# 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 | 15M1 | 专业（方向） | 移动互联网 |
| 学号 | 15352008 | 姓名 | 蔡荣裕 |
| 电话 | 13727021990 | Email | 897389207@qq.com |
| 开始日期 | 2017/10/24 | 完成日期 | 2017/10/30 |

## 实验题目

appwidget 及broadcast 使用

## 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。

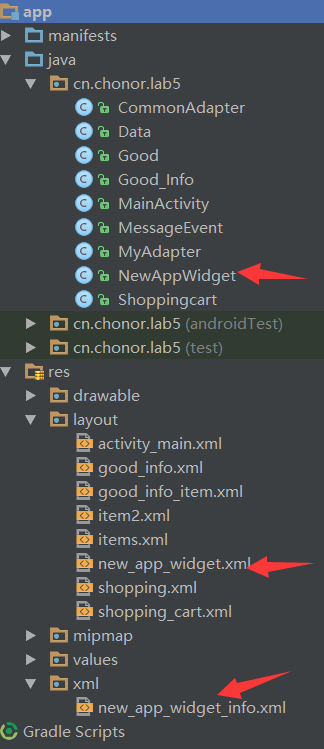
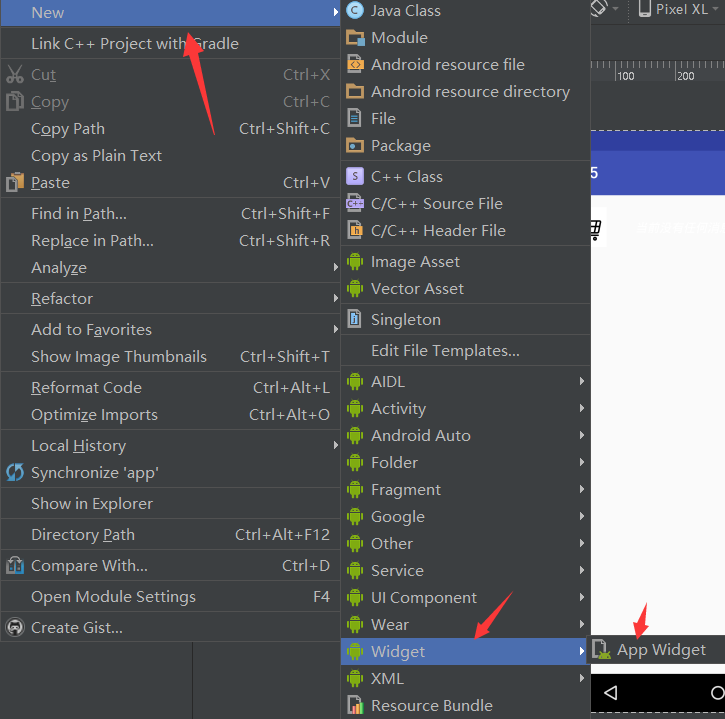
|  |  |  |  |
| --- | --- | --- | --- |
| 1. widget 初始情况如右图1。 2. 点击widget可以启动应用，并在widget随机推荐一个商品，如图2 3. 点击widget跳转到该商品详情界面，如图3 |  |  |  |
| 图1 | 图2 | 图3 |
| 1. 点击购物车图标，widget相应更新如图4 2. 点击widget跳转到购物车界面，如图5 3. 实现方式要求:启动时的widget的更新通过静态广播实现，点击购物车图标时候widget的更新通过动态广播实现。 | |  |  |
|  | | 图4 | 图5 |

## 课堂实验结果

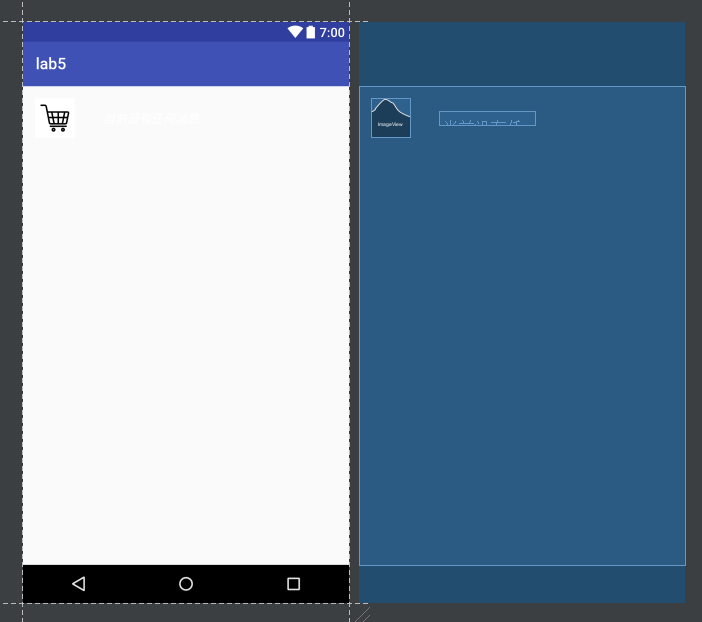
1. 实验截图：
   * 1. 虚拟机上效果如下：

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

1. 关键步骤：
2. 创建一个widget。
   1. Android studio已经提供了一个widget模板给我们使用，所以我们只需要先创建一个widget模板之后，在模板的基础上进行修改即可。



* 1. widget界面修改如下，直接展示XML的效果



* 1. 修改widget\_info.xml文件，设置widget的大小属性和布局

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <appwidget-provider xmlns:android="http://schemas.android.com/apk/res/android"  android:initialKeyguardLayout="@layout/new\_app\_widget"  android:initialLayout="@layout/new\_app\_widget"  android:minHeight="40dp"  android:minWidth="250dp"  android:previewImage="@mipmap/ic\_launcher\_round"  android:resizeMode="horizontal|vertical"  android:updatePeriodMillis="86400000"  android:widgetCategory="home\_screen|keyguard"></appwidget-provider> |

此处设置最小宽度为4格，最小高度为1格。

1. 实现点击widget启动并推荐一个商品。

此处基于上次的实验，我们只需要实现通过点击初始widget启动，启动后基于上次实验，会发出一个推送商品的广播，所以此时我们只要接收广播然后更新widget信息。

* 1. 实现widget启动应用，此时我们需要在widget被添加后第一次更新时，设置需要widget信息，和添加点击事件。此时我们只需要在onUpdate进行修改，并设置一个flag进行是否是第一次更新的判断。

|  |
| --- |
| @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) { //  if(flag){ //第一次更新  Intent i = new Intent(context, MainActivity.class);//添加跳转事件  PendingIntent pendingIntent = PendingIntent.*getActivity*(context, 0, i, PendingIntent.*FLAG\_UPDATE\_CURRENT*);  RemoteViews views = new RemoteViews(context.getPackageName(), R.layout.*new\_app\_widget*);  views.setTextViewText(R.id.*appwidget\_text*, "当前没有任何消息"); //设置初始信息  views.setImageViewResource(R.id.*appwidget\_image*, R.mipmap.*ic\_launcher*);  views.setOnClickPendingIntent(R.id.*appwidget*, pendingIntent);//设置点击事件  ComponentName me = new ComponentName(context, NewAppWidget.class);  AppWidgetManager.*getInstance*(context).updateAppWidget(me, views);//更新  }else  *updateAppWidget*(context,appWidgetManager,appWidgetId);  } } |

* 1. 应用开始后widget随机推荐一个商品，此时我们只需要接收应用运行后发出的随机推送商品的静态广播即可。

首先要现在manfast中给widget添加静态广播接收

|  |
| --- |
| <receiver android:name=".NewAppWidget">  <intent-filter>  <action android:name="android.appwidget.action.APPWIDGET\_UPDATE" />  <action android:name="cn.chonor.lab5.staticreceiver" />  </intent-filter>  <meta-data  android:name="android.appwidget.provider"  android:resource="@xml/new\_app\_widget\_info" /> </receiver> |

然后我们在widget类中添加静态广播接收者，所以此时需要重写onReceive函数进行接收，接收后需要更新信息和设置点击事件。

|  |
| --- |
| @Override  public void onReceive(Context context, Intent intent){  final String action = intent.getAction();  if(action.equals(*STATICACTION*)) {//判断是静态广播  flag = false;  Bundle extras = intent.getBundleExtra("mainActivity");  int postiton = extras.getInt("position"); //获得数据位置  Intent i = new Intent(context, Good\_Info.class);//设置转转跳  Bundle bundle = new Bundle();  bundle.putInt("position", postiton);//设置转跳时传输数据  bundle.putParcelableArrayList("data", data.getGood\_list());  bundle.putParcelableArrayList("cart", data.getCart\_list());  i.putExtra("mainActivity", bundle);  RemoteViews views = new RemoteViews(context.getPackageName(), R.layout.*new\_app\_widget*);//更新widget  PendingIntent pendingIntent = PendingIntent.*getActivity*(context, 0, i, PendingIntent.*FLAG\_UPDATE\_CURRENT*);  views.setTextViewText(R.id.*appwidget\_text*, data.getGood\_list\_index(postiton).getGoodName() + "仅售" + data.getGood\_list\_index(postiton).getGoodPrice() + "!");  views.setImageViewResource(R.id.*appwidget\_image*, data.ID[data.map.get (data.getGood\_list\_index(postiton).getGoodName())]);  views.setOnClickPendingIntent(R.id.*appwidget*, pendingIntent);//设置点击事件  ComponentName me = new ComponentName(context, NewAppWidget.class);  AppWidgetManager.*getInstance*(context).updateAppWidget(me, views);//更新  }  super.onReceive(context, intent); } |

此处需要的开启应用后,widget推荐商品已经完成。

1. 点击购物车更新widget，点击widget转跳到购车车界面。
   1. 此处在widget内增加一个静态函数，然后在商品信息界面接收到动态广播后调用此函数进行widget更新。

|  |
| --- |
| static void upDataDynamic(Context context,PendingIntent pendingIntent, String names){  Data data=new Data();  RemoteViews views = new RemoteViews(context.getPackageName(), R.layout.*new\_app\_widget*);  views.setTextViewText(R.id.*appwidget\_text*, names+"已添加到购物车");//更新信息  views.setImageViewResource(R.id.*appwidget\_image*, data.ID[data.map.get(names)]);  views.setOnClickPendingIntent(R.id.*appwidget*, pendingIntent);//设置点击事件  ComponentName me = new ComponentName(context, NewAppWidget.class);  AppWidgetManager.*getInstance*(context).updateAppWidget(me, views);//更新 } |

* 1. 之后在动态广播接收时调用这个函数进行更新

|  |
| --- |
| public class DynamicReceiver extends BroadcastReceiver {  public static final String *DYNAMICATION*= "cn.chonor.lab5.dynamicreceiver";  public DynamicReceiver(){}  @Override  public void onReceive(Context context, Intent intent) {  if(intent.getAction().equals(*DYNAMICATION*)){  Intent intent1 =new Intent (context,Shoppingcart.class);//点击事件和传输  PendingIntent pi = PendingIntent.*getActivity*(context, 0, intent1, PendingIntent.*FLAG\_CANCEL\_CURRENT*);  NewAppWidget.*upDataDynamic*(context,pi,names);  }  }  } |

1. 实验中遇到的困难和解决思路：
2. 动态广播接收的时候无法再widget内注册接收器，解决方法：在widget的java文件内写一个静态函数，在商品详情界面的时候进行接收广播的同时调用静态函数修改widget。
3. 在静态函数修改widget的时候，点击事件无效，但是可以修改。之后发现是context问题，直接改传一个PendingIntent使得点击事件可以正常使用。

## 课后实验结果

本次实验只有widget的使用，无额外扩展。

修复了一下不合理的地方，当我们清空购物车列表的此时widget还是显示了推送信息，但是此时我们打开的是一个空的购物车。

|  |  |
| --- | --- |
|  |  |

此时修复一下，当我们清空购物车的时候还原widget为初始状态。



此时在更新里添加判断。

|  |
| --- |
| static void upDataDynamic(Context context,PendingIntent pendingIntent, String names){  Data data=new Data();  RemoteViews views = new RemoteViews(context.getPackageName(), R.layout.*new\_app\_widget*);  if(names.equals("")){ //购物侧清空  views.setTextViewText(R.id.*appwidget\_text*, "当前没有任何消息"); //设置初始信息  views.setImageViewResource(R.id.*appwidget\_image*, R.mipmap.*ic\_launcher*);  }else { //动态广播更新  views.setTextViewText(R.id.*appwidget\_text*, names + "已添加到购物车");//更新信息  views.setImageViewResource(R.id.*appwidget\_image*, data.ID[data.map.get(names)]);  }  views.setOnClickPendingIntent(R.id.*appwidget*, pendingIntent);//设置点击事件  ComponentName me = new ComponentName(context, NewAppWidget.class);  AppWidgetManager.*getInstance*(context).updateAppWidget(me, views);//更新 } |

购物车被删除为空时调用更新

|  |
| --- |
| builder.setPositiveButton("确定", new DialogInterface.OnClickListener() {  @Override //设置确定按钮动作  public void onClick(DialogInterface dialogInterface, int i) {  data.removeCart\_list\_index(position);  myAdapter.notifyDataSetChanged();  Toast.*makeText*(getApplicationContext(), "成功删除 " + tmp, Toast.*LENGTH\_SHORT*).show();  if(data.getCart\_list().size()==0) {//购物车为空  Intent intent1 = new Intent(Shoppingcart.this, MainActivity.class);//点击事件和传输  PendingIntent pi = PendingIntent.*getActivity*(Shoppingcart.this, 0, intent1, PendingIntent.*FLAG\_CANCEL\_CURRENT*);  NewAppWidget.*upDataDynamic*(Shoppingcart.this, pi, "");  }  }  } |

## 实验思考及感想

本次实验是widget使用，主要就是涉及到widget的创建，更新，和接收广播。Widget本身就是一个接收者，所有我们直接使用onReceive就可以接收了。

实验中因为android studio本身已经为我们内置了一个模板了，我们可以直接使用模板进行代码，而且其对于widget的配置也是可以通过我们输入几个参数就自动配置好的。剩下的就是我们写一下如何接收到之前实验中发出的广播然后进行更新。Widget的用法比较局限，所以没想到什么可以进行拓展的地方。