More UI

Tabs, Fragments, ListViews

Tabs

- Create a Fragment for each tab
- Get ActionBar.Tab references for each tab
- Implement ActionBar.TabListener interface
  - onTabSelected method
- Add the appropriate Fragment to the appropriate tab

Fragments

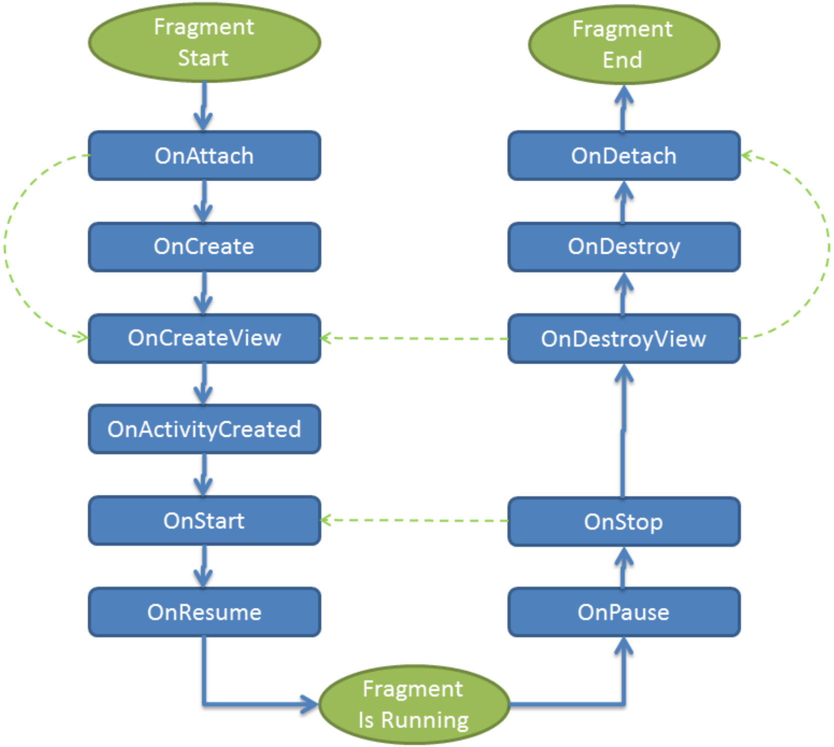
- Fragment: a “mini-activity” that can be attached to an activity.
  - Statically: put Fragment elements in your Activity’s layout file
  - Programmatically:
    - Get a FragmentManager
    - Start a transaction
    - add (or replace) a Fragment
    - commit changes

Fragment Lifecycle

OnAttach()
Called after the Fragment is associated with the Activity. This is the first method to be run when the Fragment is ready to be used.

In general, Fragments should not implement a constructor or override the default constructor. Any components that are required for the Fragment should be initialized in this method.

from: http://developer.xamarin.com/guides/android/platform_features/fragments/part_1_-_creating_a_fragment/



Fragment Comes Alive!

OnCreate()
Called by the Activity to create the Fragment.

When this method is called, the view hierarchy of the hosting Activity may not be completely instantiated, so the Fragment should not rely on any parts of the Activity's view hierarchy until later on in the Fragment's lifecycle. For example, do not use this method to perform any tweaks or adjustments to the UI of the application.

This is the earliest time at which the Fragment may begin gathering the data that it needs. The Fragment is running in the UI thread at this point, so avoid any lengthy processing, or perform that processing on a background thread.

This method may be skipped if SetRetainInstance(true) is called. This alternative will be described in more detail below.

OnCreateView()

Creates the view for the Fragment. This method is called once the Activity’s OnCreate() method is complete. At this point, it is safe to interact with the view hierarchy of the Activity. This method should return the view that will be used by the Fragment.

OnActivityCreated()

Called after Activity.OnCreate has been completed by the hosting Activity. Final tweaks to the user interface should be performed at this time.

OnStart()

Called after the containing Activity has been resumed. This makes the Fragment visible to the user. In many cases, the Fragment will contain code that would otherwise be in the OnStart() method of an Activity.

OnResume()

This is the last method called before the user can interact with the Fragment. An example of the kind of code that should be performed in this method would be enabling features of a device that the user may interact with, such as the camera that the location services. Services such as these can cause excessive battery drain, though, and an application should minimize their use in order to preserve battery life.

The Fragment Dies

OnPause()

The user is no longer able to interact with the Fragment. This situation exists because some other Fragment operation is modifying this Fragment, or the hosting Activity is paused. It is possible that the Activity hosting this Fragment might still be visible, that is, the Activity in focus is partially transparent or does not occupy the full screen.

When this method becomes active, it’s the first indication that the user is leaving the Fragment. The Fragment should save any changes.

OnStop()

The Fragment is no longer visible. The host Activity may be stopped, or a Fragment operation is modifying it in the Activity. This callback serves the same purpose as Activity.OnStop.

OnDestroyView()

This method is called to clean up resources associated with the view. This is called when the view associated with the Fragment has been destroyed.

OnDestroy()
This method is called when the Fragment is no longer in use. It is still associated with the Activity, but the Fragment is no longer functional. This method should release any resources that are in use by the Fragment, such as a SurfaceView that might be used for a camera.

This method may be skipped if SetRetainInstance(true) is called. This alternative will be described in more detail below.

**OnDetach(**)
This method is called just before the Fragment is no longer associated with the Activity. The view hierarchy of the Fragment no longer exists, and all resources that are used by the Fragment should be released at this point.

Using SetRetainInstance

It is possible for a Fragment to specify that it should not be completely destroyed if the Activity is being re-created. The `Fragment` class provides the method`SetRetainInstance` for this purpose. If `true` is passed to this method, then when the Activity is restarted, the same instance of the Fragment will be used. If this happens, then all callback methods will be invoked except the `OnCreate` and`OnDestroy` lifecycle callbacks. This process is illustrated in the lifecycle diagram shown above (by the green dotted lines).

ListViews

- A subclass of AdapterView (a subclass of ViewGroup)
- Contains scrollable row views
- Can use a standard adapter, or a custom adapter
  - Adapters take data to put in a row, row layouts
  - Standard adapters can only work with String arrays for data