

# 課程名稱:手機平台應用開發

## 實驗五: appwidget 及 broadcast 使用

姓名:鞏持茵

學號:14331075

### 【实验目的】

- 1、掌握 AppWidget 编程基础
- 2、掌握 Broadcast 编程基础
- 3、掌握动态注册 Broadcast 和静态注册 Broadcast

### 【实验内容】

实现一个 Android 应用，实现静态广播、动态广播两种改变 widget 内容的方法。在上次实验的基础上进行修改，所以一些关于静态动态广播的内容会简略。

具体要求：

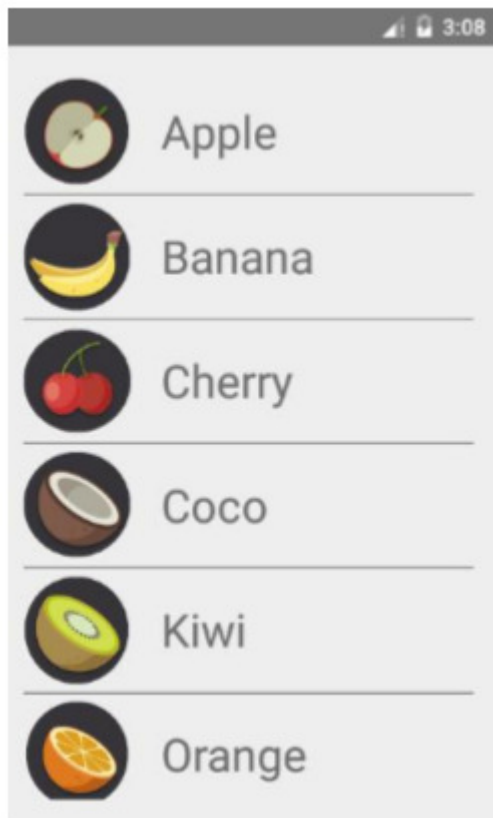
(1) 该界面为应用启动后看到的界面。



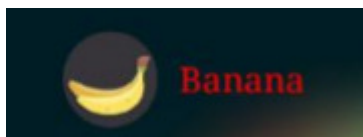
widget 初始情况如下：



( 2 ) 点击静态注册按钮，跳转至如下界面。



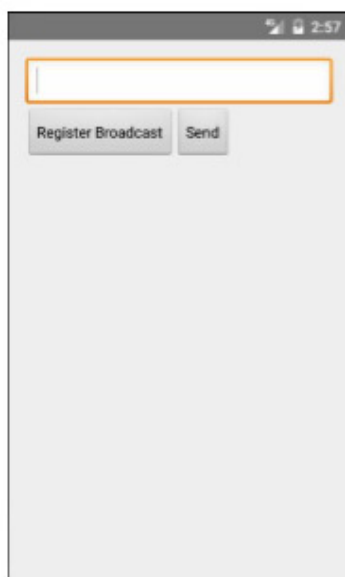
点击表单项目。如 banana。widget 会发生对应变化。点击 Widget 上的图片可以跳转回主页面



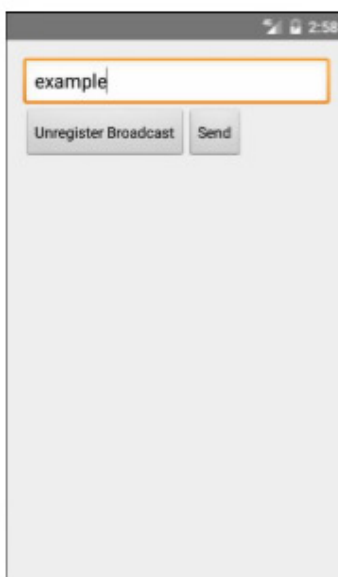
( 3 ) 点击动态注册按钮，跳转至如下界面。

实现以下功能：

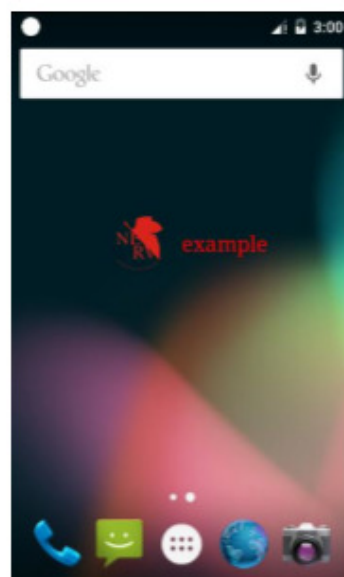
- a) 可以编辑广播的信息，点击 Send 按钮发送广播。
- b) 设置一个按钮进行广播接收器的注册与注销。
- c) 广播接收器若已被注册，发送出的广播信息能够及时更新桌面上 Widget 上文字内容及 更新为默认 dynamic 图片。
- d) 点击 Widget 上的图片可以跳转回主页面。



( 未注册广播 )



( 已注册广播 )



( 已注册后点击发送 )

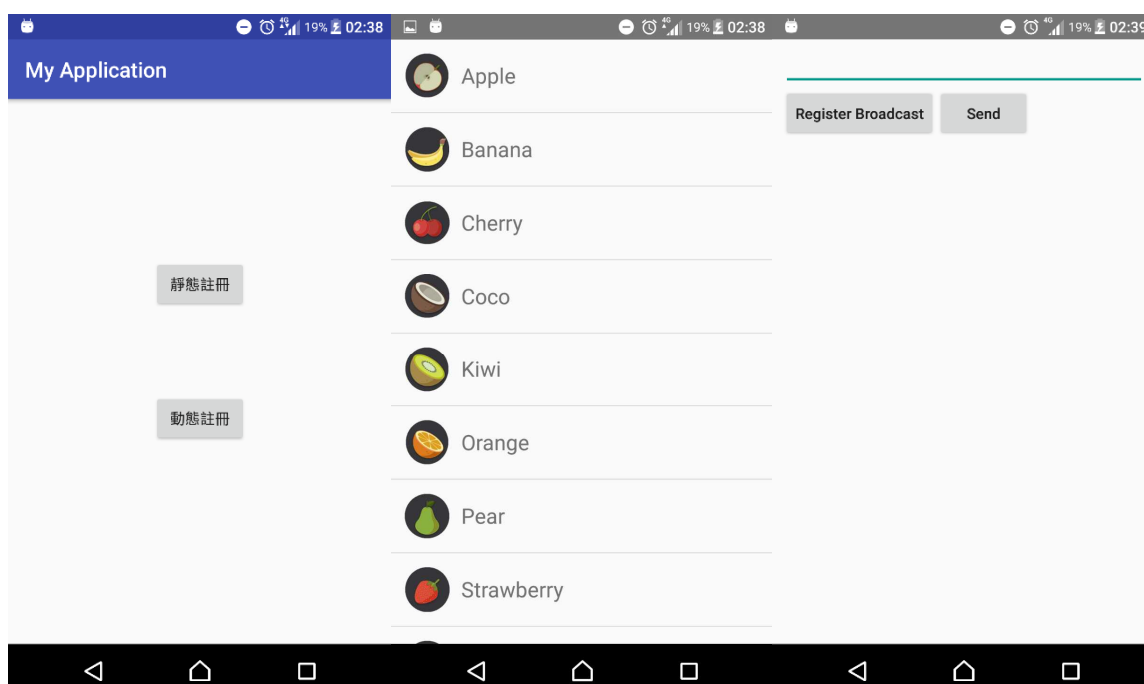
注：在设置按钮内容的时候注意大小写问题。

## 【实验步骤】

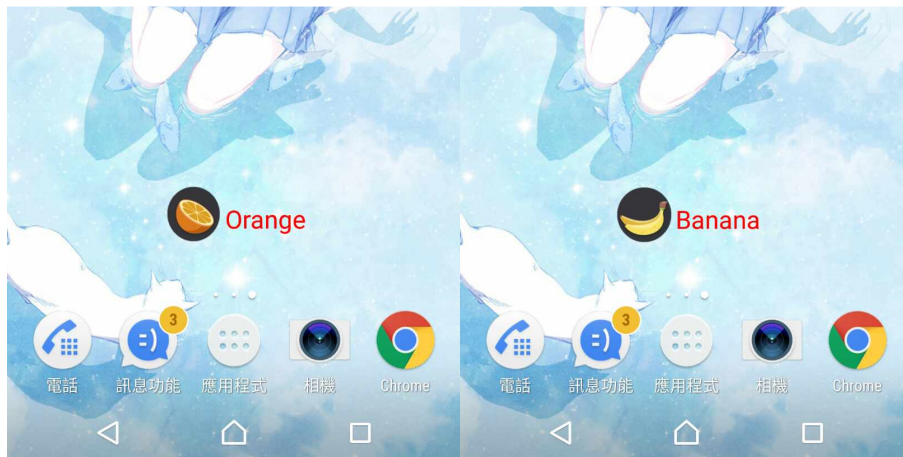
1. 接沿用上次的介面以及成果
2. 照著文档生成一个 widget 文件并注册 widget
3. 在 widget\_demo 布局文件里设好 widget 格式及透明背景
4. 在 WidgetDemo 里实现静态广播的 widget 变换更新功能
5. 在 DynamicReceiver 里实现动态注册的 widget 变换功能

## 【实验结果】

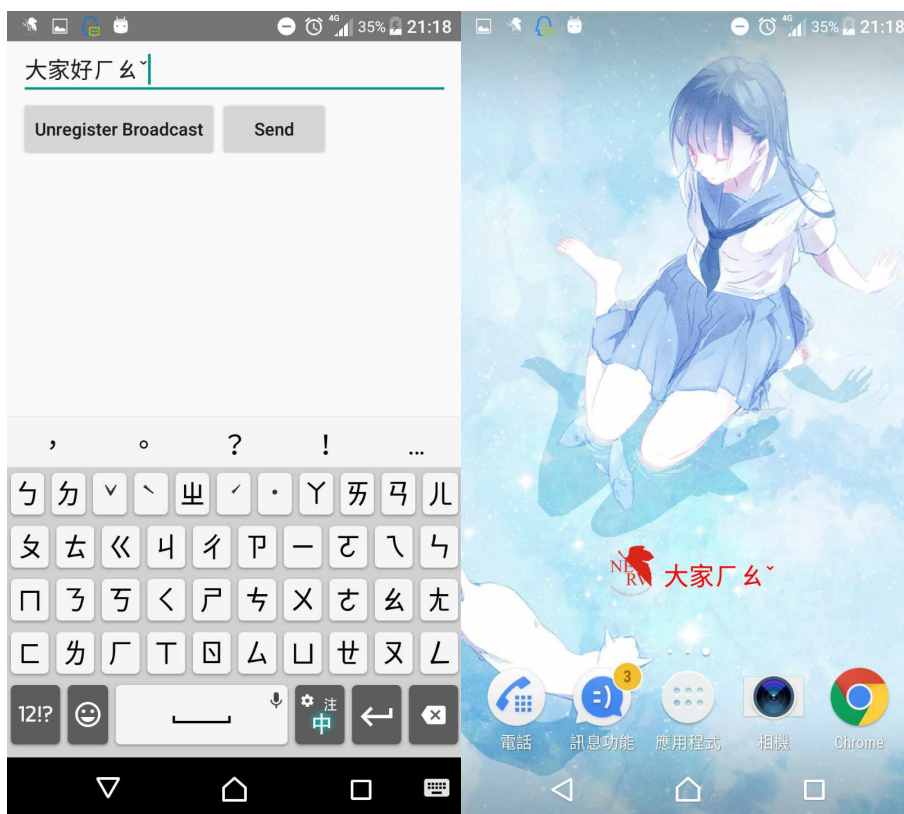
- 1) 主页点击分别跳转页面:使用 `button.setOnClickListener` 设置事件 Intent 跳转页面实现



2) 點擊水果後 widget 分別產生對應更新



3) 動態註冊，註冊信息並發送後 widget 產生對應改變



## 【实验錯誤及心得總結】

因為是從布局開始寫得所以第一個要面對的就是透明度的問題。  
在網路上找了許多資料後寫的都過於複雜，於是我就用自己的方法寫了#00000000，  
然後在 Design 的面板上也選了 imageview 的背景直接調了透明的顏色得到#00ffffff  
發現兩個都是可行的。最後得出結論關於透明度只要關注最前面兩位數即可。

```
android:background="#00ffffff" & android:background="#00000000"
```

在寫靜態廣播的 widgetDemo 時則主要是了解 setTextViewText 和  
setImageResource 這兩個函數的寫法以及在 onReceive 裡面的  
updateAppWidget 寫法。以及記得在 manifest 裡註冊，除此之外實現起來並沒有太

```
rv.setText(R.id.appwidget_text, bundle.getString("fruitname"));
rv.setImageResource(R.id.imageView, bundle.getInt("pic"));

AppWidgetManager appWidgetManagerr = AppWidgetManager.getInstance(context);
int[] widgetIDs = appWidgetManagerr.getAppWidgetIds(new ComponentName(context, WidgetDemo.class));
appWidgetManagerr.updateAppWidget(widgetIDs, rv);
```

動態廣播的 widget 則不用另外寫一個 widget。只要直接在 DynamicReceiver 裡面改動 onReceive 就好。和靜態的實現方法基本一樣，只是在最後更新 Widget 資料時的 updateAppWidget 函數的第一個參數獲取方式和靜態寫法不同。一開始沒有發現導致動態註冊的部分一直沒辦法成功更新 widget,更改之後即成功。

```
AppWidgetManager appWidgetManagerr = AppWidgetManager.getInstance(context); // 更新
int[] widgetIDs = appWidgetManagerr
    .getAppWidgetIds(new ComponentName(context, WidgetDemo.class));
appWidgetManagerr.updateAppWidget(widgetIDs, rv);
```