



实验四

broadcast 使用

【实验目的】

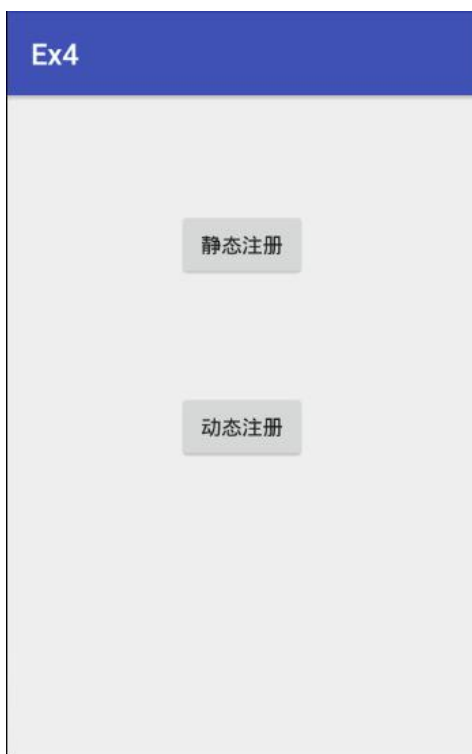
- 1、掌握 Broadcast 编程基础
- 2、掌握动态注册 Broadcast 和静态注册 Broadcast
- 3、掌握 Notification 编程基础

【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 Notification 内容的方法。

具体要求：

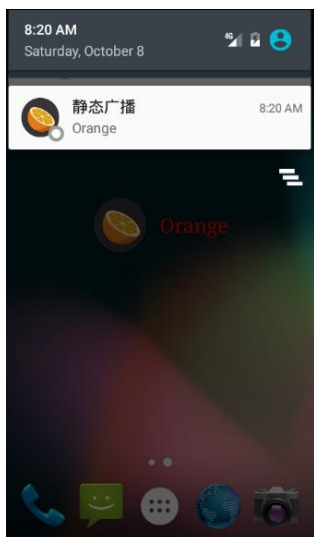
- (1) 该界面为应用启动后看到的界面。



(2) 点击静态注册按钮，跳转至如下界面。



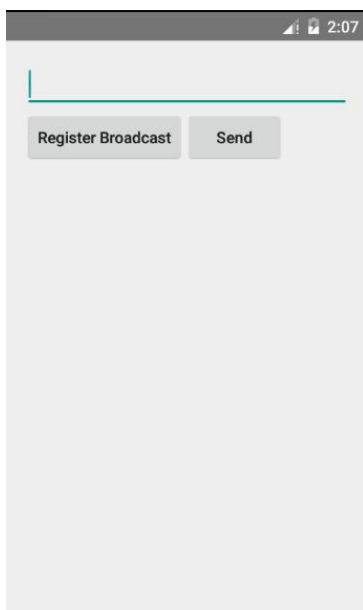
点击表单项目。如 orange。会有对应通知产生，点击通知返回主界面：



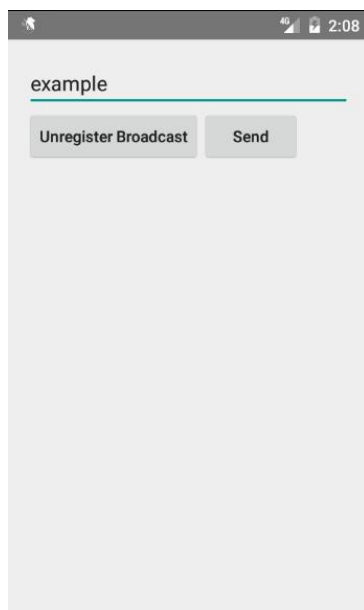
(3) 点击动态注册按钮，跳转至如下界面。

实现以下功能：

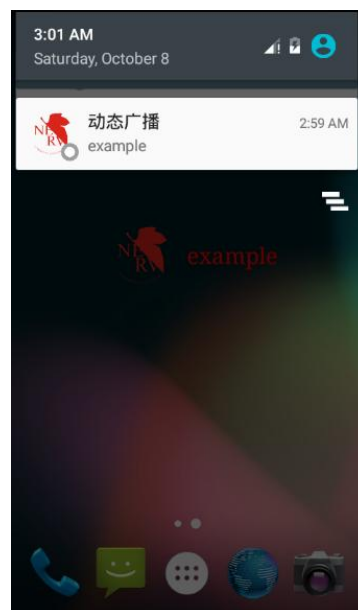
- a) 可以编辑广播的信息，点击 Send 按钮发送广播。
- b) 设置一个按钮进行广播接收器的注册与注销。
- c) 广播接收器若已被注册，发送出的广播信息会产生一个对应通知。
- d) 点击 Notification 可以跳转回主界面。



（未注册广播）



（已注册广播）

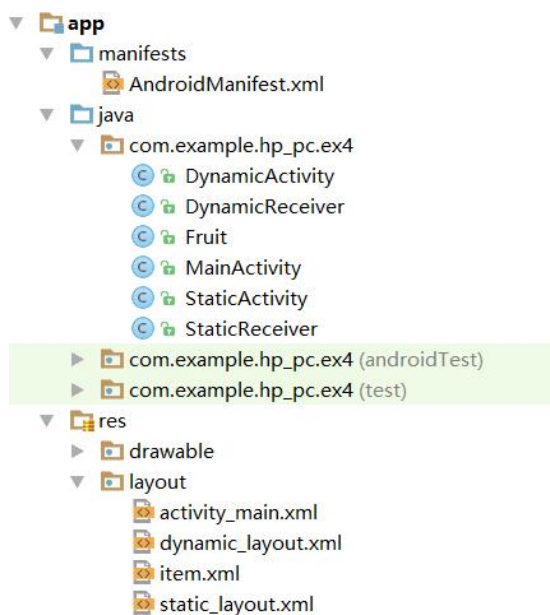


（已注册后点击发送）

注：在设置按钮内容的时候注意大小写问题。

【参考内容】

- (1) 参考文件目录如下：



(2) 静态广播部分

静态广播部分首先需要实现一个 listView 的显示，这一部分在实验三中有详细说明这里不加赘述。之后设置 `setOnClickListener`，利用 `bundle` 和 `intent` 将图片与文字内容发送出去。

```
Intent intent = new Intent(STATICACTION); // 定义Intent
intent.putExtras(bundle);
sendBroadcast(intent);
```

参考代码中的 `STATICACTION` 为自己设定的广播名称。由于是静态注册所以需要在 `AndroidMainfest.xml` 中进行注册。

```
<receiver
    android:name=".StaticReceiver"
    android:enabled="true"
    android:exported="true">
    <intent-filter>
        <action android:name="com.example.ex4.staticreceiver" />
    </intent-filter>
</receiver>
```

在静态广播类 StaticReceiver 中重写 onReceive 方法，当接收到对应广播时进行数据处理，产生通知。

```
public void onReceive(Context context, Intent intent) {  
    if (intent.getAction().equals(STATICACTION)) { //动作检测  
        Bundle bundle = intent.getExtras();  
        //TODO:添加Notification部分  
    }  
}
```

(3) 动态广播部分

1. 实现 BroadcastReceiver 子类（这里命名为 DynamicReceiver），并且重写 onReceive 方法，修改方法与静态广播类中类似。

```
public class DynamicReceiver extends BroadcastReceiver {  
    private static final String DYNAMICACTION = "com.example.ex4.dynamicreceiver";  
    @Override  
    public void onReceive(Context context, Intent intent) {  
        if (intent.getAction().equals(DYNAMICACTION)) { //动作检测  
            Bundle bundle = intent.getExtras();  
            //TODO: 添加Notification部分  
        }  
    }  
}
```

2. 创建主页面，设置注册和注销按钮(set OnClickListener)，使其能够注册或者注销广播器。

注册关键代码：

```
IntentFilter dynamic_filter = new IntentFilter();  
dynamic_filter.addAction(DYNAMICACTION); //添加动态广播的Action  
registerReceiver(dynamicReceiver, dynamic_filter); // 注册自定义动态广播消息
```

注销关键代码：

```
unregisterReceiver(dynamicReceiver);
```

其中 dynamicReceiver 为我们之前创建的 DynamicReceiver 类。用 registerReceiver 与 unregisterReceiver 分别对其进行注册与注销。

3. 设置 Send 按钮点击事件，如果广播注册，则能够点击后发送广播。发送方法与静态注册时一直，仅需修改广播名称即可。（使用 `sendBroadcast(intent)`）

4. 注意在 Android 主界面中将 `launchMode` 设置为 `singleInstance`，使得点击 Notification 后不会另外新建一个 MainActivity：

```
<activity
    android:launchMode="singleInstance"
    android:theme="@style/Theme.AppCompat.Light.NoActionBar"
    android:name=".DynamicActivity"
    android:label="DynamicActivity" />
```

(4) Notification 的使用

Notification 可以提供持久的通知，位于手机最上层的状态通知栏中。用手指按下状态栏，并从手机上方向下滑动，就可以打开状态栏查看提示消息。开发 Notification 主要涉及以下 3 个类：

1. Notification.Builder：用于动态的设置 Notification 的一些属性。

```
Notification.Builder builder = new Notification.Builder(context);
//对Builder进行配置，此处仅选取了几个
builder.setContentTitle("动态广播") //设置通知栏标题:发件人
    .setContentText(bundle.getString("name")) //设置通知栏显示内容: 短信内容
    .setTicker("您有一条新消息") //通知首次出现在通知栏，带上升动画效果的
    .setLargeIcon(bm) //设置通知大ICON
    .setSmallIcon(R.mipmap.dynamic) //设置通知小ICON(通知栏)
    .setAutoCancel(true); //设置这个标志当用户单击面板就可以让通知将自动取消
```

思考：大 ICON 如何设置？bm 是什么？

2. NotificationManager：负责将 Notification 在状态显示出来和取消；

//获取状态通知栏管理

```
NotificationManager manager = (NotificationManager) context.getSystemService(Context.NOTIFICATION_SERVICE);
```

3. Notification：设置 Notification 的相关属性。

```
//绑定Notification，发送通知请求
Notification notify = builder.build();
manager.notify(0, notify);
```

4. 点击 notification，就可以跳转到我们 intent 中指定的 activity。主要使用到 `setContentIntent` 与 `PendingIntent`。

关于 Notification，不同版本的 API 显示可能会有所不同。本次实验中必须实现的部分是标题、大图标、内容、小图标。其中标题为静态广播或动态广播；大图标与广播发送

的内容相关，为对应水果或者动态图 dynamic；内容为水果名称或动态广播发送的内容；小图标与大图标内容一样。

图片的使用方面请尽量使用 mipmap 目录下的 image asset。否则在某些 API 中可能会出现 Icon 过大的情况。

【检查内容】

- 1、布局显示是否正常。
- 2、静态广播：点击表单项是否会有对应通知产生。
- 3、动态广播：点击注册后发送内容是否会有对应通知产生。注销或未注册时，要求不会有通知产生。同时注意设置 launchMode。
- 4、点击通知是否会正确返回主界面。

【提交说明】

- 1、deadline：下一次实验课前一天晚上 12 点
- 2、提交位置：<ftp://222.200.185.18:1890/> 对应文件夹下
- 3、命名与目录要求：

附件命名及格式要求：学号_姓名_labX.zip（姓名中文拼音均可）

重复提交命名格式要求：学号_姓名_labX_Vn.zip

目录结构：

```
14331111_huashen_lab1 --  
|  
-- lab1实验报告.pdf  
|  
-- lab1_code (包含项目代码文件)
```

其中项目代码文件为项目文件夹，提交之前先 clean