

2017 年夏季 Java 小学期大作业 实验报告

王逸松 2015011369

1. 功能实现

功能	百分比 (%)	是否实现
要保证程序在安卓机上正常运行，测试过程中程序不崩溃	5	是
布局合理，点击处理正确	10	是
分类列表的删除和添加操作	10	是
正确显示新闻列表的消息，布局和展示，点击进入新闻详情页面正确。	10	是
实现新闻的本地存储，看过的新闻列表在离线的环境下也可以浏览	10	是
上拉获取更多新闻	5	是
新闻是否看过的页面灰色标记	5	是
新闻搜索	5	是
使用微信、微博等SDK分享，新闻详情页面点击分享可以分享到常用的app，分享内容带有新闻摘要、URL 和图片	10	
新闻详情页面点击收藏的添加和删除，实现收藏新闻的本地存储。收藏也的正确展示，点击可以进入新闻详情等	10	是
根据用户看过的新闻推荐相关的新闻，参考今日头条等	10	
可以语音读出新闻等	20	是
用户可以跳转到相应的百科词条等	5	
新闻屏蔽功能，通过进一步询问用户想要屏蔽掉关于什么内容的新闻而实现基于关键词的屏蔽等	5	
夜间模式，用户可以调整背景色等	5	

文字模式和图片模式转换，文字模式不显示图片，帮助用户节省流量。	5	是
流畅性强	希望得到 5~10 分	是
界面颜值高	视效果	
根据新闻文本补上相关的图片(因为部分新闻内容没有图片)	视效果	
使用了较好的框架	酌情加分	
有较完整的单元测试	5	
使用了 github 等好用的代码版本管理工具	2	是
maven 或者 gradle 等项目管理工具	2	是

总计： $(5+10+10+10+10+5+5+5+10)+(20+5+10+2+2)=70+39=109$ 分

2. 小组分工

(略)
小组中只有我一个人。

3. 具体实现

- 使用 Android Studio 开发，未使用任何框架
- 主体界面使用 Android Studio 提供的模板（Navigation Drawer类的模板）制作
- 新闻列表使用 RecyclerView 实现，其他列表使用 LinearLayout / RelativeLayout / ScrollView 等类实现
- 数据的处理（包括网络访问和存储）与UI分离，在 NewsManager 类中实现
- 使用 [Volley](#) 库管理网络调用和缓存
- 离线访问新闻是通过磁盘缓存直接实现的
- 文字模式是通过禁止请求图片来实现的
- 使用 [百度语音合成服务](#) 实现语音播报功能
- 用本地文件存储各种设置信息，其中包括收藏的新闻和已读的新闻

- 使用 github 管理代码，使用 gradle 管理项目的依赖
- 由于没有使用太多第三方库，流畅性非常好

4. 总结与心得

写了五天，1894行代码。为了赶时间只做了足够得满分的功能。

最大的收获是，在整个过程中几乎没用什么框架和第三方库，很多小功能都是自己造的，这节省了大量时间。比如说，我问室友新闻分类的标签页怎么写，他说有“轮子”（第三方库），而我觉得学一个轮子太费时间了，就用 LinearLayout 和一堆 TextView 仿了一个，也就100行代码，效果还不错。再比如，室友告诉我搜索功能可以用 SearchView 来做，而那时候我表示我已经用一个图标、一个文本框和一个按钮造好了。

当然这么做是会有不少缺陷的。整体上不用框架，就会让代码显得比用框架稍微乱一些。即使在熟悉 MV* (MVC, MVP, MVVM, ...) 的那些理论思想的基础上，没有一个整体的规划或使用现成的框架，也很难写出真正做到数据和 UI 分离的程序。比如说事实上在我的应用中每个主要的类都有其他类的指针。

另外一个问题是单元测试。要进行全面的测试，就需要先保证代码结构清晰，然后每写一部分就测试一部分，以及要将所有测试自动化。这十分困难，而且通常情况下测试的代码量不会少于工程的代码量。在这个项目中，考虑到时间因素，我并没有进行任何自动化的单元测试，而只是在 Android 物理机上手动测试了几十种用户对 UI 可能的操作。要真正的测试一个应用程序的话，还要模拟不同的使用环境，来测试应用的稳定性，比如网络不稳定，比如手机CPU很慢，或内存很小等等，测试真的不简单。

对这个项目的看法有很多，只说重要的。在内容上，我希望这个项目（也包括之前的小作业）能跟课程更加配套，或者是课程能跟项目更加配套，这样能在课程中真正学到东西，而不是在查文档的过程中学东西。举一个很好的例子，清华的 [操作系统](#) 课，就做得相对比较好，可以说是我在清华上过的最好的课程之一。

在形式上，我希望这个项目能够有更加好的评价方式，以及更加充分的时间进行开发、讨论和交流，比如在课上进行展示，然后大家讨论并打分，而不是“实现了XXX，能得到XXX分”。之前说的测试这回事就是，我觉得把测试写充分了，至少有30~40的工作量的。

总的来说就这个样子。MIT 也有 Android 开发的课程，[6.S198](#)，他们的课程设置就合理很多，可以拿来比较和参考。