Mobile Application Development

Produced by

Eamonn de Leastar (edeleastar@wit.ie)

Department of Computing, Maths & Physics
Waterford Institute of Technology

http://www.wit.ie

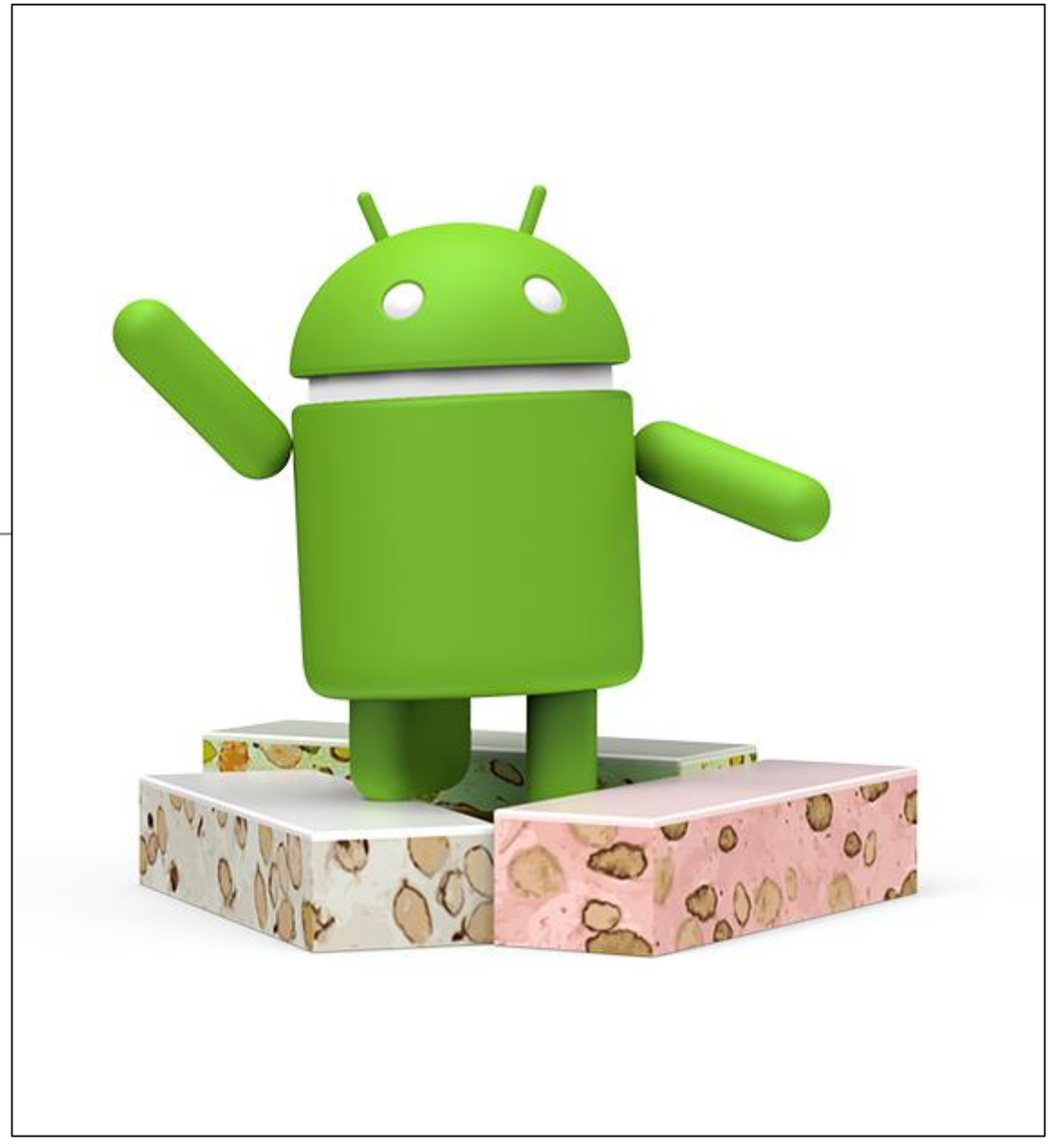
http://elearning.wit.ie

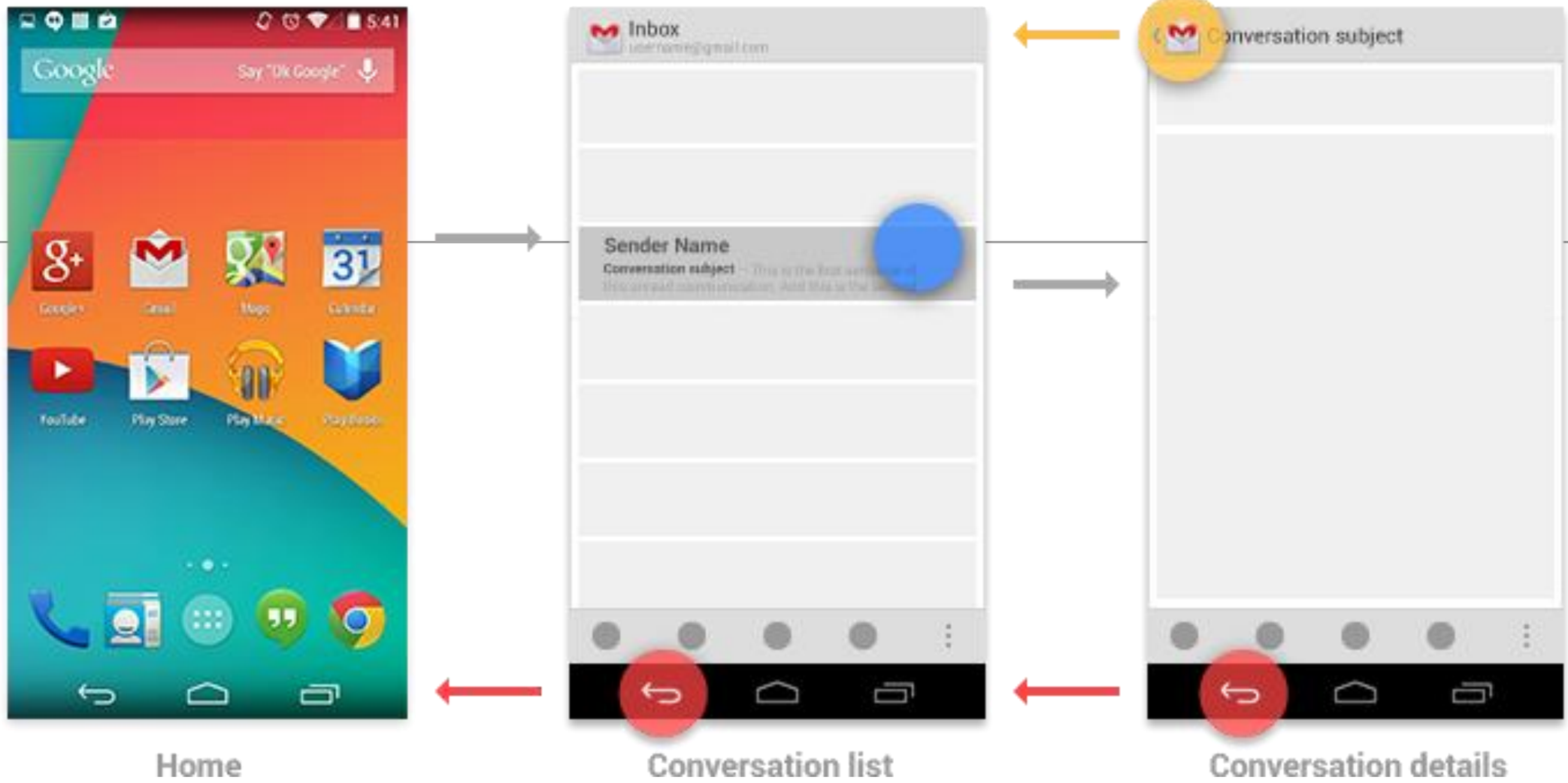
Waterford Institute *of* Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

eLearning support unit

Up button Support

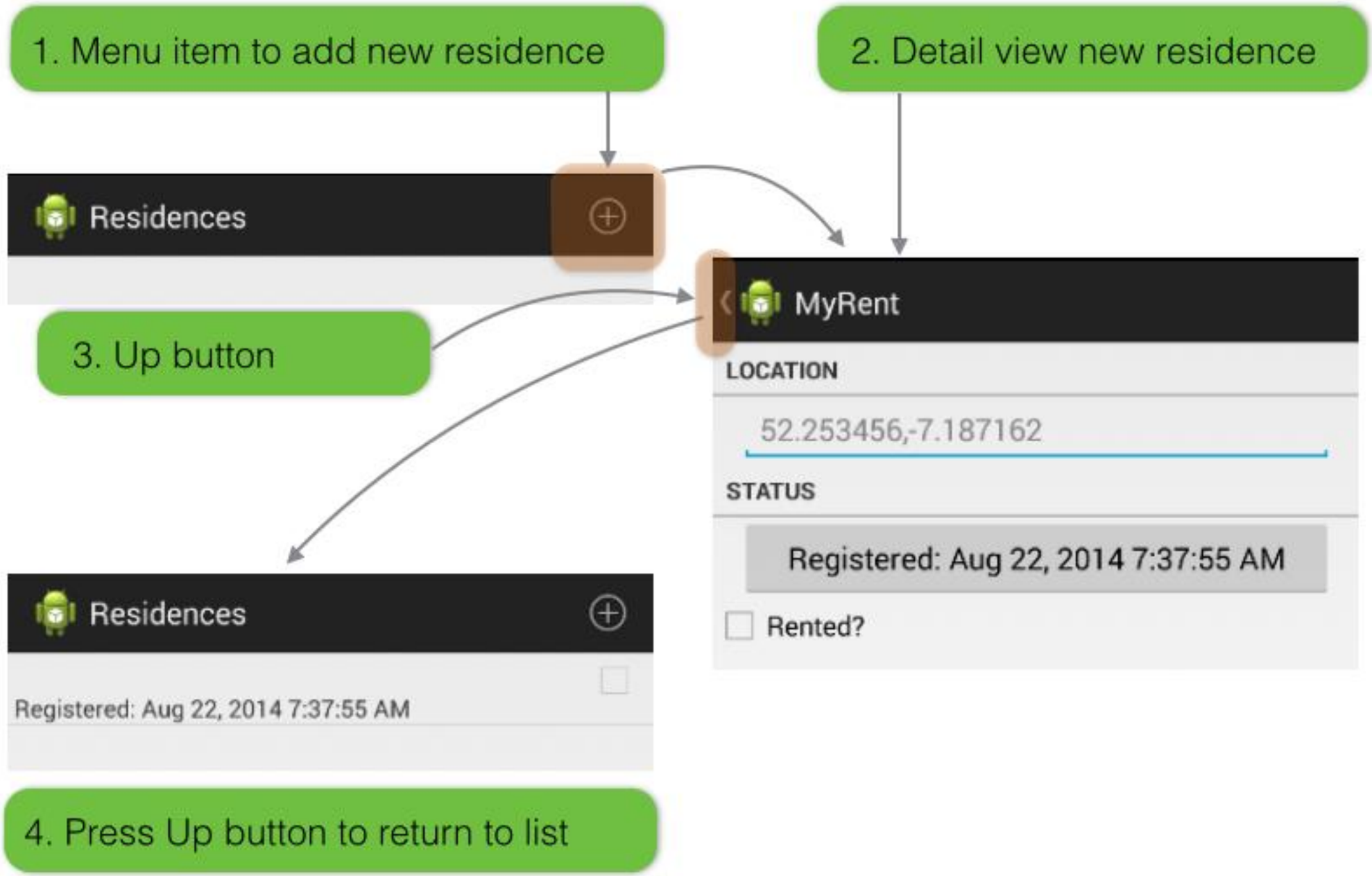
Manual coding required!





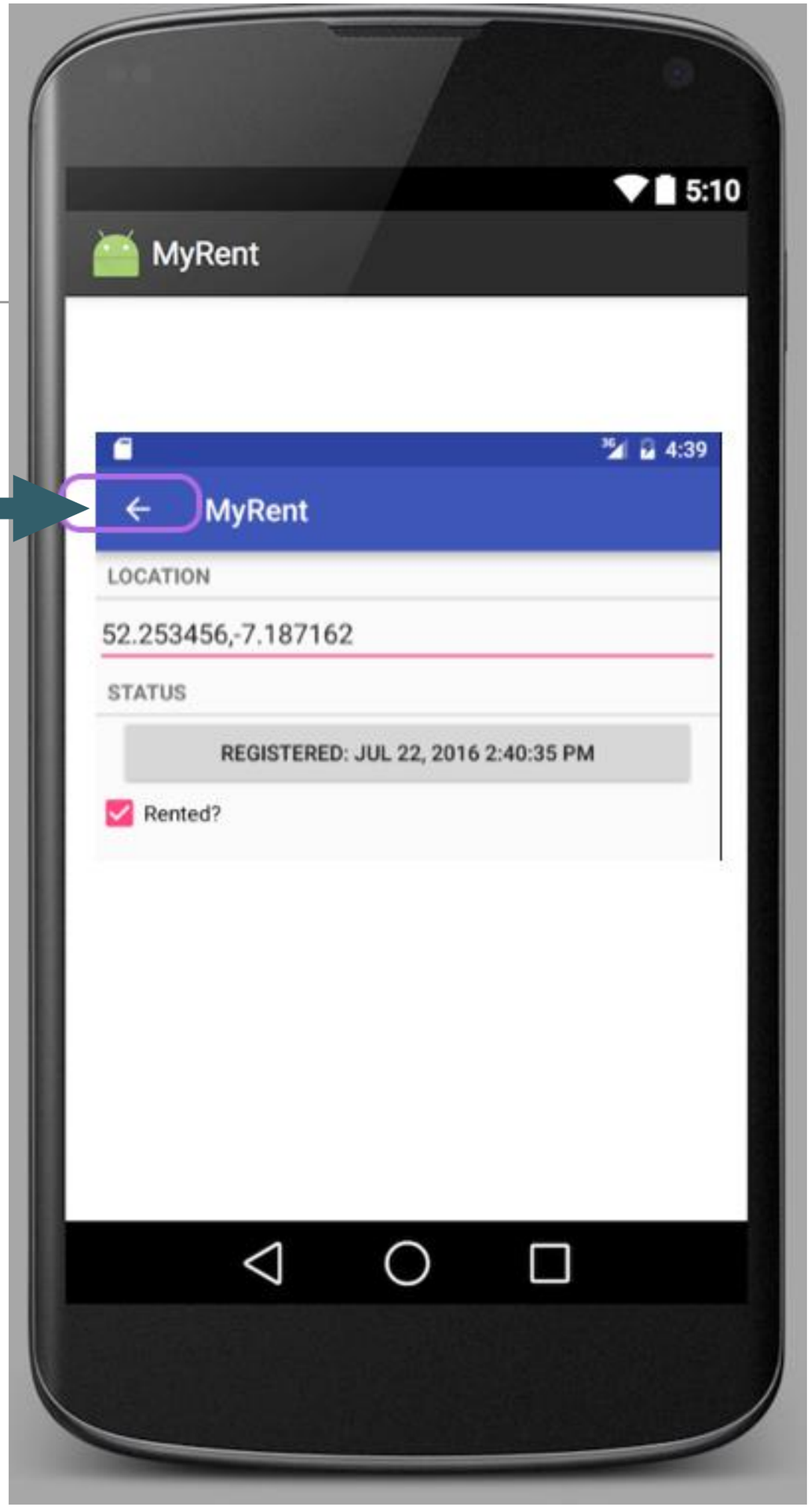
- When the previously viewed screen is also the hierarchical parent of the current screen, pressing the Back button has the same result as pressing an Up button—this is a common occurrence.

- However, unlike the Up button, which ensures the user remains within your app, the Back button can return the user to the Home screen, or even to a different app.

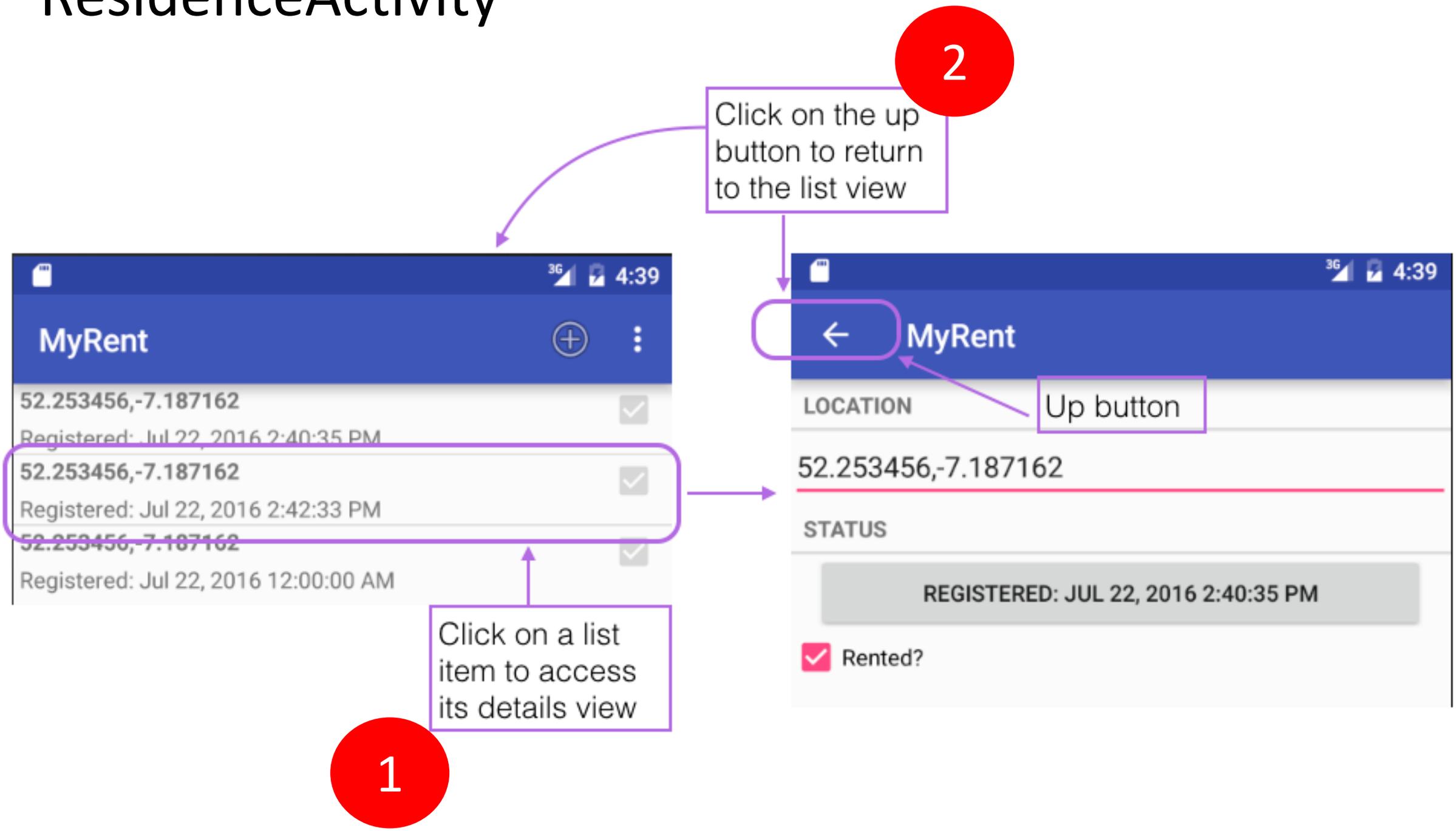

1. Menu item to add new residence
2. Detail view new residence
Residences
MyRent
3. Up button
LOCATION
52.253456,-7.187162
STATUS
Registered: Aug 22, 2014 7:37:55 AM
Rented?
Residences
Registered: Aug 22, 2014 7:37:55 AM
4. Press Up button to return to list

ResidenceActivity

- “Up” support not available by default
- Needs to be explicitly programmed



ResidenceActivity



Enabling “Up” button support

1. Write a method that will handle the “up” navigation

2. Enable the “Up” button in the Action Bar

3. Override the onOptionsItemSelected method to invoke the handler method (written in step 1 above) when the “Up” button is clicked.

4. Declare the “parent” activity in the Android Manifest file.

1. Write a method that will handle the “up” navigation

- Introduce helper to encapsulate up navigation

```
public static void navigateUp(Activity parent)
{
  Intent upIntent = NavUtils.getParentActivityIntent(parent);
  NavUtils.navigateUpTo(parent, upIntent);
}
```

NavUtils

public final class NavUtils

extends Object

java.lang.Object

↳ android.support.v4.app.NavUtils

NavUtils provides helper functionality for applications implementing recommended Android UI navigation patterns. For information about recommended navigation patterns see Tasks and Back Stack from the developer guide and Navigation from the design guide.

Public methods	
static Intent	getParentActivityIntent(Context context, ComponentName componentName) Obtain an Intent that will launch an explicit target activity specified by sourceActivityClass's PARENT_ACTIVITY <meta-data> element in the application's manifest.
static Intent	getParentActivityIntent(Context context, Class<?> sourceActivityClass) Obtain an Intent that will launch an explicit target activity specified by sourceActivityClass's PARENT_ACTIVITY <meta-data> element in the application's manifest.
static Intent	getParentActivityIntent(Activity sourceActivity) Obtain an Intent that will launch an explicit target activity specified by sourceActivity's PARENT_ACTIVITY <meta-data> element in the application's manifest.
static void	navigateUpFromSameTask(Activity sourceActivity) Convenience method that is equivalent to calling navigateUpTo(sourceActivity, getParentActivityIntent (sourceActivity)).
static void	navigateUpTo(Activity sourceActivity, Intent upIntent) Navigate from sourceActivity to the activity specified by upIntent, finishing sourceActivity in the process.

2. Enable the “Up” button in the Action Bar

- Enable Up Button in Action Bar

ResidenceActivity

```
@Override
public void onCreate(Bundle savedInstanceState)
{
  super.onCreate(savedInstanceState);
  setContentView(R.layout.activity_residence);
  getSupportActionBar().setDisplayHomeAsUpEnabled(true);
  ...
}
```

3. Override the onOptionsItemSelected method

ResidenceActivity

```
@Override
public boolean onOptionsItemSelected(MenuItem item)
{
  switch (item.getItemId())
  {
    case android.R.id.home:  navigateUp(this);
                             return true;
  }
  return super.onOptionsItemSelected(item);
}
```

http://developer.android.com/reference/android/app/Activity.html#onOptionsItemSelected(android.view.MenuItem)

---

public boolean **onOptionsItemSelected** (MenuItem item) Added in API level 1

This hook is called whenever an item in your options menu is selected. The default implementation simply returns false to have the normal processing happen (calling the item's Runnable or sending a message to its Handler as appropriate). You can use this method for any items for which you would like to do processing without those other facilities.

Derived classes should call through to the base class for it to perform the default menu handling.

Parameters

item The menu item that was selected.

Returns

boolean Return false to allow normal menu processing to proceed, true to consume it here.

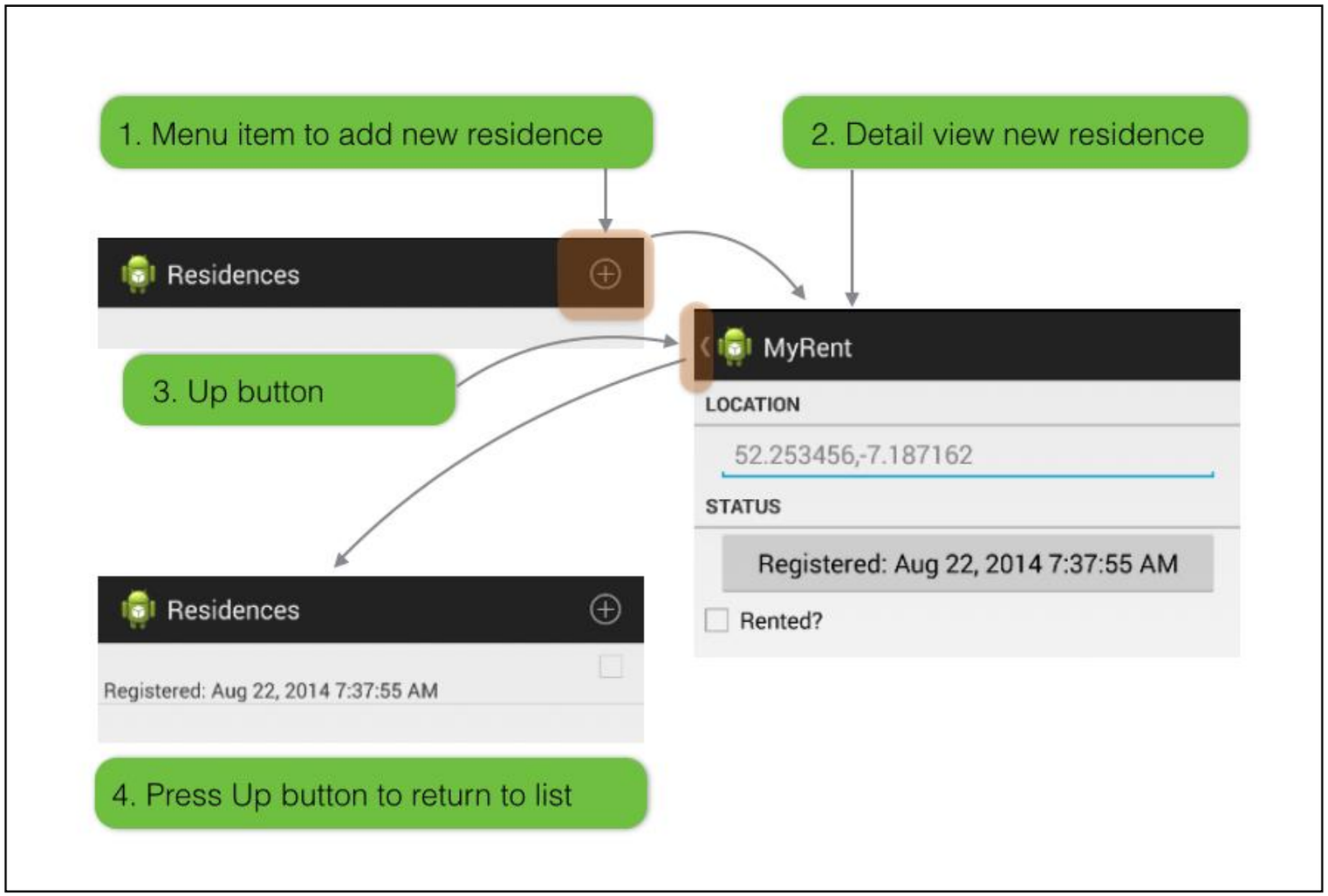
See Also

`onCreateOptionsMenu(Menu)`

4. Declare the “parent” activity in the Android Manifest file

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

Declare ResidenceListActivity as the ‘parent’ of ResidenceActivity in the manifest.



Questions?