中山大学数据科学与计算机学院本科生实验报告

（2017 年秋季学期）

课程名称：手机应用平台开发　　　　　　　　任课教师： 刘宁

年级	15	专业（方向）	计应
学号	15331059	姓名	邓松华
电话	13542996213	Email	dengsonhua@163.com
开始日期	2017.10.19	完成日期	2017.10.19

一、 实验题目

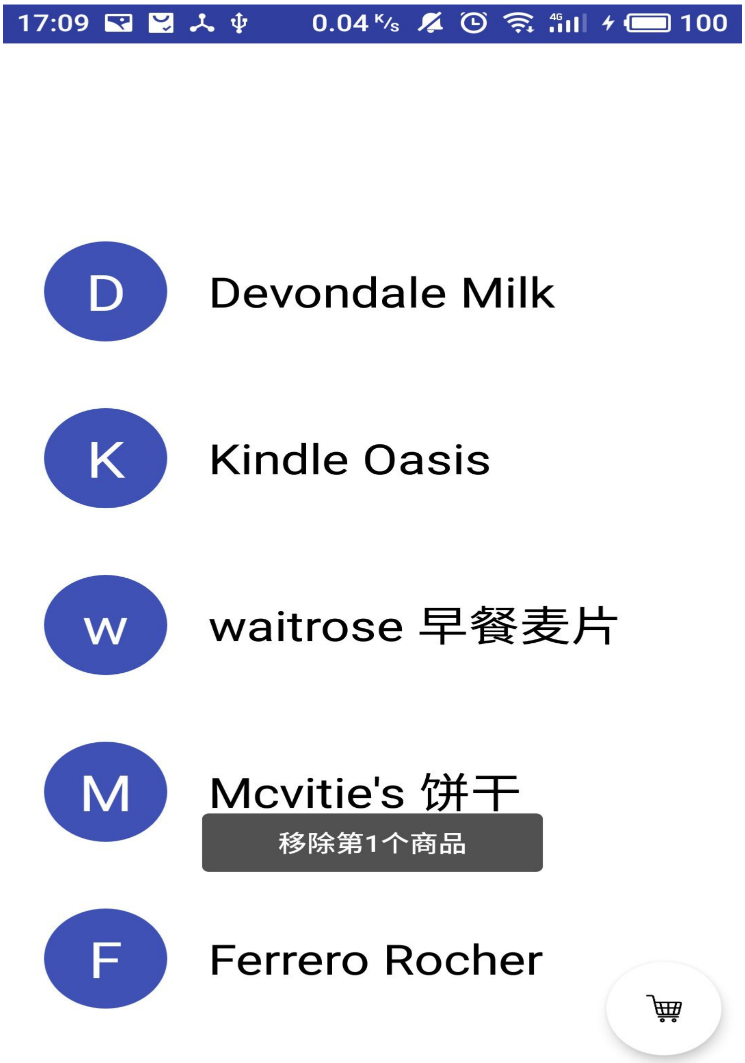
Intent、Bunble 的使用以及 Recycleview、ListView 的应用

二、 实现内容

模拟商品表

三、 实验结果

（1）实验截图



Devondale Milk

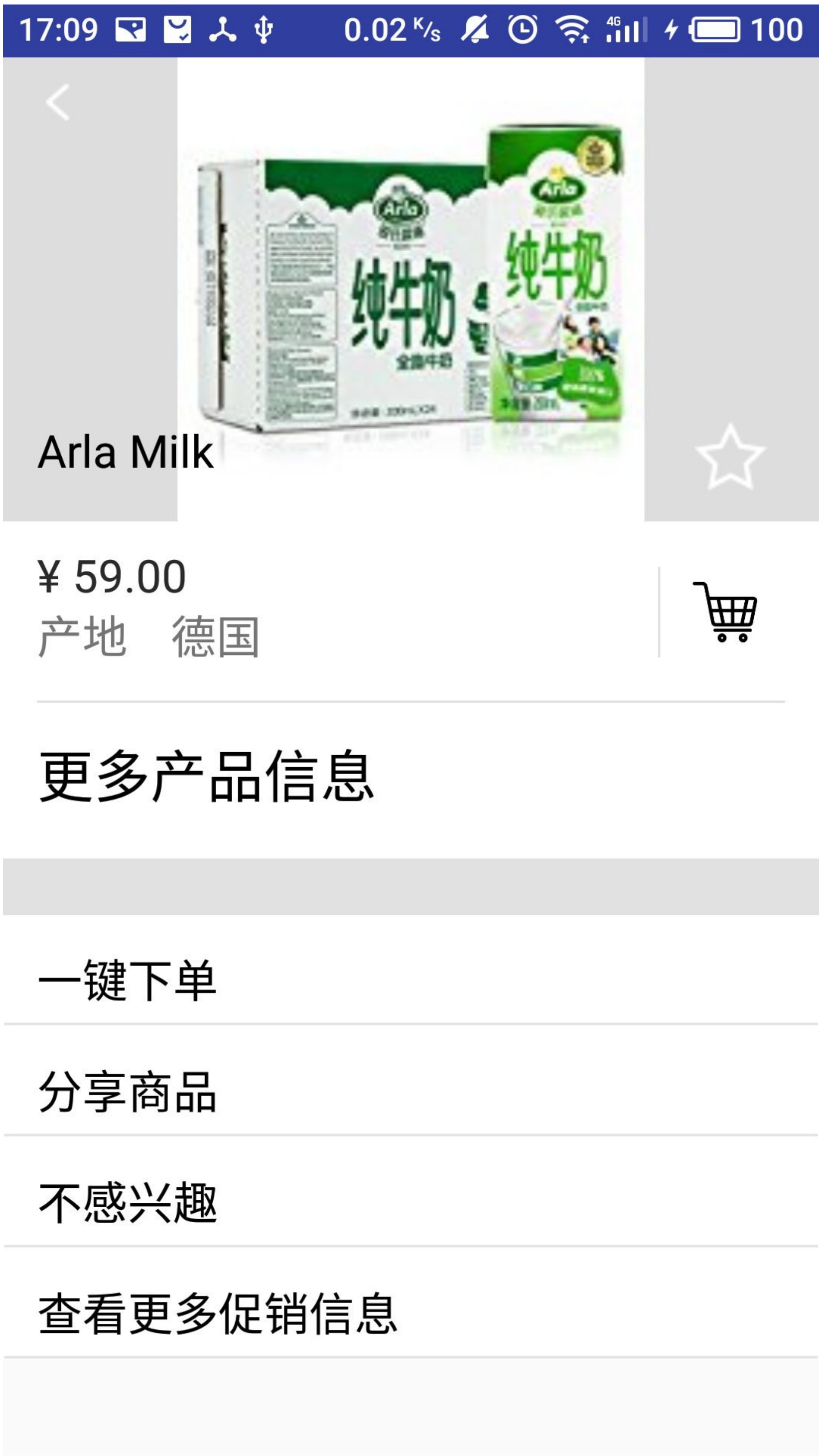
Kindle Oasis

waitrose 早餐麦片

Mcvitie's 饼干

移除第1个商品

Ferrero Rocher

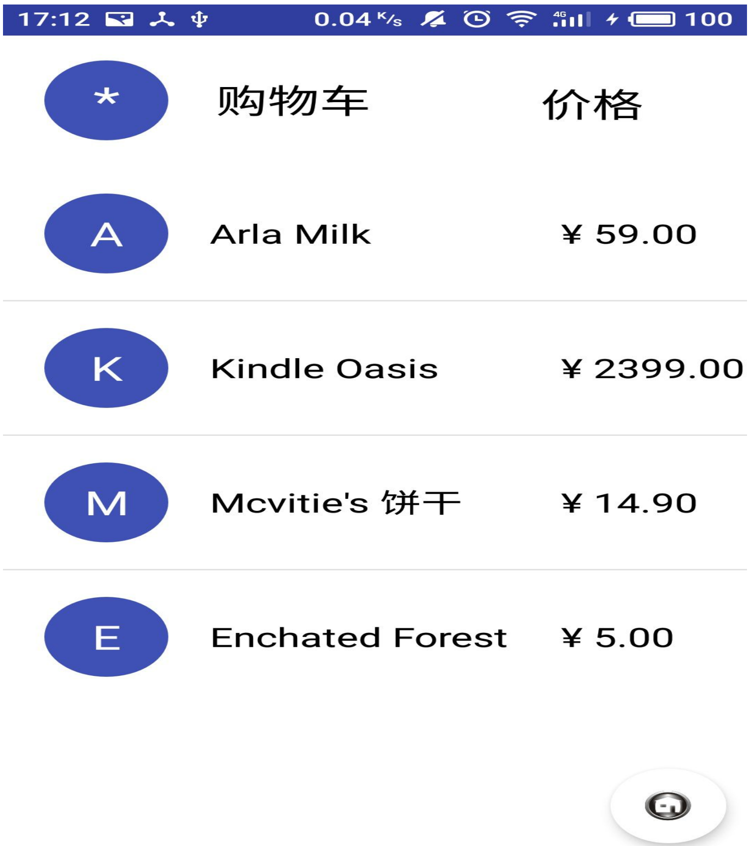


Arla Milk

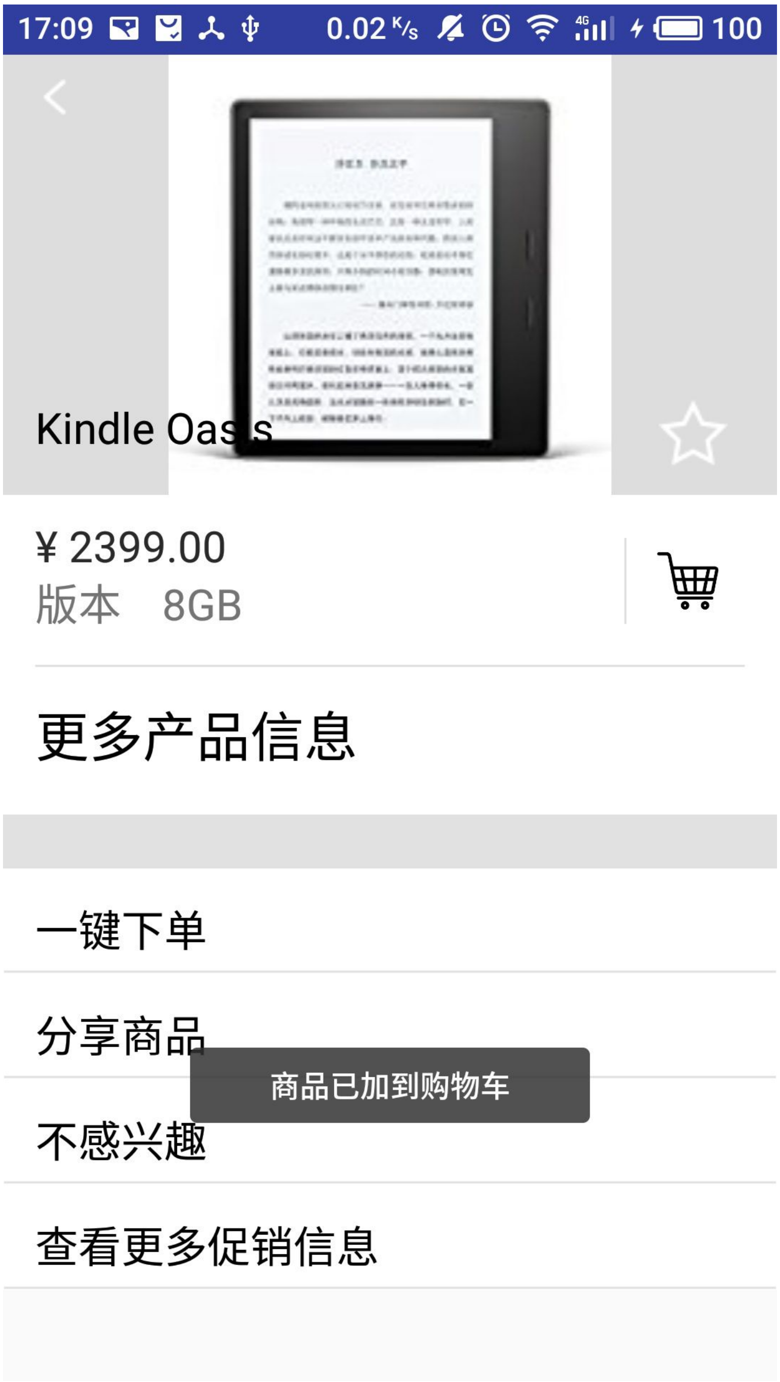
¥ 59.00

产地　德国

更多产品信息



*	购物车	价格
A	Arla Milk	¥ 59.00
K	Kindle Oasis	¥ 2399.00
M	Mcvitie's 饼干	¥ 14.90
E	Enchated Forest	¥ 5.00

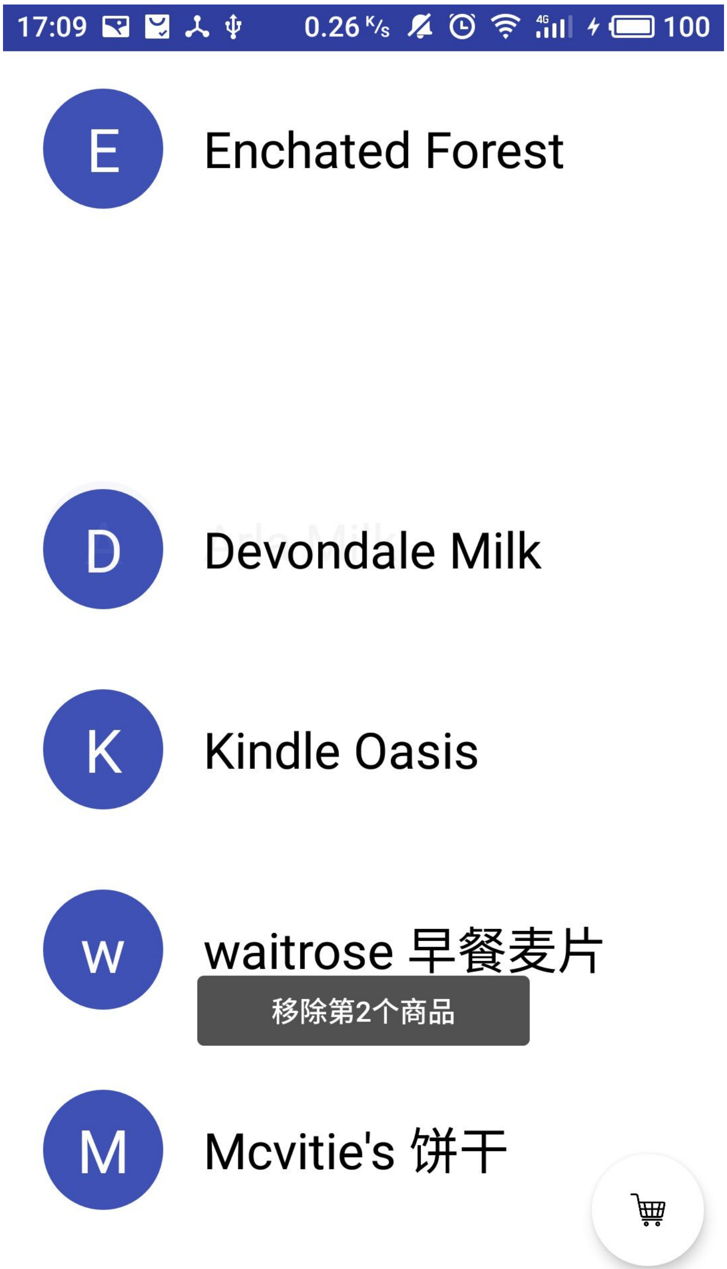


Kindle Oasis

¥ 2399.00

版本　8GB

商品已加到购物车



17:09 0.26 K/s 100

E Enchated Forest

D Devondale Milk

K Kindle Oasis

w waitrose 早餐麦片

移除第2个商品

M Mcvitie's 饼干

（2）实验步骤以及关键代码

A.实现商品列表 RecycleView

关键步骤：

1. 点击和长点击的实现。通过在 adapter 类中引入借口，并在 activity 的类中实现接口

```
public ItemsAdapter(List<Items> r) {
    rItems=r;
}

private OnItemClickListener clickListener;

public void setClickListener(OnItemClickListener clickListener) {
    this.clickListener = clickListener;
}

public static interface OnItemClickListener {
    void onClick(View view,Items flag,int position);
}
private OnItemLongClickListener longClickListener;

public void setLongClickListener(OnItemLongClickListener clickLongListener) {
    this.longClickListener = clickLongListener;
}

public static interface OnItemLongClickListener {
    void onLongClick(View view,Items flag,int position);
}
```

2. 实现接口，点击后，打开详情页面，长点击后删除。

```
    adapter.setClickListener(new ItemsAdapter.OnItemClickListener() {
    public void onClick(View view, Items flag,int position) {
        Intent intent=new Intent(MainActivity.this,detail.class);
        intent.putExtra("name",flag.getName());
        intent.putExtra("rPosition",position);
        intent.putExtra("lPosition",-1);
        intent.putExtra("message",flag.getMessage());
        intent.putExtra("price",flag.getPrice());
        intent.putExtra("imageId",flag.getImageId());
        intent.putExtra("style",flag.getStyle());
        startActivityForResult(intent,1);
    }
});
    adapter.setLongClickListener(new ItemsAdapter.OnItemLongClickListener(){
         public void onLongClick(View view, Items flag,int position) {
         int pos=position+1;
         Toast.makeText(getApplicationContext(),"移除第"+pos+"个商品",Toast.LENGTH_SHORT).show();
        rItems.remove(position);
    }
});
```

3. 加入动画

```
adapter= new ItemsAdapter(rItems);
ScaleInAnimationAdapter animationAdapter=new ScaleInAnimationAdapter(adapter);
animationAdapter.setDuration(1000);
recyclerView.setAdapter(animationAdapter);
SlideInUpAnimator animator = new SlideInUpAnimator(new OvershootInterpolator(1f));
recyclerView.setItemAnimator(animator);
recyclerView.getItemAnimator().setRemoveDuration(1000);
```

B. 实现悬浮按钮

关键：切换图片，并切换 visible，通过 floa 来判断

```
floatingActionButton.setOnClickListener(new View.OnClickListener(){
    @Override

    public void onClick(View view) {
          if(floa==1) {
            FloatingActionButton floatingActionButton=(FloatingActionButton)
findViewById(R.id.floatButton);
            floatingActionButton.setImageResource(R.drawable.shoplist);
            RecyclerView recyclerView=(RecyclerView) findViewById(R.id.itemsRecyclyer);
            recyclerView.setVisibility(View.VISIBLE);
            LinearLayout linearLayout=(LinearLayout)findViewById(R.id.layoutList);
            linearLayout.setVisibility(View.GONE);
            floa=0;
        }
        else {
             FloatingActionButton floatingActionButton=(FloatingActionButton)
findViewById(R.id.floatButton);
             floatingActionButton.setImageResource(R.drawable.mainpage);
             RecyclerView recyclerView=(RecyclerView) findViewById(R.id.itemsRecyclyer);
             recyclerView.setVisibility(View.GONE);
             LinearLayout linearLayout=(LinearLayout)findViewById(R.id.layoutList);
             linearLayout.setVisibility(View.VISIBLE);
             floa=1;
        }
    }
});
```

C. 实现购物车

关键步骤：

1. 购物车的 item 的确定。通过从详情页面的放回数据确定（一个商品列表打开，一个购物车打开）。

```
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    if(data!=null) {
        int position=data.getExtras().getInt("rPosition");
        if(position !=-1) {
            Items items=rItems.get(position);
            lItems.add(items);
        }
        else {
            position=data.getExtras().getInt("lPosition");
            Items items=lItems.get(position);
            lItems.add(items);
        }
        adapter1.notifyDataSetChanged();
    }
}
```

2. 点击和长点击，直接重载自带函数

```
listView.setOnItemClickListener(new AdapterView.OnItemClickListener(){
    @Override    public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
        Items flag=lItems.get(position);
        Intent intent=new Intent(MainActivity.this,detail.class);
        intent.putExtra("name",flag.getName());
        intent.putExtra("lPosition",position);
        intent.putExtra("rPosition",-1);
        intent.putExtra("message",flag.getMessage());
        intent.putExtra("price",flag.getPrice());
        intent.putExtra("imageId",flag.getImageId());
        intent.putExtra("style",flag.getStyle());
        startActivityForResult(intent,1);
    }
});
listView.setOnItemLongClickListener(new AdapterView.OnItemLongClickListener(){
    public boolean onItemLongClick(AdapterView<?> parent, View view, int position, long id) {
        AlertDialog.Builder dialog=new AlertDialog.Builder(MainActivity.this);
        dialog.setTitle("移除商品");
        final int pos=position;
       dialog.setMessage("从购物车移除"+lItems.get(position).getName()+"?");
        dialog.setNegativeButton("取消",new DialogInterface.OnClickListener(){
            @Override
               public void onClick( DialogInterface dialogInterface,int which) {
            }
        });
        dialog.setPositiveButton("确定",new DialogInterface.OnClickListener(){
            @Override
             public void onClick( DialogInterface dialogInterface,int which) {
                lItems.remove(pos);
                adapter1.notifyDataSetChanged();
            }
        });
        dialog.show();
        return true;
    }
});
```

D. 详情页面的设置

关键步骤

1. 分析主页面的数据

```
Bundle extra=getIntent().getExtras();
String name=extra.getString("name");
String price=extra.getString("price");
String style=extra.getString("style");
String message=extra.getString("message");
int imageId=extra.getInt("imageId");
```

```
final int rp=extra.getInt("rPosition");
final int lp=extra.getInt("lPosition");
```

2. 修改商品的信息

```
TextView nameDetail=(TextView) findViewById(R.id.nameDetail);
nameDetail.setText(name);
ImageView imageDetail =(ImageView) findViewById(R.id.imageDetail);
imageDetail.setImageResource(imageId);
TextView styleDetail=(TextView) findViewById(R.id.styleDetail);
styleDetail.setText(style);
TextView messageDetail=(TextView) findViewById(R.id.messageDetail);
messageDetail.setText(message);
TextView priceDetail=(TextView) findViewById(R.id.priceDetail);
priceDetail.setText(price);
```

3. 修改星星的图片。这里是通过把当前的图片与空星图片比较。若相等则换满星，否则换空星

```
ImageView starDetail=(ImageView) findViewById(R.id.imageStar);
starDetail.setOnClickListener(new View.OnClickListener(){
    @Override

    public void onClick(View view) {
        ImageView starDetail=(ImageView) findViewById(R.id.imageStar);
        final Bitmap bmap = ((BitmapDrawable)starDetail.getDrawable()).getBitmap();
        Drawable myDrawable = getDrawable(R.drawable.empty_star);
        final Bitmap myStar = ((BitmapDrawable) myDrawable).getBitmap();
        if (bmap.sameAs(myStar)) {
            starDetail.setImageResource(R.drawable.full_star);
        }
        else {
            starDetail.setImageResource(R.drawable.empty_star);
        }
    }
});
```

4. 点击购物车，点击后返回给主活动数据

```
ImageView shop=(ImageView) findViewById(R.id.shopDetail);
shop.setOnClickListener(new View.OnClickListener(){
    @Override

    public void onClick(View view) {
        Toast.makeText(view.getContext(),"商品已加到购物车",Toast.LENGTH_SHORT).show();
        Intent intent=new Intent();
        intent.putExtra("lPosition",lp);
        intent.putExtra("rPosition",rp);
        setResult(0,intent);
    }
});
```

四、实验思考及感想

这次实验不算很难。但代码量算是很多啊。做起来也挺繁琐。ListView 和 RecycleView 的 adapter 设置，查到资料感觉就没什么难度。我觉得最难的应该是购物车商品和消息的传递的构思。如果把购物车放到另一个活动上，那更加酸爽了。其他一些细节也挺烦的，做起来耗时间。最主要网上有一些博客是错误或是很复杂，在查阅的时候浪费了很多时间。哎，还需努力学习。