

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：郑贵峰

年级	15 级	专业 (方向)	软件工程 (移动信息工程) 互联网方向
学号	15352211	姓名	林苗
电话	13763360840	Email	554562948@qq.com
开始日期	2017.11.6	完成日期	2017.11.6

一、 实验题目

appwidget 及 broadcast 使用

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

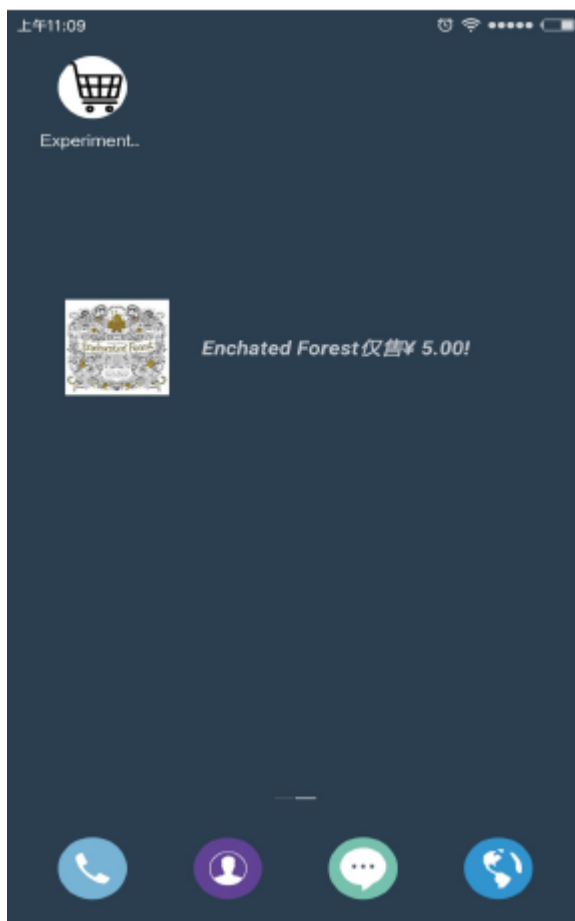
上进行修改，所以一些关于静态动态广播的内容会简略。

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：



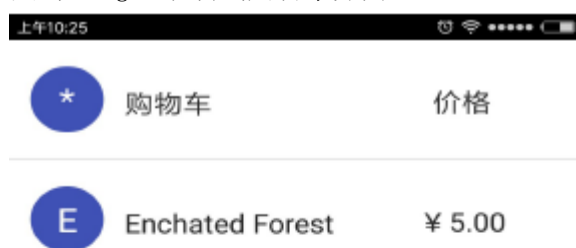
(3) 点击 widget 跳转到该商品详情界面：



(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。

三、 课堂实验结果

(1) 实验截图

初始界面:



点击 widget, 启动程序:



返回 widget，看到 widget 随机推荐一个商品：



点击 widget，跳到对应商品的详情界面：



点击购物车图标，加入购物车，查看 widget 显示：

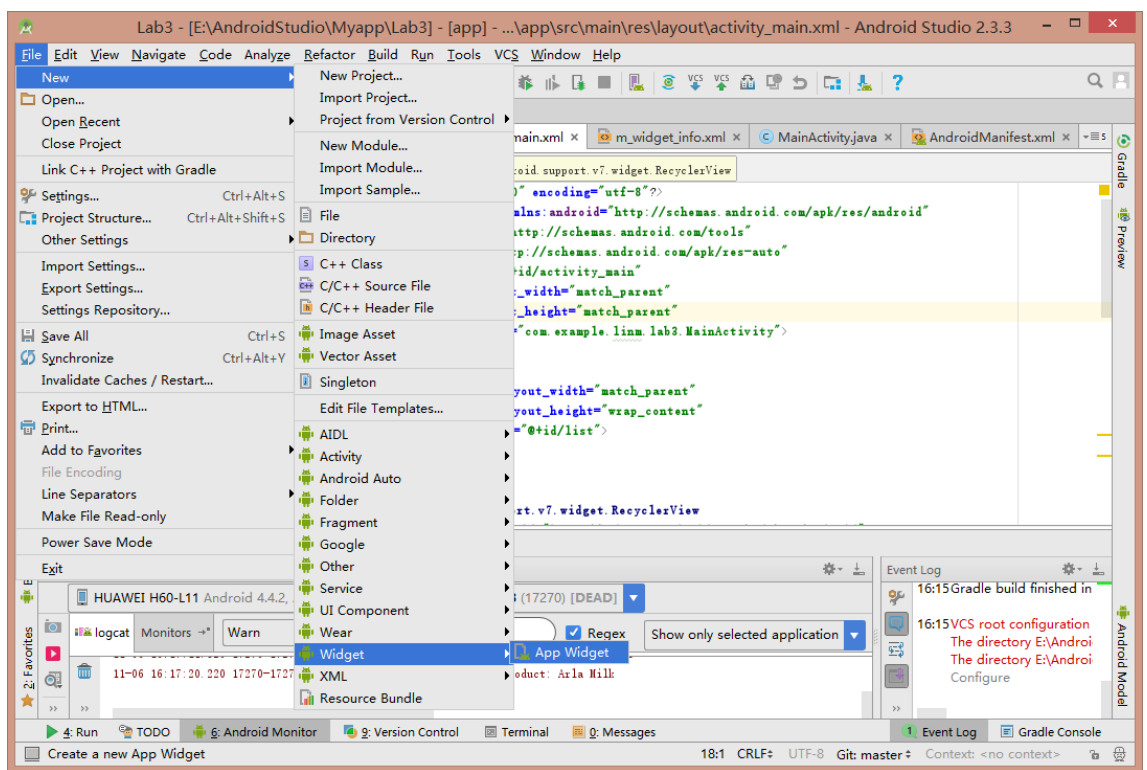


点击 widget，跳到购物车界面：

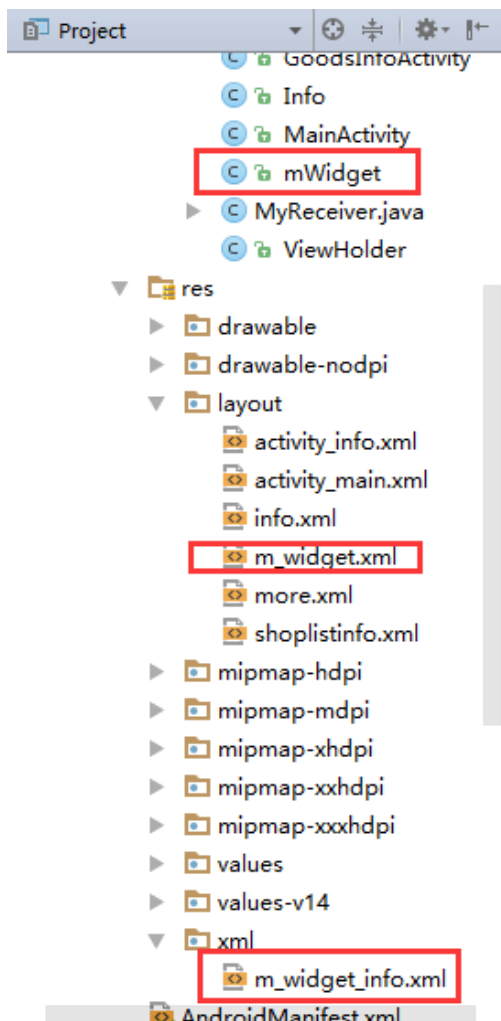


(2) 实验步骤及关键代码

新建 widget



主要用到这三个文件：



✚ 修改 m_widget_info.xml 文件

```
<?xml version="1.0" encoding="utf-8"?>
<appwidget-provider xmlns:android="http://schemas.android.com/apk/res/android"
    android:initialKeyguardLayout="@layout/m_widget"
    android:initialLayout="@layout/m_widget"
    android:minHeight="50dp"
    android:minWidth="300dp"
    android:previewImage="@mipmap/shoplist"
    android:resizeMode="horizontal|vertical"
    android:updatePeriodMillis="86400000"
    android:widgetCategory="home_screen|keyguard">

</appwidget-provider>
```

✚ 修改 m_widget.xml 文件

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="8dp"
    android:id="@+id/widget">

    <ImageView
        android:layout_width="100dp"
        android:layout_height="match_parent"
        android:src="@mipmap/shoplist"
        android:id="@+id/appwidget_image"
        android:layout_gravity="center_vertical"
    />

    <TextView
        android:id="@+id/appwidget_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="105dp"
        android:layout_centerVertical="true"

        android:contentDescription="当前没有任何消息"
        android:text="当前没有任何消息"
        android:textColor="#ffffff"
        android:textSize="15sp"
        android:textStyle="bold|italic"
        android:layout_gravity="center_vertical"
        android:gravity="center"/>

</RelativeLayout>
```

✚ 修改 mWidget.java 代码，重写 onUpdate


```

@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);
    }*/
    //实例化RemoteView, 与相应的布局对应起来
    RemoteViews updateViews = new RemoteViews(context.getPackageName(), R.layout.m_widget);
    //点击跳转到MainActivity页面
    Intent i = new Intent(context, MainActivity.class);
    PendingIntent pi = PendingIntent.getActivity(context, 0, i, PendingIntent.FLAG_UPDATE_CURRENT);
    updateViews.setOnClickPendingIntent(R.id.widget, pi);

    ComponentName me = new ComponentName(context, mWidget.class);
    //更新widget
    appWidgetManager.updateAppWidget(me, updateViews);
}

```

✚ 在 AndroidManifest.xml 文件中, 在 mWidget 下注册静态广播

```

<receiver android:name=".mWidget">
    <intent-filter>
        <action android:name="android.appwidget.action.APPWIDGET_UPDATE" />
        <action android:name="widgetStaticBroadcast" />
    </intent-filter>

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

```

✚ 在 MainActivity.java 的 onCreate 函数中
获取随机数

```

//随机数
Random random = new Random();
int random_i = random.nextInt(Infos.size());

```

发送静态广播

```

Intent wi = new Intent();
wi.setAction("widgetStaticBroadcast");
//传入随机数
wi.putExtra("info", random_i);
sendBroadcast(wi);

```

✚ 在 mWidget.java 中重写 onReceive 函数
更新文字为推荐商品的相关信息, 图片为推荐商品的图片

```

@Override
public void onReceive(Context context, Intent intent) {
    super.onReceive(context, intent);
    if (intent.getAction().equals("widgetStaticBroadcast")) {
        //得到随机数
        int i = intent.getIntExtra("info", 0);
        //实例化RemoteView
        RemoteViews updateViews = new RemoteViews(context.getPackageName(), R.layout.m_widget);
        //设置文字
        updateViews.setTextViewText(R.id.appwidget_text,
            MainActivity.Infos.get(i).getName() + "仅售" + MainActivity.Infos.get(i).getPrice() + "!");
        //设置图片
        updateViews.setImageViewResource(R.id.appwidget_image, MainActivity.Infos.get(i).getImgid());
    }
}

```

单击 widget 跳到商品详情介绍界面

```

//点击跳转
Intent newintent = new Intent(context, GoodsInfoActivity.class);
newintent.addCategory(Intent.CATEGORY_LAUNCHER);
String productname = MainActivity.Infos.get(i).getName();
Bundle bundle = new Bundle();
bundle.putString("name", productname); //属性为name, 数据为productname
newintent.putExtras(bundle);
PendingIntent pi = PendingIntent.getActivity(context, 0, newintent, PendingIntent.FLAG_UPDATE_CURRENT);
//实现单击跳转事件
updateViews.setOnClickPendingIntent(R.id.widget, pi);
//更新widget
ComponentName me = new ComponentName(context, mWidget.class);
AppWidgetManager appWidgetManager = AppWidgetManager.getInstance(context);
appWidgetManager.updateAppWidget(me, updateViews);
}
}

```

至此，静态广播部分基本实现。

商品添加到购物车时，同时相应更新 widget，因此动态广播只需在上一次的动态 DynamicReceiver 类的 onReceive 中增加以下代码

```

//实例化RemoteView
RemoteViews updateViews = new RemoteViews(context.getPackageName(), R.layout.m_widget);
//设置文字
updateViews.setTextViewText(R.id.appwidget_text, intent.getStringExtra("name") + "已经添加到购物车");
//设置图片
updateViews.setImageViewResource(R.id.appwidget_image, Imgid);
//点击跳转事件
Intent newintent = new Intent(context, MainActivity.class);
newintent.addCategory(Intent.CATEGORY_LAUNCHER);
newintent.putExtra("mode", 1);
PendingIntent pi = PendingIntent.getActivity(context, 0, newintent, PendingIntent.FLAG_UPDATE_CURRENT);
updateViews.setOnClickPendingIntent(R.id.widget, pi);
//更新widget
ComponentName me = new ComponentName(context, mWidget.class);
AppWidgetManager appWidgetManager = AppWidgetManager.getInstance(context);
appWidgetManager.updateAppWidget(me, updateViews);

```

动态广播实现完毕。

(3) 实验遇到的困难及解决思路

widget 的文字有时会叠到图片上面去。

解决思路：图片的宽度大小为 100dp，因此可以设置文字 TextView 距离右侧 105dp，这样就保证文字的开始是在图片之外了。

四、 实验思考及感想

关于 RemoteViews:

RemoteViews 是一种远程视图，在其它进程中展示，但却能够在另一个进程中更新。其构造函数的第一个参数为包名，第二个参数为布局资源 id。

```
RemoteViews updateViews = new RemoteViews(context.getPackageName(), R.layout.m_widget);
```

通过 setTextViewText、setImageViewResource 来设置 widget 中显示的文字和图片。之后可以使用 AppWidgetManager 类的 updateAppWidget 来对 widget 进行更新。

关于 addCategory(Intent.CATEGORY_LAUNCHER)

Category 属性用于指定动作 Action 被执行的环境，系统提供如下常用的 Category 属性：

- *CATEGORY_DEFAULT：默认的 Category
- *CATEGORY_BROWSABLE：能被浏览器安全调用
- *CATEGORY_LAUNCHER：该应用的入口 Activity
- *CATEGORY_HOME：随系统启动而运行
- *CATEGORY_TEST：用于测试
- *CATEGORY_OPENABLE：增加一个可打开的分类

在动态和静态广播中，设置 widget 点击事件时，都有如下代码：

```
Intent newintent = new Intent(context, MainActivity.class);  
newintent.addCategory(Intent.CATEGORY_LAUNCHER);
```

第二句表示会打开注册有 CATEGORY_LAUNCHER 的 Activity，看看 AndroidManifest.xml 文件：

```
<activity  
    android:name=".MainActivity"  
    android:launchMode="singleTask">  
    <intent-filter>  
        <action android:name="android.intent.action.MAIN" />  
  
        <category android:name="android.intent.category.LAUNCHER" />  
    </intent-filter>  
</activity>
```

MainActivity 注册了这个 category。

如果没有 addCategory 这一句，系统仍然会打开 MainActivity，但是如果 addCategory 里面的内容再 intent-filter 中找不到定义，那么将不会打开任何 Activity，且会抛出异常。

参考资料

【Android 开发教程】添加 Category

<http://blog.csdn.net/manoel/article/details/7577345>

Intent 的七大属性

<http://www.cnblogs.com/anni-qianqian/p/5302947.html>