

中山大学移动信息工程学院本科生实验报告

(2017 年秋季学期)

课程名称：移动应用开发

任课教师：郑贵锋

年级	2015 级	专业 (方向)	移动信息工程 (移动互联网)
学号	15352083	姓名	冯灏帆
电话	13712309295	Email	ares_fung@qq.com
开始日期	2017.10.31	完成日期	2017.11.13

一、 实验题目

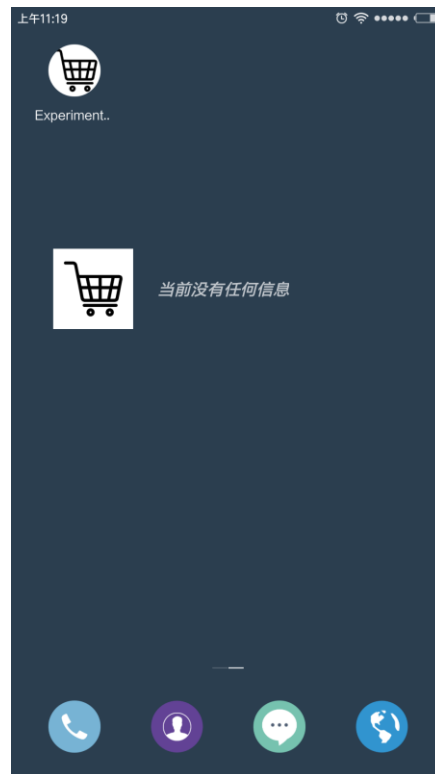
- 1、掌握 AppWidget 编程基础
- 2、掌握 Broadcast 编程基础
- 3、掌握动态注册 Broadcast 和静态注册 Broadcast

二、 实现内容

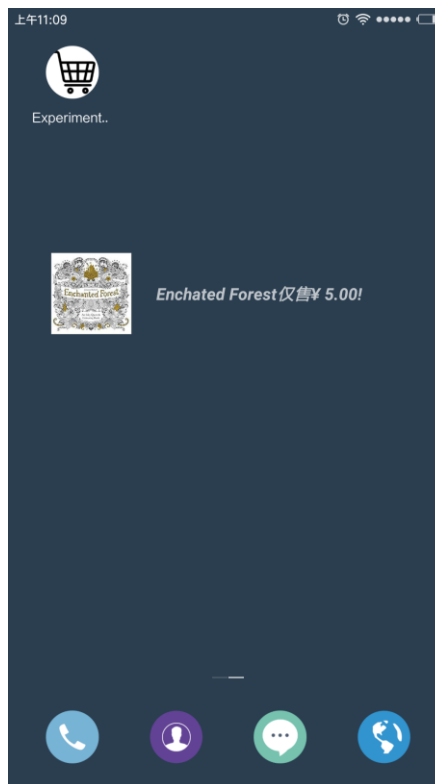
实现一个 Android 应用，实现静态广播、动态广播两种改变 widget 内容的方法。在上次实验的基础上进行修改，所以一些关于静态动态广播的内容会简略。

具体要求:

(1)widget 初始情况如下：



(2)点击 widget 可以启动应用，并在 widget 随机推荐一个商品:



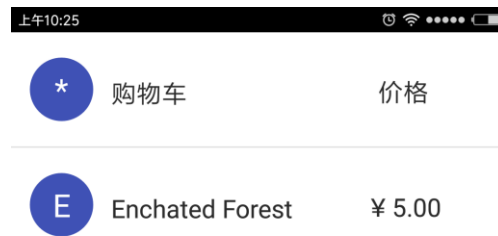
(3)点击 widget 跳转到该商品详情界面:



(4)点击购物车图标，widget 相应更新:



(5)点击 widget 跳转到购物车界面:



(6)实现方式要求:启动时的 widget 的更新通过静态广播实现, 点击购物车图标时候 widget 的更新通过动态广播实现。Widget 布局文件 m_widget.xml, 布局中有一个 ImageView, 一个 TextView。要求: 文字颜色为白色, 大小为 15sp, 整体背景为透明。

三、 课堂实验结果

(1) 实验截图



(2) 实验步骤以及关键代码

首先在 AndroidManifest.xml 中注册 widget 的静态 receiver。

```
<receiver android:name=".MyWidget">
    <intent-filter>
        <action android:name="android.appwidget.action.APPWIDGET_UPDATE" />
        <action android:name="com.lab5.MyStaticWidgetFilter" />
    </intent-filter>

    <meta-data
        android:name="android.appwidget.provider"
        android:resource="@xml/my_widget_info" />
</receiver>
```

撰写 widget 的布局文件，并设置其默认显示图片与文字。

```
<ImageView
    android:id="@+id/appwidget_image"
    android:layout_width="100dp"
    android:layout_height="match_parent"
    android:layout_alignParentLeft="true"
    android:layout_marginLeft="4dp"
    android:gravity="left|center_vertical"
    android:src="@mipmap/shoplist"
/>

<TextView
    android:id="@+id/appwidget_text"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_toRightOf="@id/appwidget_image"
    android:layout_marginLeft="4dp"
    android:layout_marginRight="0dp"
    android:gravity="left|center_vertical"
    android:background="#00000000"
    android:layout_centerVertical="true"
    android:contentDescription="Lab5"
    android:text="当前没有任何信息"
    android:textColor="#ffffff"
    android:textSize="15sp"
    android:textStyle="bold|italic" />
```

重载 MyWidget 类的成员函数 onReceiver，对于静态广播与动态广播做出不同的操作。

收到对应的静态广播时，设置推荐商品的图片与推荐文字；并创建 pendingintent 用于点击跳转至 details 中商品详细信息。

```

if(intent.getAction().equals("com.lab5.MyStaticWidgetFilter")) {
    Merchandise tempMerchandise = (Merchandise) receivedata.getSerializable( key: "Merchandise");
    Bitmap bm = BitmapFactory.decodeResource(context.getResources(), tempMerchandise.getPicname());
    remoteViews.setImageBitmap(R.id.appwidget_image, bm);
    remoteViews.setTextViewText(R.id.appwidget_text, text: tempMerchandise.getName()+"仅售*"+Double.toString(tempMerchandise.getPrice())+"!");
    Intent todetail = new Intent(context, details.class);
    Bundle senddata = new Bundle();
    senddata.putSerializable("Merchandise", tempMerchandise);
    todetail.putExtras(senddata);
    PendingIntent pi = PendingIntent.getActivity(context, (int) System.currentTimeMillis(), todetail, PendingIntent.FLAG_UPDATE_CURRENT);
    remoteViews.setOnClickPendingIntent(R.id.appwidget_image, pi);
    remoteViews.setOnClickPendingIntent(R.id.appwidget_text, pi);
    AppWidgetManager.getInstance(context).updateAppWidget(new ComponentName(context.getApplicationContext(), MyWidget.class), remoteViews);
};

```

收到对应的动态广播时，设置已添加到购物车的商品的图片与购物车文字信息；并创建 pendingintent 用于点击跳转至 mainactivity 中购物车列表。

```

if(intent.getAction().equals("com.lab5.MyDynamicWidgetFilter")) {
    Merchandise tempMerchandise = (Merchandise) receivedata.getSerializable( key: "Merchandise");
    Bitmap bm = BitmapFactory.decodeResource(context.getResources(), tempMerchandise.getPicname());
    remoteViews.setImageBitmap(R.id.appwidget_image, bm);
    remoteViews.setTextViewText(R.id.appwidget_text, text: tempMerchandise.getName()+"已添加到购物车");
    Intent toMainActivity = new Intent(context, MainActivity.class);
    Bundle senddata = new Bundle();
    senddata.putString( key: "Backto", defaultValue: "ListView");
    toMainActivity.putExtras(senddata);
    PendingIntent pi = PendingIntent.getActivity(context, (int) System.currentTimeMillis(), toMainActivity, PendingIntent.FLAG_UPDATE_CURRENT);
    remoteViews.setOnClickPendingIntent(R.id.appwidget_image, pi);
    remoteViews.setOnClickPendingIntent(R.id.appwidget_text, pi);
    AppWidgetManager.getInstance(context).updateAppWidget(new ComponentName(context.getApplicationContext(), MyWidget.class), remoteViews);
};

```

重载 MyWidget 类的成员函数 onUpdate，添加 Widget 点击事件使应用启动。

```

@Override
public void onUpdate(Context context, AppWidgetManager appWidgetManager, int[] appWidgetIds) {
    // There may be multiple widgets active, so update all of them
    //for (int appWidgetId : appWidgetIds) {
    //    updateAppWidget(context, appWidgetManager, appWidgetId);
    //}
    RemoteViews updateViews = new RemoteViews(context.getPackageName(), R.layout.my_widget);
    Intent intent = new Intent(context, MainActivity.class);
    PendingIntent pendingIntent = PendingIntent.getActivity(context, (int) System.currentTimeMillis(), intent, PendingIntent.FLAG_UPDATE_CURRENT);
    updateViews.setOnClickPendingIntent(R.id.appwidget_text, pendingIntent);
    updateViews.setOnClickPendingIntent(R.id.appwidget_image, pendingIntent);
    ComponentName me = new ComponentName(context, MyWidget.class);
    appWidgetManager.updateAppWidget(me, updateViews);
}

```

(3) 实验遇到困难以及解决思路

一开始所有静态广播都无法接受并发出 notification，但以动态广播形式发出的广播都能被监听到。后来发现是 AVD 的版本问题，Android 8.0 的静态广播存在未知的 bug，部分在 XML 文件中注册的静态广播接收器没有创建导致静态广播没有被监听到。更改为 Android 4.4 以上的其他版本以及 SDK 版本后，成功收到所有静态广播。

还有一个 Bug 解决了很久，对于商品详细页面点击购物车图标后重复发出了多个 notification 的问题，通过查看 log 发现了由于动态广播监听器的重复创建，导致了一条广播被多个广播监听器收听，造成重复响应的现象。所

以除了应用启动后第一次添加商品到购物车时发出一条 notification，后续添加其他商品到购物车时会发出多条 notification。

四、 课后实验结果

五、 实验思考及感想

为了解决本次实验遇到的困难，翻遍了 CSDN 与 Stack Overflow。对于不同情况下 activity 的跳转下，其成员函数的调用进行了深入的了解，明确了对于一个 activity 对内容更新的代码需要放在哪些重载的成员函数中。同时对于迷人的虚拟按键——返回键对 activity 的成员函数调用的影响有了一定的了解，避免了之前点击虚拟返回按键导致应用崩溃的情况。代码编写所花费的时间相对不多，更多时间花在对官方文档与开发者笔记的阅读学习上，同时多阅读成熟 app 的实现代码对于自己的学习很有帮助。