewPager.html>

[Accessed 2016-10-10]

2. Android Programming The Big Nerd Ranch Guide. Philips B. & Hardy B. Chapter 11. 2013

3. Oracle: Abstract Methods and Classes

<https://docs.oracle.com/javase/tutorial/java/IandI/abstract.html>

[Accessed 2016-10-10]



Except where otherwise noted, this content is licensed under a Creative Commons Attribution-NonCommercial 3.0 License.

For more information, please see <http://creativecommons.org/licenses/by-nc/3.0/>



Waterford Institute of Technology  
INSTITIÚD TEICNEOLAÍOCHTA PHORT LÁIRGE

