Android Applications

Dr Tadhg O’Sullivan

Application

- Android application is a collection of loosely coupled components, *i.e.*
  - *Activities*
  - *Content Providers*
  - *Intents*
  - *Services*
  - Broadcast Receivers
  - Notifications
  - Widgets

Application Lifecycle

- Applications have limited control over their runtime cycles
- App components must listen for changes in the app state and react accordingly
- Android aggressively manages its resources, doing whatever it takes for the device to remain responsive, *i.e.* killing processes (sometimes even without a warning)

Application Priorities

- Application Priority is equal to its highest priority component
- If application depends on a service or content provider of another application, the latter will have priority no less than the former
- If two processes have an equal priority, the one that was running the longest will be killed first

Application States

Priority	State	
Critical	Active	●Foreground Activities, ●Broadcast Receivers, ●Services executing onStart(), onCreate(), onDestroy() ●Foreground Services
High	Visible	Visible, but inactive Activities
High	Started Service	Process hosting services that have been started
Low	Background Process	Activities that aren't visible and have no started Services
Low	Empty Process	Cached copies of finished applications (routinely cleaned)

Application Class

- Extend Application class to
  - Maintain application state
  - Transfer objects between components
  - Manage and maintain resources used by several application component

Application Class Cont'd

```
public class MyApplication extends Application
{
        private static MyApplication singleton;
        // Returns the application instance
        public static MyApplication getInstance()
        {
        return singleton;
        }
        @Override public final void onCreate()
         {
        super.onCreate();
        singleton = this;
        }
}
```

Application Class Cont'd

- Register new application class in AndroidManifest.xml:

```
<application
android:icon="@drawable/icon"
android:name="MyApplication">
...
</application>
```

Overriding Lifecycle Events

Method called	Lifecycle event
onCreate()	Initialize app singleton, app variables and shared resources
onTerminate()	App is terminated (no guarantee it will be called!)
onLowMemory()	Opportunity for well-behaved apps to prove themselves :-) (clear cache & remove unnecessary resources)
onConfigurationChanges()	Dealing with configuration changes of the handset/device

Questions

- Please ask in the Student Forum