

MyNewsApp 实验文档

Author: 张诗颖 (经12-计18, 2021011056, shiyang-21@mails.tsinghua.edu.cn)

本实验文档仅供程设小学期 Java + Android 部分大作业使用

1 代码结构

1.1 概述

Java类源文件方面, MyNewsApp 主要由两个 Activity 控制类, 三个 Fragment 展示类以及一个 CollectionFragment 类族共同组成, 对应了切换 app 页面、详细新闻界面、收藏新闻界面、历史记录界面、搜索界面和按照分类展示不同新闻缩略信息等主要功能。其余辅助类源文件包括封装了本地存储功能的 Storage 类、app应用的全局类 GlobalApplication 和新闻类 News。

res 资源文件方面, MyNewsApp 主要包括了 layout 布局文件, drawable 图案和颜色控制文件, font 字体文件, menu 菜单导航文件和 values 字符串及主题色设置文件。

接下来将依次简要介绍各部分的功能及结构。

1.2 新闻类 News

News 新闻类主要以存储新闻信息为主, 也包括了简单的功能接口, 具体如下:

1. 信息记录变量

newsID 用来记录每一条新闻的唯一标识。

主要信息记录变量包括了 title, category, origin, time, content, imageUrls, videoUrls, imageExist, videoExist, imageCount 等从给定的 url 可以直接爬取的信息, 以及 fav, like, read, readDetail 等记录用户阅读和点赞新闻状态的信息。后者在新闻类对象 news 创建时默认为 false。

同时, News 类内还有一个 ArrayList<String> 变量 images 专门存储所有浏览过新闻图片, 采用字符串方式存储。该部分详见 Storage 类内接口函数 stringToBitmap 和 bitmapToString 的介绍。

2. 构造函数: News(JSONObject news)

采用 JSONObject 对象构造新闻类对象。其中又通过 getImage 和 getVideo 两个成员函数负责将原始文本的图片视频连接转换为字符串数组。

3. 图片/视频url处理

主要实现了 **图片去重**、**不同格式图片url提取** 等功能。后者包括但不限于以 , 或 [] 分条的url格式、空url 的删除等。

1.3 Storage类与GlobalApplication类

1. Storage类

主要负责本地存储功能，是 SharedPreferences 及 SharedPreferences.Editor 功能的一个封装内，其中所有的函数均为 public static（公有静态），方便写入存储和读取本地数据时调用。

下面简要介绍主要接口函数：

```
1 //将键值对写入存储
2 void write(Context context, String key, String value)
3
4 //增减收藏夹与历史记录的新闻newsID信息
5 void addFav(Context context, String newsID)
6 void removeNewsFromFav(Context context, String newsID)
7 void addHis(Context context, String newsID)
8
9 //findValue及其衍生函数：提取页码page信息、收藏/历史记录的新闻newsID信息、新闻News对象
10 String findValue(Context context, String key)
11 ArrayList<String> findListValue(Context context, String key)
12 News findNewsValue(Context context, String newsID)
13 int findPageValue(Context context, int category)
14
15 //String与Bitmap图片格式转换函数
16 Bitmap stringToBitmap(String image)
17 String bitmapToString(Bitmap bitmap)
```

同时，提供五种键值对检索方法：

```
1 // [key, value]
2 ["currentPage", String[] currentPages] 记录各个分类新闻浏览的当前页码
3 ["today" String today] 日期信息
4 ["fav", String[] newsIDs] 收藏新闻的newsID信息, join by ','
5 ["his", String[] newsIDs] 历史记录的新闻ID信息, join by ','
6 [String newsID, News news] newsID对应的新闻类对象信息, parsed by gson
```

2. GlobalApplication类

该类为 Storage 类的辅助类，主要替代了本来只能在 Activity 的 Context 下调用的 getApplicationContext() 函数，提供全局静态接口 getAppContext()

1.4 MainActivity界面控制类

其余主要类的功能结构如下所示：

```
1 |--MainActivity: 主要功能为界面导航和切换
2 |   |--Collection类族: 详见1.5部分讲解
3 |     |--CollectionFragment
4 |     |--CollectionFragmentAdapter
5 |     |--CategoryFragment
6 |     |--CategoryFragmentAdapter
7 |   |--Discover类族: 实现了搜索界面及搜索新闻结果的展示
8 |     |--DiscoverFragment（个性搜索结果界面）
9 |     |--DiscoverFragmentAdapter
10 |     |--SearchFragment（搜索界面）
```

```
11 | |--Favorites类族：实现了收藏新闻界面
12 | |--FavoritesFragment
13 | |--FavoritesFragmentAdapter
14 | |--History类族：实现了新闻历史记录界面
15 | |--HistoryFragment
16 | |--HistoryFragmentAdpater
17 |--DetailNewsActivity：详细新闻界面，实现了新闻主体内容的呈现、点赞与收藏功能，是上述各
   | 界面Adapter的通讯对象
```

主界面由以下组件构成，与上述主要类相对应：

1. Discover：呈现按类别分类的随机新闻列表（对应 Collection 类族），搜索结果界面（对应 Discover 类族）
2. Favorites：呈现收藏新闻列表
3. History：呈现历史新闻列表
4. SearchButton：点击后进入搜索界面，可按照关键词、时间、分类进行个性化新闻搜索

1.5 Collection 分类新闻展示

此部分对应 Discover 界面随机或按照类别分类的新闻列表展示。具体实现为ViewPager2 + TabLayout + Fragment。此部分代码应于 CollectionFragment 和 CollectionFragmentAdapter 类。

1.6 Fragment + FragmentAdapter：缩略新闻信息列表

app中“新闻缩略页面+点击查看详细界面”的功能主要依赖RecyclerView完成，而 Fragment + FragmentAdapter 是具体实现的基本结构。如上所示，Collection，Discover，Favorites，History 四个主要部件都是以这种方式实现的。



如图所示便是MainActivity所控制的主界面，Discover部分由可以左右滑动浏览不同类别新闻的ViewPager2组成，下段有Favorites和Hitsory文件夹，以及搜索悬浮图标共同构成。

2 具体实现

1. 新闻缩略列表：RecyclerView

【布局概述】整体app风格偏简约但重视控件的立体感和空间感，颜色主题为莫兰迪淡色系，动画强调与用户的实时交互性。RecyclerView以卡片形式展现新闻缩略信息列表，淡蓝色的边框与底部的导航栏呼应，圆角矩形和阴影实现新闻卡片的立体感，用户操作较为流畅丝滑，加载时使用动态动画减少延时感。

【图片加载】新闻缩略列表的图片加载逻辑为：若仅有一张图片则显示一张图片，若有两张及以上的图片则显示两张图片，若有视频则在图片之后显示一个视频，均尽可能完整显示更多的图片/视频资源。这里的主要技术难点在于：① 超过两张图片时需要判断所有请求图片链接的线程全部进行完再进行统一布局；② 若遇到有图片链接访问失败尽量寻找链接能访问成功的图片进行替代。在这个难点上，继承自Handler的自定义类MyHandler发挥了处理不同时段传入信息的重大作用。多种分类讨论使得不同情况都能细致地处理好图片呈现。

特朗普:扎克伯格上周在白宫拍我马屁 这个怪胎拍我一整晚

国内

中华网 2022-09-05 14:49:56



英国第三位女首相特拉斯是谁？将接手一个“危机重重”的英国

国内

网易新闻 2022-09-05 19:51:22



【视频加载】视频加载的逻辑是：缩略页面图片和视频均会显示，而详情页面如果有视频则图片隐藏。视频自动播放，可以暂停、快退快进。播放完一遍之后停止播放（不循环）。此处主要采用了VideoView和MediaController两个类来实现。

【细节简释】字体为另导入的等线和等线加粗（因为android自带的字体与中文的适配性不高）；行间距与各种边距精细化动态调整，适应各种篇幅的文字与不同尺寸的屏幕；可点击空间在点击时会有淡彩色涟漪，提高与用户的实时交互性；右上角×标志点击后可以从列表中删除新闻；不同界面的RecyclerView采用不同颜色予以区别（随机浏览新闻的淡蓝色，收藏夹里的淡粉色以及历史记录的灰色）；阅读后/本地缓存完成的新闻边框变灰

2. 新闻详情页：ScrollView

【布局概述】：所有图片以Horizontal ScrollView的方式呈现在页面最上方，图片全部缩放到统一高度，用户可以左右滑动浏览所有图片。图片下显示标题，分类，出处及时间。再往下为以Vertical ScrollView方式呈现的新闻正文，文末呈现有点赞和收藏的动画图标。



3. 本地存储：SharedPreferences

【存储方式】：在对比了file方法和database方法后，鉴于代码见解性原则，本app在SharedPreferences的基础上封装类Storage完成本地存储功能。具体功能表述为：所有点击查看过的新闻内容及图片都可以本地离线查看。这里的主要技术难点在于图片的存储和读取。在这个问题上，采用String方法存储图片，具体为①通过URL链接获取图片信息并转为Bitmap形式，②采用Storage实现的bitmapToString接口将图片转换为字符串保存，具体为调用Gson类的相关接口即可简单操作。读取图片时，将字符串用Storage实现的stringToBitmap接口转换为Bitmap便可以直接呈现。

【便捷性】：由于SharedPreferences本身的“键值对”存储方式与News类型、ArrayList数组类型等相比需要进一步的编码与解码，因此Storage提供解析和编码各种类型数据的统一接口，方便外部类直接存储各种类型的数据：

```
1 //将键值对写入存储
2 void write(Context context, String key, String value)
3
4 //增减收藏夹与历史记录的新闻newsID信息
5 void addFav(Context context, String newsID)
```



```

6 void removeNewsFromFav(Context context, String newsID)
7 void addHis(Context context, String newsID)
8
9 //findValue及其衍生函数：提取页码page信息、收藏/历史记录的新闻newsID信息、新闻News对象
10 String findValue(Context context, String key)
11 ArrayList<String> findListValue(Context context, String key)
12 News findNewsValue(Context context, String newsID)
13 int findPageValue(Context context, int category)
14
15 //String与Bitmap图片格式转换函数
16 Bitmap stringToBitmap(String image)
17 String bitmapToString(Bitmap bitmap)

```

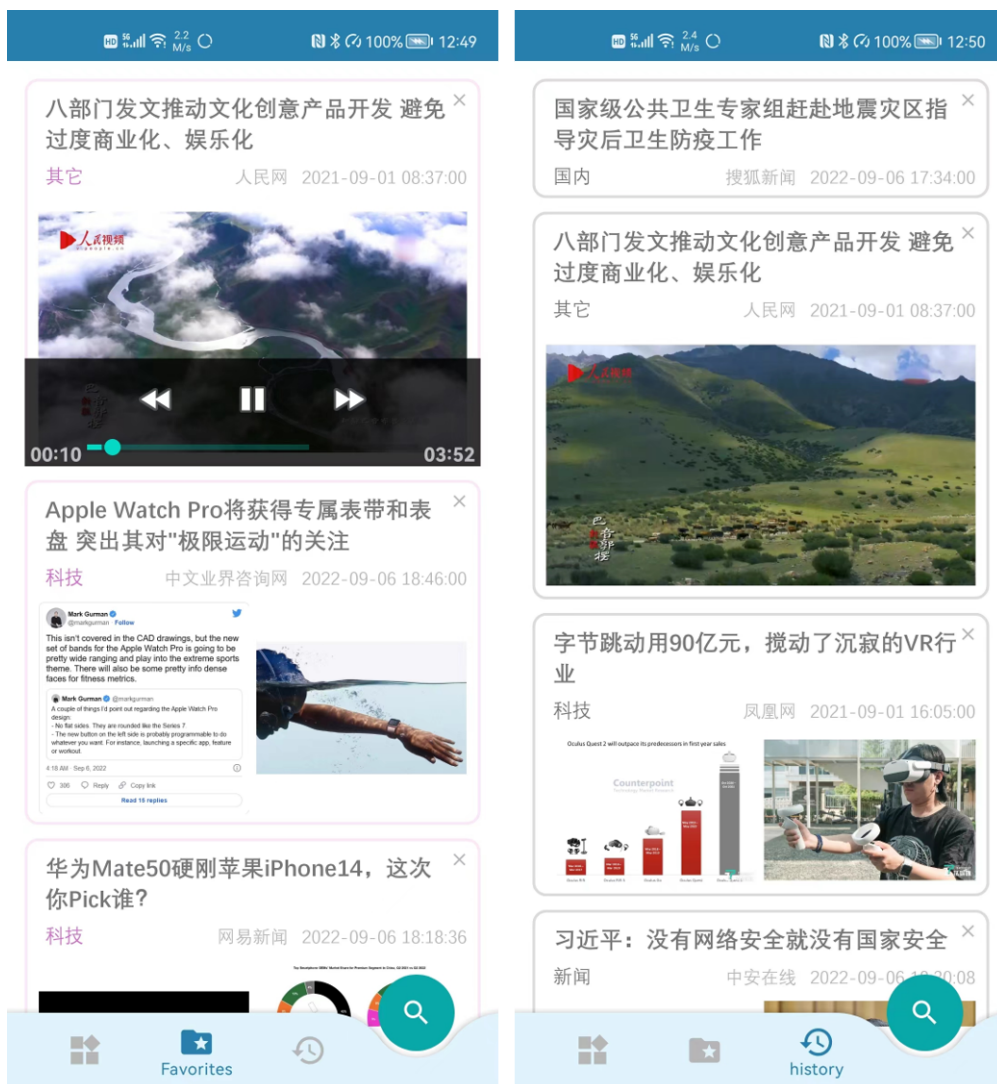
提供五种键值对检索方法：

```

1 // [key, value]
2 ["currentPage", String[] currentPages] 记录各个分类新闻浏览的当前页码
3 ["today" String today] 日期信息
4 ["fav", String[] newsIDs] 收藏新闻的newsID信息, join by ','
5 ["his", String[] newsIDs] 历史记录newsID信息, join by ','
6 [String newsID, News news] newsID对应的新闻类对象信息, parsed by gson

```

【收藏列表与历史列表查看】采用 `setStackFromEnd(true)` 和 `setReverseLayout(true)` 方法使得最新的收藏/历史记录出现在整个立标的最上方，**新闻排序正确**。



4. 网络请求与多线程：HttpURLConnection, Thread, Handler

【解析数据】：采用android自带的JSON包进行数据层层提取

【网络请求流程】：由于android要求涉及网络请求的部分必须另开线程进行（即不能占用主线程），因此网络请求的模型主要为**Thread发出强求 + JSON解析数据 + 告知Handler处理message**，从而拆到最底层的JSONObject交给 News 的构造函数进行 News 对象的构建。其中读取数据采用了反复利用的 readFromStream 接口函数，通过 buffer 的方式一部分一部分地将所有数据读全。

【网络传输效率】：为了保证延迟较低，默认每次请求新闻在20-40条，经测试这样的请求强度能够比较能为人接受地不太敢到延迟。同时，子线程的运用也使得各个网络请求互不影响，因而可以利用“非操作时间”持续加载图片并缓存到本地。

5. 下拉刷新与上拉加载：SwipeRefreshLayout + OnScrollListener

【下拉刷新】：刷新采用了 SwipeRefreshLayout 实现，其中刷新加载过程中用动画表示，三种颜色的缓冲图案渐变（淡粉色、深粉色、浅紫色）的配色业余延缓动画和莫兰迪色系和谐一致。

【上拉加载】：加载过程中会弹出 Toast 对话框（写有：“Discovering more news...”）表示加载正在进行。加载后整体画面保持平稳，依旧可以浏览之前加载的新闻，点赞收藏即已读变灰效果也都保留。

上拉加载的本质实现是将url请求头中加入 &page=x 操作，其中 x 为当前当前页码。随机新闻部分、各个分类部分的新闻页码均记录在 storage 中，因此每次 ++page 即可以上拉或者刷新得到的新闻都是尚未浏览过的。



6. Activity/Fragment之间信息传递：ActivityResultLauncher

【点击进入新闻详情页】：最开始采用了最平常的 Intent 方法，但是由于该方法无法获取信息接受者对于信息发送者的反馈回应（即无法将点赞/收藏信息及时return back给新闻缩略列表），因此后续改为ActivityResultLauncher方法，具体如下图所示：

```
1 ActivityResultLauncher<String> launcher = registerForActivityResult(new
  CategoryFragment.ResultContract(), new ActivityResultCallback<String>() {
2     @Override
3     public void onActivityResult(String result) {
4         ...//parse result
5     });
6
7 //重写ResultContract类
8 class ResultContract extends ActivityResultContract<String, String> {
9     @NonNull
```



```

10     @Override
11     public Intent createIntent(@NonNull Context context, String input) {
12         Intent intent = new Intent(getContext(), DetailNewsActivity.class);
13         intent.putExtra("newsID", input);
14         return intent;
15     }
16
17     @Override
18     public String parseResult(int resultCode, @Nullable Intent intent) {
19         return intent.getStringExtra("feedback");
20     }
21 }

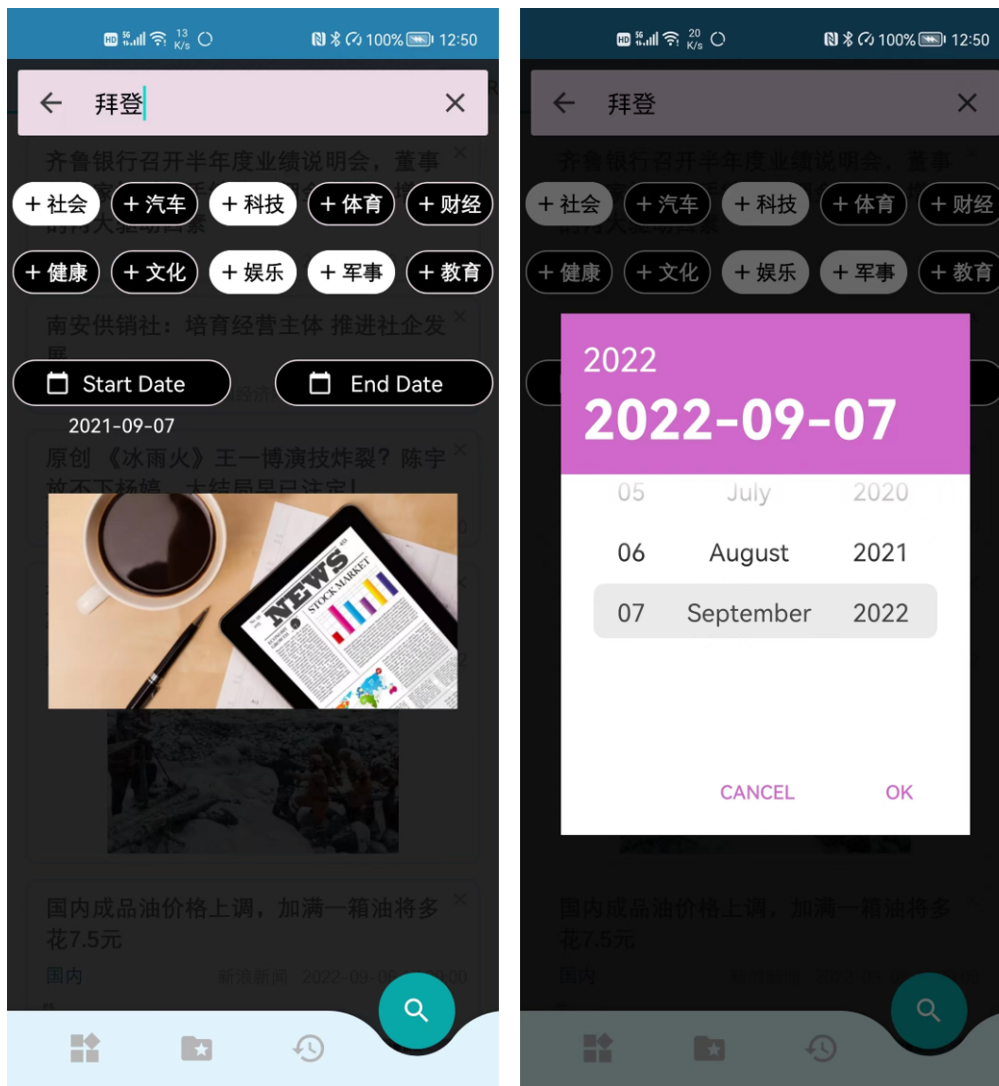
```

【点赞、收藏以及已读新闻变灰等需要即使同步的用户操作】这种方法最大的好处就是可以接收点赞、收藏以及已读新闻变灰等用户操作信息，反馈给新闻缩略列表。实际操作而言，无论从随机新闻漫步主页、收藏页还是历史页，无论时间间隔（退出后立刻重新点击进入或者杀掉程序进程之后重新打开程序进入）点击同一条新闻，前一次的操作记录都会即时保留保存并反馈给用户。

例如：在收藏页点开新闻取消收藏，右滑退出后即可发现新闻从收藏页中移除；在历史记录页点开新闻标识收藏，切换到收藏页即可发现新收藏的新闻出现在收藏夹最上方。

7. 搜索：多分类选项搜索

【多分类选项搜索】：采取开不同线程访问不同类别新闻的方式并汇总呈现，一次加载（暂未下拉刷新或上滑加载）呈现20-40条新闻（无论选择分类有几种均保持在一个合理的加载范围）。在打开搜索模式后，上拉下滑依然可以查看更多符合相关要求的新闻。在重新返回主界面后搜索模式自动关闭。



8. 布局与组件Reference: Material Design 3 + github开源项目

【Material Design / google / android】: ConstraintLayout, ViewPager2, TabLayout, SwiperefreshLayout, Gson

【github开源项目】:

com.github.mancj:MaterialSearchBar:0.8.5 [mancj/MaterialSearchBar](https://github.com/mancj/MaterialSearchBar): 搜索页面的顶侧搜索框

com.github.ybq:Android-SpinKit:1.4.0 [ybq/Android-SpinKit](https://github.com/ybq/Android-SpinKit): 页面转换时加载的动画

com.sackcentury:shinebutton:1.0.0 [ChadCSong/ShineButton](https://github.com/ChadCSong/ShineButton): 点赞与收藏的icon极其点击特效

com.github.niwattep:material-slide-date-picker [niwattep/material-slide-date-picker](https://github.com/niwattep/material-slide-date-picker): 搜索页面的日期选择dialog



9. 关于离线状态下的补充说明

以下若未经特别说明“离线”以设置手机为飞行模式为例

正在阅读的Discover新闻列表会实时缓存新闻信息（但仅缓存除图片视频意外的其他信息）

- Discover：左右滑动不同分类的新闻列表，已加载出来的新闻列表仍可查看（包括图片）；未加载任何新闻列表的页面显示“Network Failure”，无法搜索到结果；仅点击阅读过的新闻（边框为灰色）可以正常点击进入离线浏览，未阅读过的新闻点击进入详情页后无法浏览图片信息（但文字部分内容正常）；刷新后界面提示“Network Failure”，无法搜索到结果。
- Favorites：列表即点击进入的详细页面均完整（包括图片，不包括视频），可以进行点赞、收藏、取消收藏操作。
- History：列表即点击进入的详细页面均完整（包括图片，不包括视频），可以进行点赞、收藏、取消收藏操作。
- Search：搜索页面正常显示，搜索结果页面提示“Network Failure”，无法搜索到结果

3 总结与心得

总结

其实小学期没有之前听学长学姐说的那么恐怖吧。我个人认为小学期给我的收获还是相当大的，我也确实很喜欢这种开发工程。

我个人认为小学期最大的难点应该是怎么迅速上手——怎么知道有这个控件可以用，怎么知道这个功能应该怎么去实现.....因此对于“善用搜索引擎”、“听课”、“问学长学姐”，“将C++学到的面向对象思想和开发模式迁移到其他语言和项目的开发中”有了更高的要求 and 锻炼。在最开始的摸索期一定是所谓清华式“fly bitch”、深感绝望的，但是也会发现掌握到门道之后飞速的进步，亦有实现了一个一个功能之后的成就感和小欢喜。做工程应该是一个“正反馈”很迅速的过程——那种debug了好久终于调试完美的感觉真的很好，and 做出一个很生活化的游戏和app也是一件非常有成就感和激动人心的事情。

在小学期的过程中我真的学到了很多。比如现在更会看报错信息了（哈哈哈哈），对于搜索引擎愈加熟练了，也能在上手新事物的过程中感到不同知识的迁移和“换汤不换药”的本质思想。在请教学长的过程中，我也对于计算机存储系统、进程和线程的底层原理有了更本质的东西了解。在与同学的交流和学长答疑的过程中，我也明白了不同人/经验更丰富的人是怎样解决某一特定问题/实现某一特定功能的。这种思维上的碰撞和启发对我来说受益匪浅。

致谢

郭昊（计97）

郭老师在我Qt和Java+Android小学期阶段（其实还包括上学期的OOP和可能之后的一切计算机专业课）提供了极其巨大的帮助。他的形象讲解和耐心解答使得我不仅更本质的理解了很多更底层的逻辑实现方式，还让我这四周的小学期体验极佳（做项目的极强成就感）。在他的帮助下我两个大作业的基本功能一周内即完成，有更多的时间完善或学习要求之外的东西。暴风式感谢.jpg/

Google/StackOverflow/Developer.Android.com/建站

我几乎花了半数以上的实践在搜索引擎上学习新知识和debug。本次小学期深刻地感受到了编程社区的强大。感谢！（犹记我有一个上午梯子挂掉了没法上google，那个上午我基本上毫无生产力，下午修好了梯子之后生产力upup...）

李国良老师、许斌老师、助教团队

从内容详细的PPT中受益匪浅。