

飞花令 android 开发设计

山东大学计算机科学与技术学院--张涵

一、开发背景

行酒令是筵宴上助兴取乐的饮酒游戏，饮酒行令在士大夫中特别风行，他们还常常赋诗撰文予以赞颂。飞花令，就是酒令中的一种。属雅令。行使“飞花令”时，在诗句中第几个字为“花”即按一定顺序列次由第几个人喝酒。

而近日在《中国诗词大会》中，央视引进并改良了“飞花令”，为每场比赛设置一个关键字，不再仅用“花”字，而是增加了“云”、“春”、“月”、“夜”等诗词中的高频字，双方 必须在极短时间内完整说出一联诗句，这十分考查选手的诗词储备。

而为了能够更加快速准确地找到要求的诗句，就在 android studio 平台上开发了这么一款 app。

二、开发环境

Windows 10 x64

Android Studio 2.2.3

HUAWEI EVA-AL00(Android 7.0,API 24)

三、开发设计

作为一个初级的简单 app，飞花令只需要实现以下几个基本功能即可（做为一个小白，再复杂的功能我也做不出来了...QAQ）

- 1、拥有输入框，输出框和按钮，能够输入要查询的字，并且输出相应的诗句。
- 2、将用户查询的字在诗句中以高亮显示。
- 3、界面简洁友好。

四、具体实施

- 1、先在 eclipse 上编写主体代码，熟悉具体构成。
- 2、安装 Android Studio

在安装 Android Studio 的过程中遇到了很多问题，第一次是 SDK Manager 打不开，后来在文件目录里强行打开了，

第二次是出现未知错误，找了半天找不到哪里错了（哭...），第三次是 SDK 里的工具包没有安装完整，第四次是创建新项目之后 Android Studio 平台直接崩了（我也很绝望啊...），直至第七次安装，我才彻底搞定它！至此，才开始正式编写飞花令的代码。

3、正式编写代码

- 1) 创建输入组件 `edittext`，输出组件 `textview` 和按钮 `button`

```
private TextView output;
private EditText input;
private Button research;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    output = (TextView) findViewById(R.id.textView);
    input = (EditText) findViewById(R.id.editText);
    research = (Button) findViewById(R.id.button);
}
```

- 2) 在 监 听 事 件 中 写 入 主 代 码

```
SpannableStringBuilder style1, style2 = new SpannableStringBuilder();

String single = input.getText().toString();
int i = 0;
String res = "", temp = "";
try {
    InputStream inputStream = getResources().openRawResource(R.raw.shi);
    InputStreamReader read = new InputStreamReader(inputStream, "UTF-8");
    BufferedReader bufferedReader = new BufferedReader(read);
    String lineTxt;
    while ((lineTxt = bufferedReader.readLine()) != null) {
        if (lineTxt.contains(": ")) {
            temp = lineTxt;
        } else if (lineTxt.contains(single)) {
            i++;
            style1 = new SpannableStringBuilder(lineTxt);
            style1.setSpan(new BackgroundColorSpan(Color.RED), lineTxt.indexOf(single), lineTxt.indexOf(single)+1, Spannable.SPAN_EXCLUSIVE_EXCLUSIVE);
            style1.setSpan(new BackgroundColorSpan(Color.RED), lineTxt.lastIndexOf(single), lineTxt.lastIndexOf(single)+1, Spannable.SPAN_EXCLUSIVE_EXCLUSIVE);
            style2.append(res).append(": ");
            style1.append("\n").append(" ").append("——").append(temp).append("\n");
            style2.append(style1);
            res = "";
        }
    }
}
```

- 3) 将关键字以高亮显示

（代码在上图中）

- 4) 用 `toast` 方法显示查询总数

```
Toast.makeText(MainActivity.this, "共找到" + i + "条", Toast.LENGTH_SHORT).show();
```

五、优化改进

- 1、滚动设置

初次完成的程序在运行时不能进行滚动，有很多诗句看不到，在查看了 dalao 们的报告之后，找到了两句代码，直

接复制粘贴了（主要是我也不会...）

2、输出的诗句中包含题目

因为文本格式固定，所以很容易就可以以数字为关键字来进行排除，将题目和作者并放一边。

3、同时输出题目和作者

```
if(lineTxt.contains(":")) {  
    temp=lineTxt;
```

只有这么一行代码就实现了这个功能，并且格式化输出就行了。

4、设置背景图片

经过在网上的查找和请教舍友 dalao，我明白了设置背景图片的方法，将图片放在 **drawable** 文件里，再在 **activity-main.xml** 里添加 `android:background="@drawable/backing"`（“ ” 里是图片的名称）就可以了。不过要注意图片需要是.png 格式。

六、不足之处

因为是第一次自己开发 app，而且自己还是小白（这个是关键），所以这个程序有很多不足之处。

1、字体高亮

本想只把字体变色变亮，最后却把字体整个方块变亮了...

2、字体大小

我找到了一行调整字体大小的代码，本想再弄两个按钮来让用户自己调整适合的字体大小，但是技术有限，未能成功...

3、字体颜色

可能有些字在背景图片上看不太清，我也想过弄个选项，让用户选择喜欢的颜色，但还是有了设想没有成功实现...

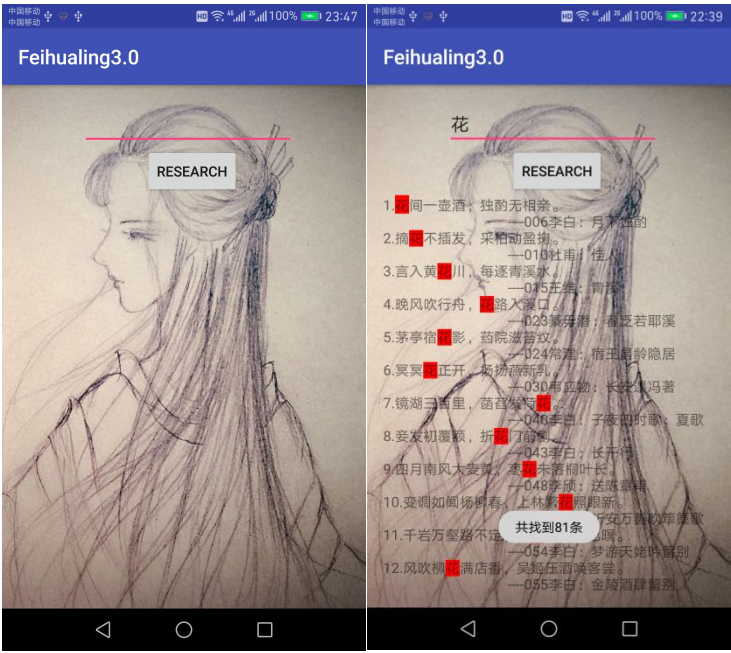
4、.....

个人认为不足之处还有好多，例如交互不足，文本库太小等等，需要以后继续改进。

七、总结感悟

第一次使用 **Android Studio** 进行 app 开发，所以遇到了很多问题，不仅仅是安装过程中，在编写代码过程中的问题也很多，虽然同是 **java** 语言，但是这里面的语法确实我们所没有学过的，不得不一直求教度娘。

不过还好好的是，有一批大佬已经为我们探好了路，老师也在群里和我们积极互动，虽然困难，但是最终也完成了这个项目，也从中学到了很多东西，特别是自主学习能力，有了很大提高。



最后展示一下最终结果：