

# 中山大学移动信息工程学院本科生实验报告

---

课程名称： 移动应用开发

任课老师： 刘宁

K	V
年级	2015
专业（方向）	计算机应用软件
学号	15331435
姓名	周长安
电话	15692430700
Email	zchangan@163.com
开始日期	2017年11月18日
完成日期	2017年11月26日

## 一、实验题目

安卓其中项目——三国武将电子词典

## 二、实现内容

- 激活相关的三个页面
- 数据库增删和随机相关接口
- 部分广播事件
- 按钮动画效果
- 新建和修改页面的合并和修改

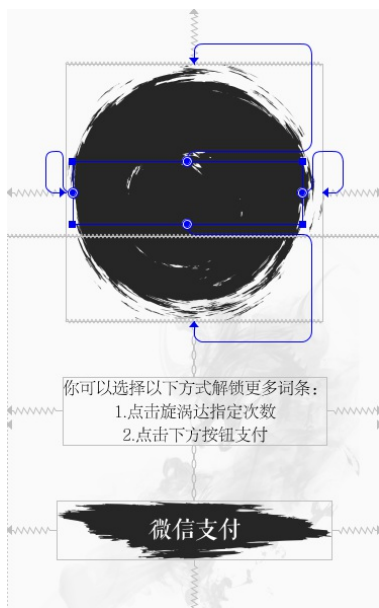
## 三、实验结果

(1) 实验截图



## (2) 实验步骤以及关键代码

### 激活页面



上方的旋涡按钮应该有大写数字表示需要点击的次数

其中，需要点击次数的存取在SharedPreferences中完成

```
sharedPref = getPreferences(Context.MODE_PRIVATE);
// 读
unlockCount = sharedPref.getInt(getString(R.string.saved_unlock_count), 2);
// 写
SharedPreferences.Editor editor = sharedPref.edit();
unlockCount *= 2;
editor.putInt(getString(R.string.saved_unlock_count), unlockCount);
editor.apply();
```

## 小写数字转大写数字

```
public static String num2String(int number) {
    if (number == 0) return "解锁";
    String[] NUMBER_TABLE = {"零", "壹", "贰", "叁", "肆", "伍", "陆", "柒", "捌", "玖"};
    String[] UNIT_TABLE = {"", "拾", "佰", "仟", "万"};
    String numberString = new
StringBuffer(String.valueOf(number)).reverse().toString().trim();
    StringBuilder builder = new StringBuilder();
    for (int i = 0; i < numberString.length(); i++) {
        builder.append(UNIT_TABLE[i]);
        builder.append(NUMBER_TABLE[numberString.charAt(i) - 48]);
    }
    builder.reverse();
    String result = builder.toString();
    result = result.replaceAll("零[拾佰仟]", "零");
    result = result.replaceAll("零万", "万");
    while (result.charAt(result.length() - 1) == '零') {
        result = result.substring(0, result.length() - 1);
    }
    return result;
}
```

## 按钮动画效果

### build.gradle

```
compile 'com.daimajia.easing:library:2.0@aar'
compile 'com.daimajia.androidanimations:library:2.3@aar'
```

## 触发动画

```
findViewById(R.id.btn_unlock_pay).setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View view) {
        YoYo.with(Techniques.Pulse).duration(200).playOn(view);
    }
});
```

## 解锁过程



动画结束后进入下一个activity

```
fadeIn.setAnimationListener(new Animation.AnimationListener() {
    @Override
    public void onAnimationStart(Animation animation) {
    }

    @Override
    public void onAnimationEnd(Animation animation) {
        Intent intent = new Intent(context, UnlockSuccessActivity.class);
        startActivity(intent);
        context.finish();
        context.overridePendingTransition(android.R.anim.fade_in,
        android.R.anim.fade_out);
    }

    @Override
    public void onAnimationRepeat(Animation animation) {
    }
});
```

解锁成功页面



该页面通过调用数据库接口获取随机解锁的结果

```
List<Character> res =
    DbReader.getInstance(this).getRandomCharacters(Tools.random(1, 10));
```

并将解锁结果通过广播通知MainActivity更新

```
Intent broadcast = new Intent(MainActivity.NOTIFY_ITEMS_ADDITION);
broadcast.putExtra("characters", (ArrayList<Character>) res);
sendBroadcast(broadcast);
```

将结果通过CommonAdapter渲染到RecyclerView

```
final CommonAdapter<Character> adapter = new CommonAdapter<Character>(this,
    R.layout.character_item, res) {
    @Override
    public void convert(ViewHolder holder, Character data) {
        TextView name = holder.getView(R.id.name);
        name.setText(data.name);
        TextView belong = holder.getView(R.id.belong);
        belong.setText(data.belong);
        TextView description = holder.getView(R.id.abstract_description);
        description.setText("\t\t\t\t" + data.abstractDescription);
    }
};
```

数据库增删和随机接口

将一个列表的所有内容加入数据库

```
public void addCharacters(List<Character> list) {
    db.beginTransaction();
    for (Character ch : list) {
        ch._id = db.insert(TABLE_CHARACTER, null, ch.toContentValues());
    }
    db.setTransactionSuccessful();
    db.endTransaction();
}
```

删除

```
public void deleteCharacter(Character ch) {
    db.delete(DbHelper.TABLE_CHARACTER, Character._ID + " = ?", new String[]
    {String.valueOf(ch._id)});
}
```

通过Order By Random和Limit x获取随机数量的数据

```
public List<Character> getRandomCharacters(int count) {
    String select = "SELECT * " +
        "FROM " + DbHelper.TABLE_CHARACTER + " as c " +
        "WHERE c._id NOT IN ( " +
        "SELECT character_id " +
        "FROM " + DbHelper.TABLE_OWN + " " +
        ")" +
        "ORDER BY RANDOM() LIMIT " + count;
    Cursor cursor = db.rawQuery(select, null);
    List<Character> res = getAllCharactersByCursor(cursor);
    cursor.close();
    return res;
}
```

### (3) 实验遇到困难以及解决思路

动态广播无法接收的问题

忘记注册Filter

```
IntentFilter filter = new IntentFilter();
filter.addAction(NOTIFY_ITEM_DELETION);
filter.addAction(NOTIFY_ITEMS_ADDITION);
filter.addAction(NOTIFY_ITEMS_ADDITIONS);
```

```
filter.addAction(NOTIFY_ITEMS_MODIFY);  
Receiver receiver = new Receiver();  
registerReceiver(receiver, filter);
```

## 编辑页面复用新增页面

一开始编辑页面和新增页面是两个几乎一模一样的页面，为了增加代码复用程度，我对两个页面进行了合并。

通过判断`Intent`中是否有`Charactor`数据来判断是哪种类型，维护`isEdit`布尔值，最后决定按下保存按钮时候的表现。