

# App Widgets

MultiUni

Trần Vũ Tất Bình

# Tổng quan

Là một dạng view của ứng dụng được thiết kế để nằm trong một ứng dụng khác (thường là màn hình Home)

<http://developer.android.com/guide/topics/appwidgets/index.html>

[http://developer.android.com/guide/practices/ui\\_guidelines/widget\\_design.html](http://developer.android.com/guide/practices/ui_guidelines/widget_design.html)



# Cấu trúc cơ bản

- Cần một đối tượng [AppWidgetProviderInfo](#) khai báo về layout, kích thước, chu kỳ cập nhật... của widget (nên khai báo trong xml)
- Một lớp implement từ [AppWidgetProvider](#), xử lý các trạng thái của widget (updated, enabled, disabled, deleted) và cập nhật hiển thị của widget

# Khai báo trong Manifest

- Khai báo lớp AppWidgetProvider dưới thẻ receiver trong Manifest (vì anh này thực tế là một Receiver)

```
<receiver android:name="ExampleAppWidgetProvider" >  
    <intent-filter>  
        <action  
            android:name="android.appwidget.action.APPWIDGET_UPDATE" />  
    </intent-filter>  
  
    <meta-data android:name="android.appwidget.provider"  
        android:resource="@xml/example_appwidget_info" />  
</receiver>
```

# Khai báo AppWidgetProviderInfo

- Đặt trong thư mục res/xml:

```
<appwidget-provider
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:minWidth="294dp"
  android:minHeight="72dp"
  android:updatePeriodMillis="86400000"
  android:initialLayout="@layout/example_appwidget"
  android:configure="com.example.android.ExampleAppWidgetConfigure"
  ">
</appwidget-provider>
```

- Khai báo khung chứa widget có kích thước ra sao
- Khai báo chu kỳ cập nhật widget
- Khai báo layout của widget
- Khai báo Activity nào dùng cho user configure widget

# Lưu ý khi tạo layout cho widget

- Đọc link này:  
<http://developer.android.com/guide/topics/appwidgets/index.html#CreatingLayout>
- Các Layout được dùng: LinearLayout, FrameLayout, RelativeLayout
- Các View trong widget: [AnalogClock](#), [Button](#), [Chronometer](#), [ImageButton](#), [ImageView](#), [ProgressBar](#), [TextView](#)

# Lớp AppWidgetProvider

- Lớp này kế thừa từ BroadcastReceiver, giúp nhận các sự kiện cần thao tác với widget:
  - onUpdate: cập nhật widget
  - onDelete: khi một widget bị xóa
  - onEnabled: khi một instance của widget được tạo
  - onDisabled: khi tất cả các instance của widget bị xóa
  - onReceive: kế thừa từ BroadcastReceiver, có thể implement để xử lý các sự vừa nói hoặc để tự class phân phối lại vào các phương thức ở trên.

# Activity để điều chỉnh widget

- Người dùng có thể thông qua một activity để điều chỉnh widget theo ý muốn (dạng settings của widget)
- Có thể khai báo để khi widget được tạo thì sẽ gọi activity này:

```
<activity android:name=".ExampleAppWidgetConfigure">  
  <intent-filter>  
    <action  
      android:name="android.appwidget.action.APPWIDGET_CONFIG  
      URE" />  
  </intent-filter>  
</activity>
```



# Activity để điều chỉnh widget

- The App Widget host calls the configuration Activity and the configuration Activity should always return a result. The result should include the App Widget ID passed by the Intent that launched the Activity (saved in the Intent extras as [EXTRA\\_APPWIDGET\\_ID](#)).
- The [onUpdate\(\)](#) method **will not be called** when the App Widget is created (the system will not send the ACTION\_APPWIDGET\_UPDATE broadcast when a configuration Activity is launched). It is the responsibility of the configuration Activity to request an update from the AppWidgetManager when the App Widget is first created. However, [onUpdate\(\)](#) will be called for subsequent updates—it is only skipped the first time.