

Android 移动应用开发期中项目报告

王者荣耀英雄大全

组号：15

小组成员：

梁颖霖 16340132 806179953@qq.com 806179953

梁育诚 16340133

梁庭 16340131

余崇斌 13331217

目录	2
目录	
1 概述	3
2 背景	3
3 项目说明	4
3.1 功能模块.....	4
3.2 逻辑流程.....	6
3.3 技术说明.....	7
3.4 代码架构.....	10
4 应用效果	13
5 开发过程中遇到的问题及解决办法	16
6 思考及感想	18
7 小组分工	19
8 参考资料	20

1. 概述

本项目是一个王者荣耀英雄大全手机端应用，提供给用户查询、添加、删除、修改、对比英雄等功能。主要有三个页面：主页、详情页面和英雄对比页面。其中主页和详情页面都运用了TabLayout来布局，因此内容是非常丰富的。在基本页面的基础上，还增加了侧栏，后端有强大的数据库作为数据支撑，使得整个应用非常的立体和多样化。

从需求来说，这个应用能够满足绝大多数手机玩家的需求。王者荣耀作为一款火热的手游，其用户肯定是希望能在手机端有一个应用能够查看英雄资料。因此在整个应用设计的过程中，虽然我们参考的是王者荣耀网页端的布局，但是我们从布局上和逻辑处理上都充分考虑了手机端的使用便捷性，让这个应用变得比较友好。比较独特的功能在于支持英雄属性的对比，我认为这对于游戏玩家来说是一个非常重要的功能。

以上就是我们对这个项目的概述了。

2. 背景

1. 实用性：如上面所说，我们在整个开发过程中都是将用户需求放在第一位的，用户怎么用得方便、怎么用得舒服，我们就怎么设计。我相信这款应用是非常满足用户的需求的。
2. 创新性：同款的应用已经有，是官方出品。本次项目是期中指定项目，因此这点不再分析。
3. 用户体验：页面上使用了TabLayout和侧栏，支持用户许多滑动操作。UI设计非常清晰，用户可以很直观地获取到信息。页面之间通讯已经优化，因此应用在页面跳转时都非常稳定，运行流畅。
4. 本项目的开发用到的都是比较基础的功能，因此是兼容大部分的版本与平台的，后续优化可能会对不同版本与平台详细讨论。
5. 本应用仅仅只是一个数据增删改查与显示的应用，没用用户功能，也不涉及到读取手机信息的部分。后续拓展可以考虑实现用户功能，绑定用户信息。目前版本不涉及用户隐私，是安全的。

3. 项目说明

3.1 通过功能模块图，说明系统功能需求。

王者荣耀英雄大全的主要功能分别有：

1. 通过英雄名字获取对应英雄的具体信息
2. 通过物品名字获取对应物品的具体信息
3. 在英雄具体信息的显示页中可以收藏或者取消已经收藏的英雄
4. 在收藏以及主页中能够通过英雄名字搜索对应英雄
5. 在物品界面能够通过物品名字搜索对应物品
6. 从侧栏能够进入收藏，主页，物品界面
7. 能够比较两个英雄的四个属性强弱的功能
8. 注销功能

以上的功能可以通过图1：功能模块图看出。

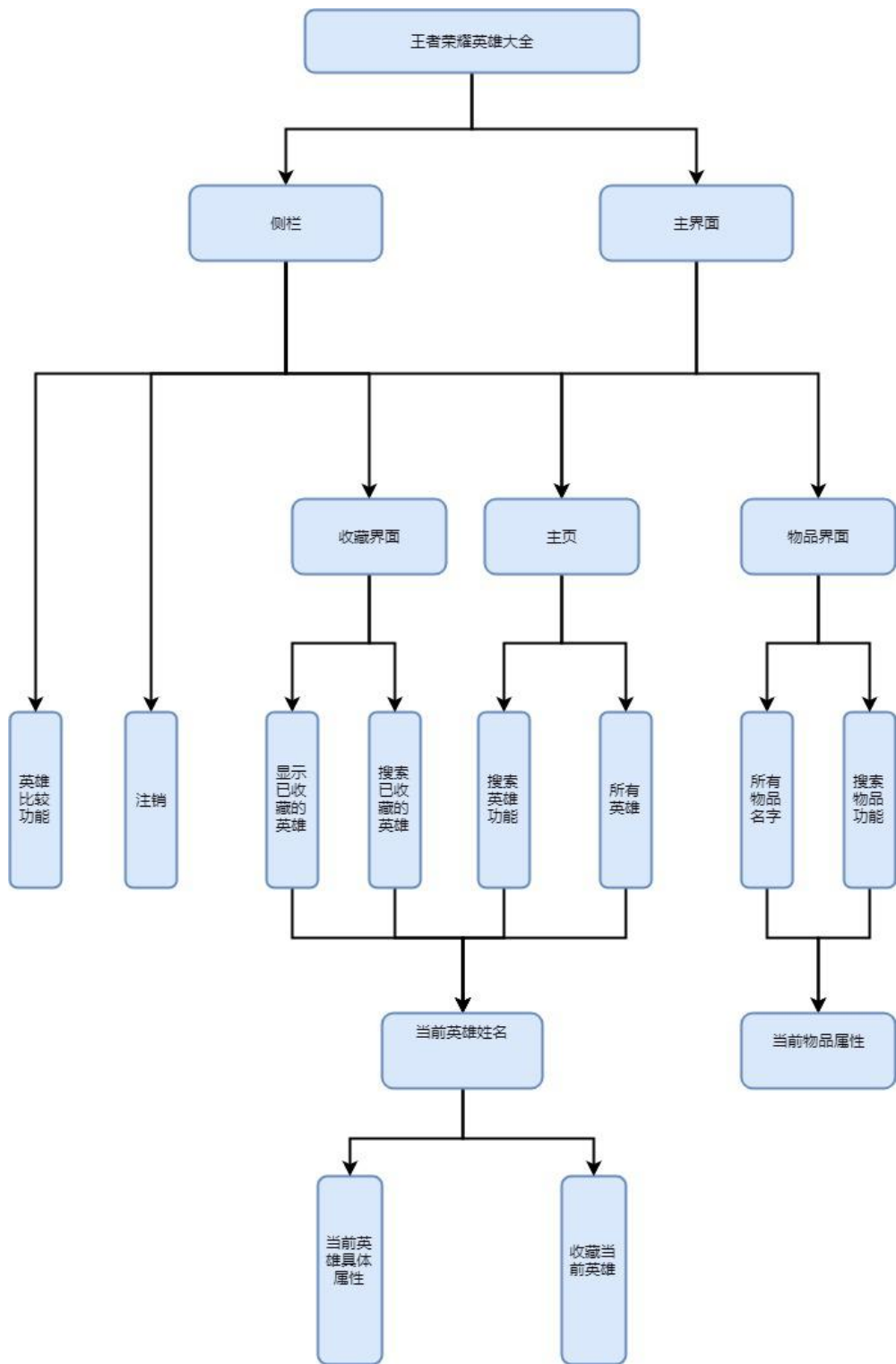


图1：功能模块图

3.2 逻辑流程

以下是我们应用的逻辑流程图，展示用户使用我们应用时候的操作逻辑。其中图2为总应用的逻辑流程，图3为比较功能的逻辑流程。

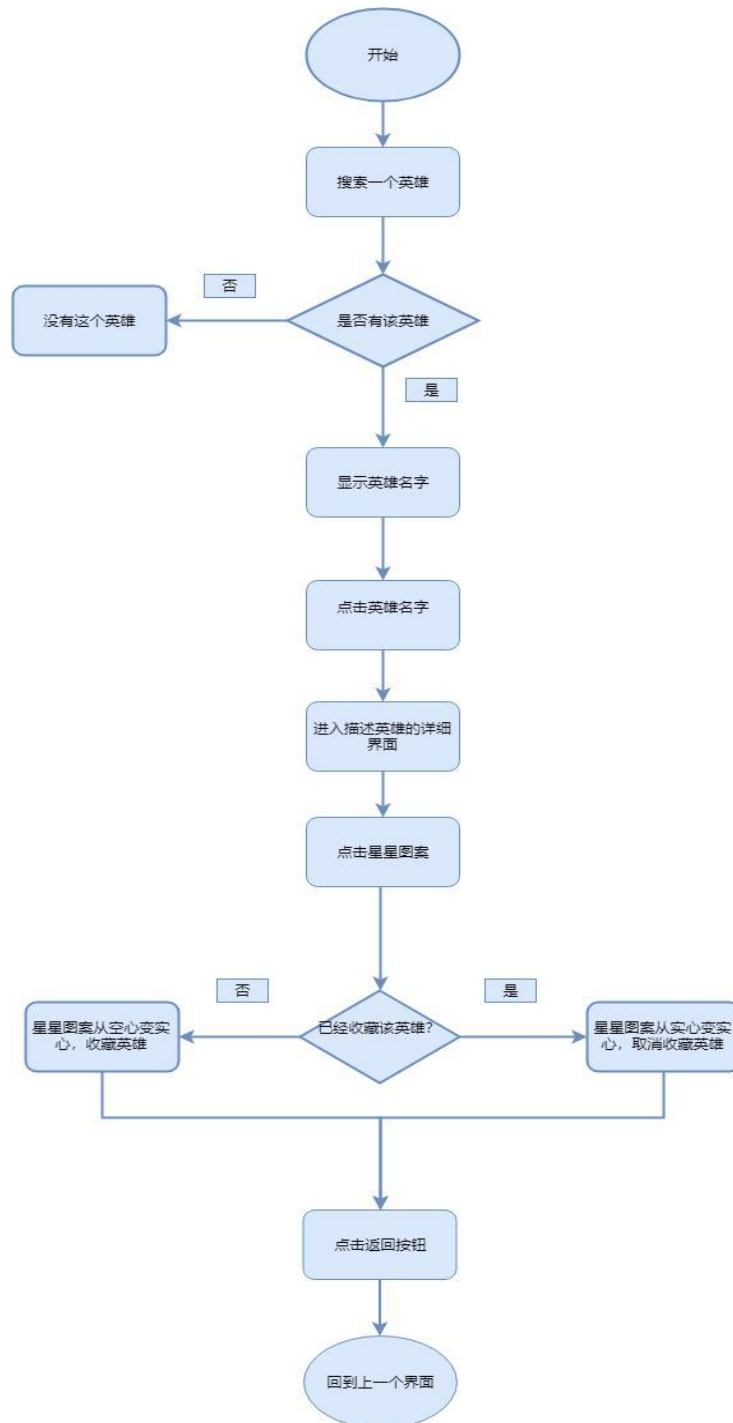


图2：总逻辑流程图

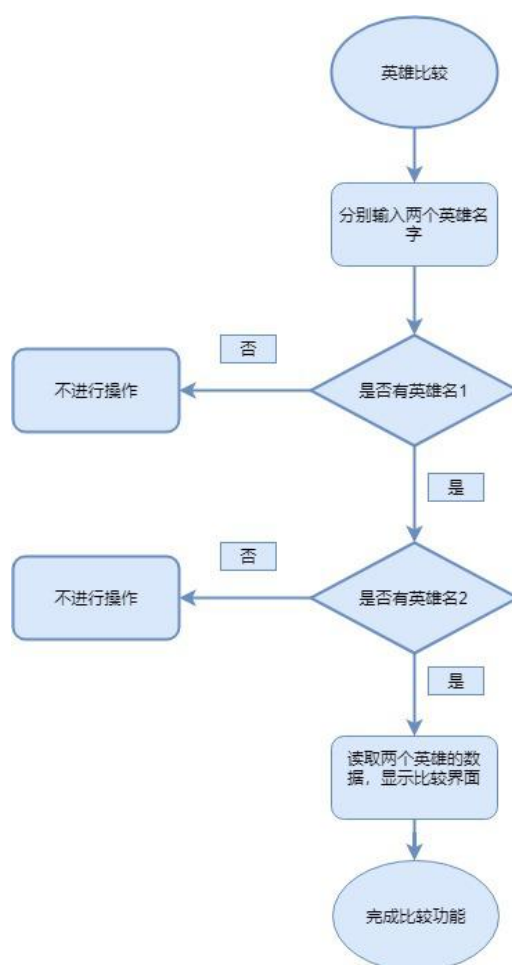


图3：比较功能逻辑流程图

3.3 技术说明

1. 主页部分：

功能：主页面和侧栏。大致功能是在侧栏能跳到任意一个页面（或子页面），主页面的3个tab自由切换，而且显示正确，搜索筛选功能正确。

侧栏

侧栏方面，我使用materialdrawer，比较顺利(不考虑版本问题的话-.-)的创建出来。先是在页面加个toolbar，然后依托这个toolbar，增加这个抽屉就好了

搜索筛选功能

因为可以做的功能不多，采取的是更改其中一个的时候，同时考虑其他条件进行更新，最后完成功能（三个子页面都有些小的不同，只展示主页面）

当然，首先是增加布局，因为还要对每个spinner的原属进行操作，所以就不在

xml里面添加数组，而是在Java里面进行操作。

子页面

子页面的话，采用了**tab+pager**的方式进行构建。让tab作为显示上的标签，pager作为容器承载子页面。然后使用3个fragment来处理3个子页面的显示和逻辑。

首先就是在主页面上添加两个东西，作为依托的**tablayout**和**pager**

然后再里面使用3个子页面fragment添加到标签页，并把tab和pager绑定

2.详情页面部分：

我负责的是英雄详情页面的设计与逻辑实现。实现的功能是，对于一个指定的英雄，需要显示其英雄的基本信息、技能，以及相应的出装建议。大致包括了一个界面设计和一个逻辑设计，当然一个大的页面中含有许多小的部分。

页面布局结构：一个总的是detail.xml，下面包含有两个Fragment:fragment1.xml和fragment2.xml，然后还有一个显示装备属性的布局populayout.xml。

逻辑文件包含三个：Detail.java、Fragment1.java和Fragment2.java，它们分别控制着对应页面的逻辑业务。还有一个用于渲染页面的MyFragmentPagerAdapter.java。

其中用到了**CircularProgressBar**和**PopupWindow**这两个很特别的新的开源控件。

3.英雄比较部分：

英雄比较界面的关键步骤

首先，通过https://blog.csdn.net/slow_liao/article/details/20062119，将pk按钮设置成抖动状态。

然后通过https://blog.csdn.net/qq_27485935/article/details/63681745，将软键盘进行隐藏。

4. 数据库部分

我做了以下的五个主要操作，英雄装备数据的抓取、数据导入数据库、数据库链接入Android应用、Hero, HeroEquip, HeroSkill三个基类的构建、数据库提供增删查改的接口API。

关于利用Python爬虫抓取王者荣耀网站的英雄装备信息，我利用简单的分析html树来用正则表达式来匹配，爬取所需要的信息。**正则表达式的匹配**以及对相应的数据进行**格式化分割操作**，利用**文件流**写入读出数据。

利用爬取过来的英雄信息txt文件，我采用了Navicat Premium 12这个软件，可以方便将txt文件转化成SQLite数据库内容。具体操作为，要对txt文件的头部加上相对应的属性，然后它会根据你的属性以及分隔符来构建一条条的英雄资料条目。

最为困难的部分应该在将外部的SQLite文件导入到用户的安卓手机存储之中，这里我编写了一个**manager类来处理数据库文件的读入**，主要功能包括第一次打开应用将数据库写入用户的手机，而后面的打开会先判断数据库是否存在，若已经存在了，则直接获取该路径即可，不需要重复写入。实际上也是在操作SQLiteDatabase，利用其接口来判断以及获取已经存在手机的数据库。

3.4代码架构

通过图4.1, 4.2, 4.3, 4.4的代码结构图可以看出以下结构

adapter是作为相应的容器管理类:

```
MyFragmentPagerAdapter.java,  
MyRecyclerViewAdapter.java,  
MyViewHolder.java,  
TitleFragmentPagerAdapter.java
```

View里面的类是相关页面的显示管理, 和逻辑类:

```
Detail.java,  
CompareActivity.java,  
MainActivity.java,  
MyFragmentTICollection.java,  
MyFragmentTOGood.java,  
MyFragmentTOMain.java,  
Fragment1.java,  
Fragment2.java
```

其中主要的页面类是

```
Detail.java,  
CompareActivity.java,  
MainActivity.java。
```

而 MyFragmentTICollection.java , MyFragmentTOGood.java , MyFragmentTOMain.java则是在主页面的3个子页面的类, 分别处理三个子页面的显示和逻辑。

Fragment1.java, Fragment2.java则是处理Detail里面的相关逻辑与显示的类。

imageDownloader.java则是用来从网上下载图片用的。

AssetsDatabaseManager.java和Mydb.java是用来处理数据库的相关逻辑。

Equip.java，HeroDetail.java，HeroEquip.java，HeroSkill.java则是用到的实体类，分别表示装备，英雄详情，英雄装备，英雄技能类。

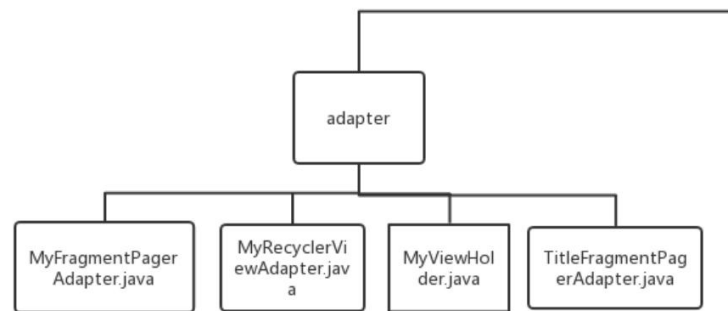


图 4.1: 王者荣耀英雄大区代码架构adapter部分

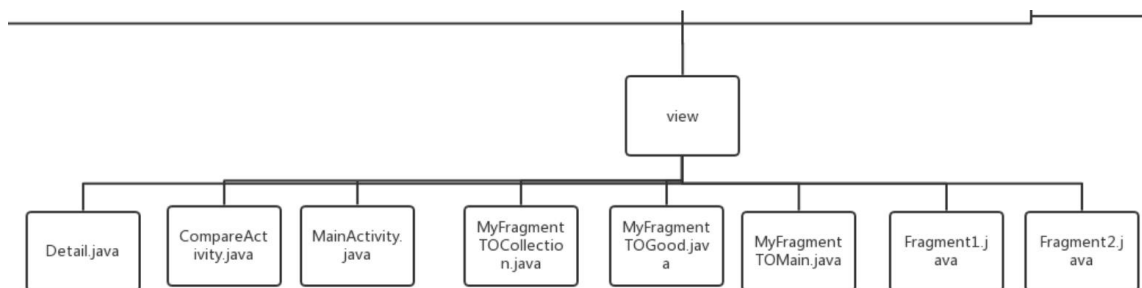


图 4.2: 王者荣耀英雄大区代码架构view部分

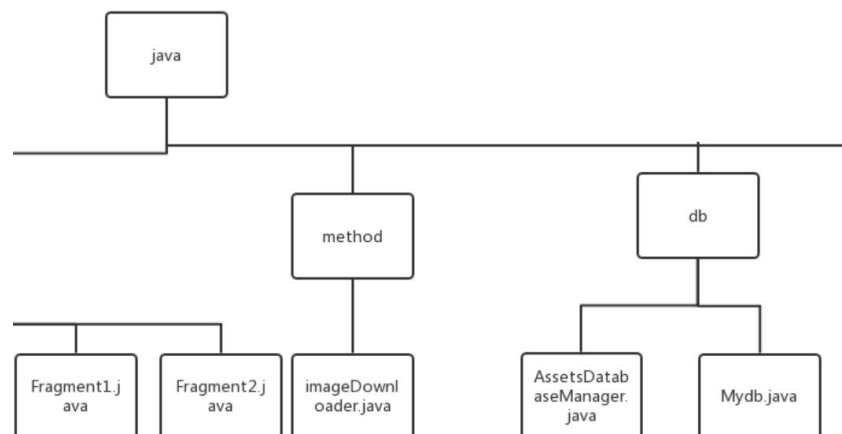


图 4.3: 王者荣耀英雄大区代码架构method与db部分

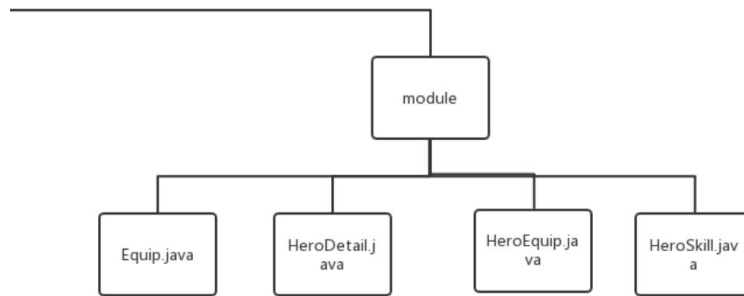


图 4.4: 王者荣耀英雄大区代码架构module部分

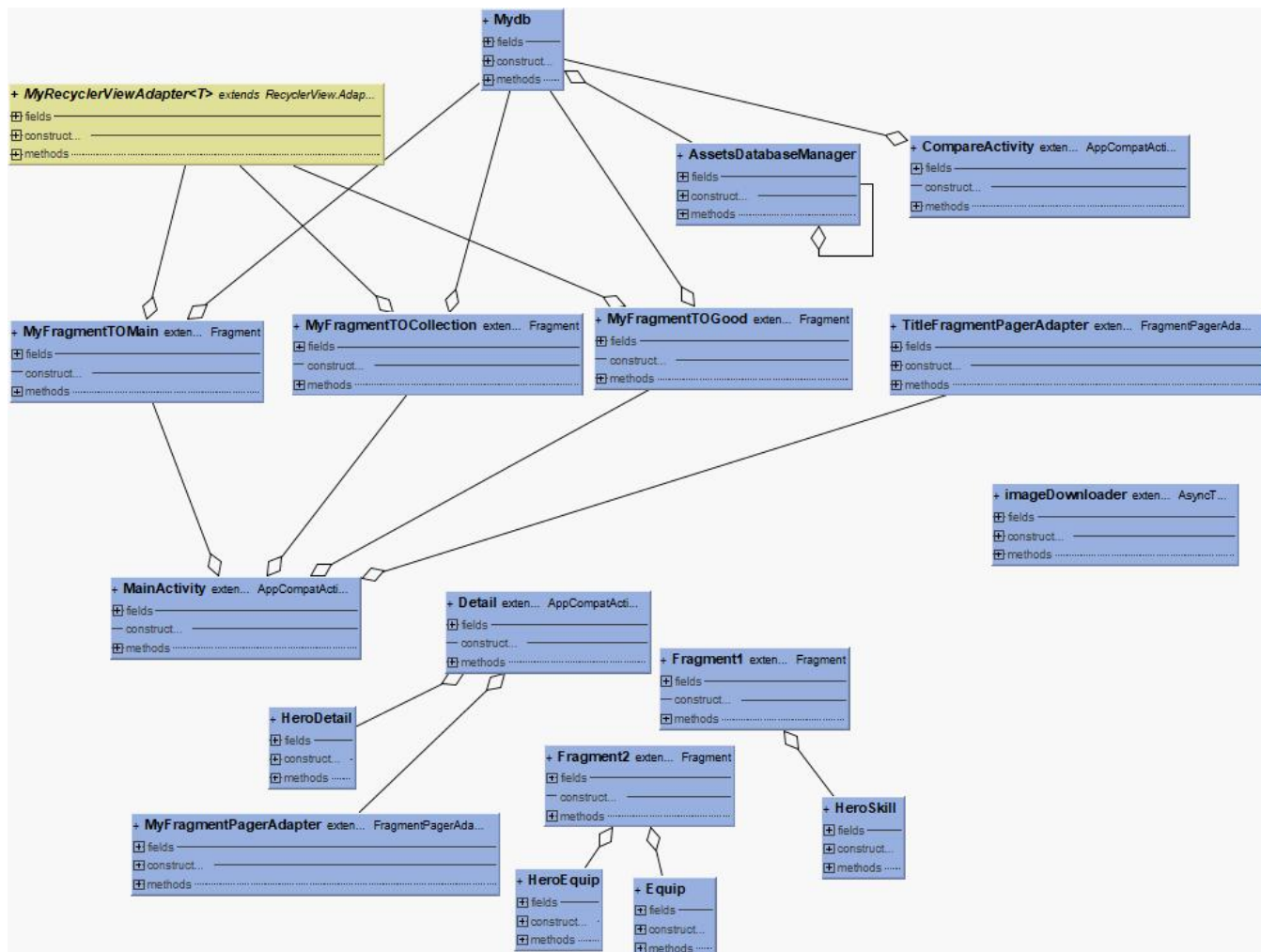
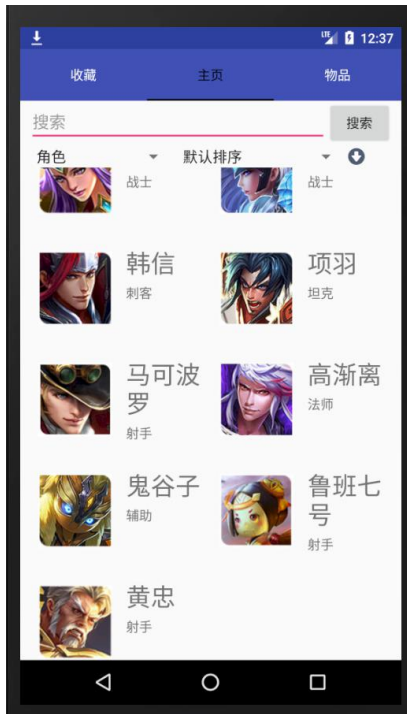
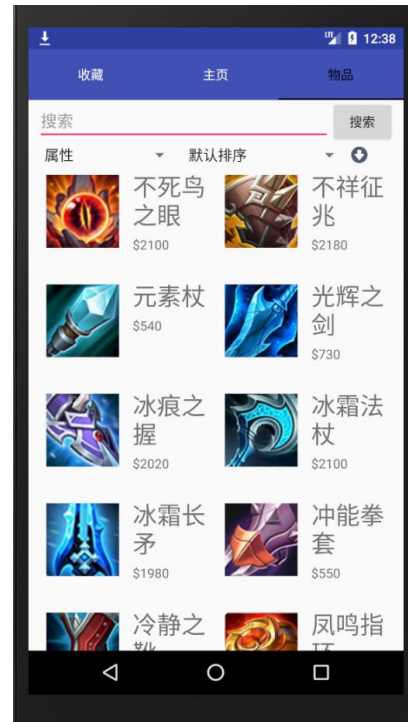


图 5: 实验十类关系图

4. 应用效果



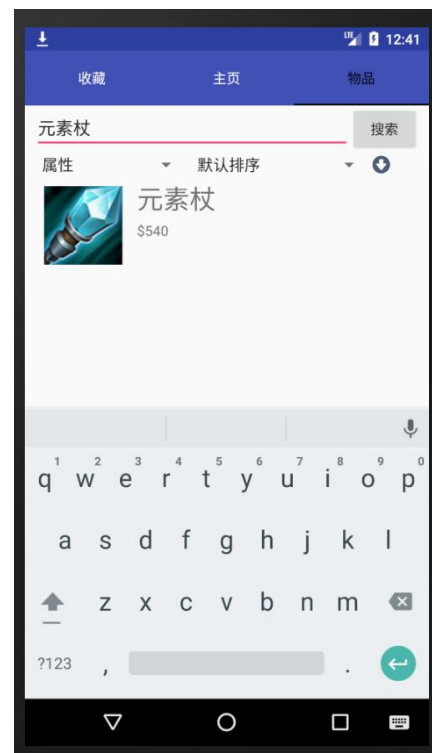
主页英雄列表



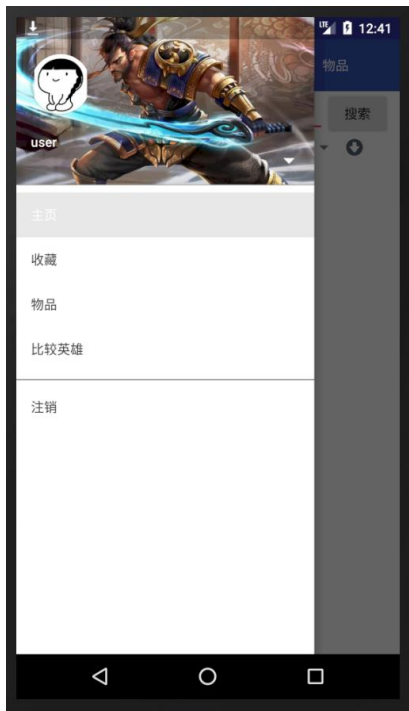
主页装备列表



搜索英雄【小乔】



搜索物品【元素杖】



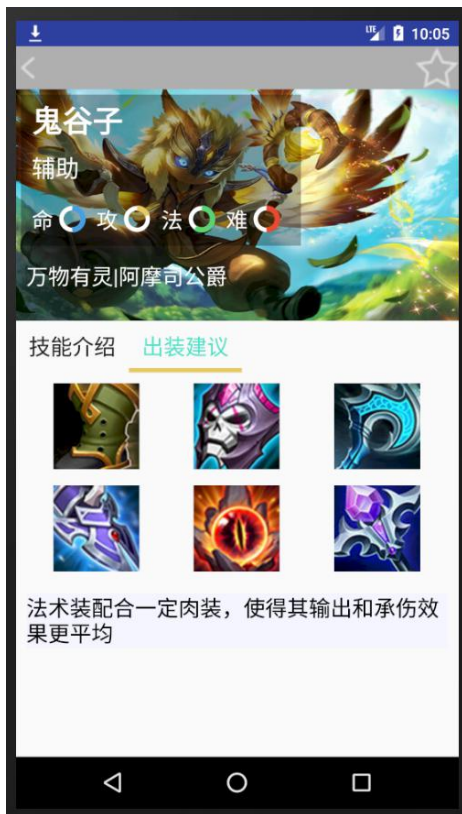
右滑拉出侧栏:



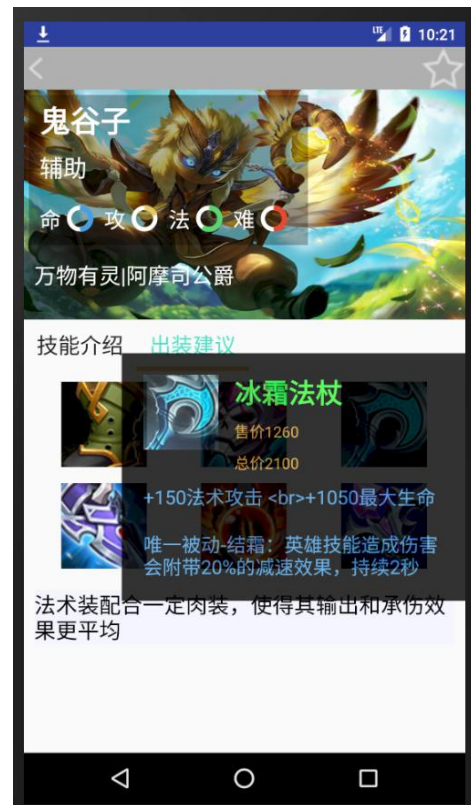
比较英雄【小乔】和【墨子】



点击【鬼谷子】进入该英雄详情界面,展示被动技能与大招



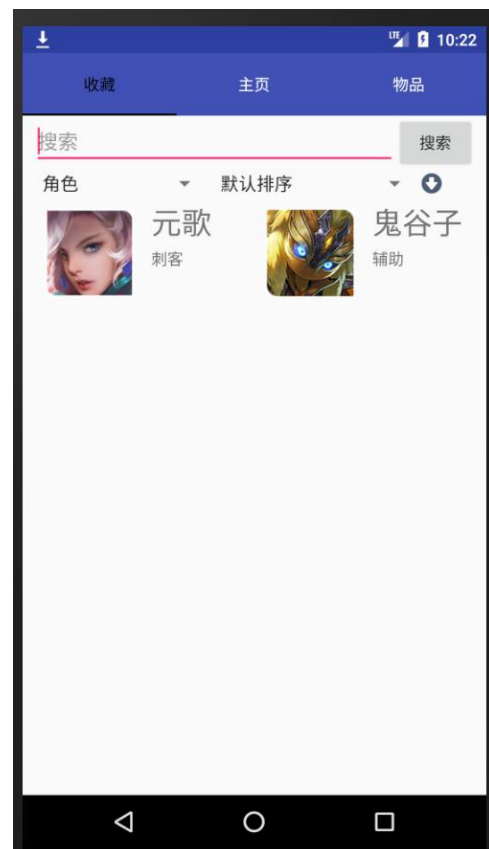
展示推荐装备页面



点击装备显示框



测试收藏功能



收藏成功, 显示在收藏夹



测试取消收藏



取消收藏后，收藏夹删除该英雄

5. 开发过程中遇到的问题及解决办法

问题一：如何使用两个新的控件：TabLayout和PopupWindow？

TabLayout是一个很常用的控件，使用这个控件的灵感来源于淘宝。对于手机用户来说，滑动动作要比点击动作方便，而且整体会显得更加流畅。因此上网搜索了很久后发现了这个控件。使用这个控件的灵活性很大，我们可以对每个Fragment再重新给出新的布局，这就很适合这次的项目了。适配器等代码网上基本都有教程，其实他们的工作原理与之前做过的ListView很类似，所以仔细看完教程后也很快写出来了。

问题二：PopupWindow的使用。

这是一个比较新的东西。刚开始我想使用的是Toast或者AlertDialog，但是发现这两种显示方式非常不友好。Toast更多的是通知与提示，而AlertDialog是比较严重的阻塞式提醒（一般是删除等操作的确认）。这里我需要的是展示信息，因此找到了PopupWindow这个控件，它的优点也很明显，可以自由设计显示框的布局！

问题三：页面间通讯。

详情页面一般是从主页点击某个英雄跳转过来的，可以参考之前作业那样将Hero封装好用Intent传过来。之前还一直在想是否要使用EventBus来处理复杂的页面通讯，但是这次我们使用了数据库，因此只需要传一个ID就可以完成功能了，这一块倒是没花很多时间。

问题四：materialdrawer的安装。

先是下载不下来，后来是同步不了。真是让人头大。后来一个一个的找issue才发现这个是因为版本更迭，要更新sdk和ide才可以，所以在把android studio升级到最新版本，把项目迁移到androidx之后，才顺利同步下来。然后的话终于顺利的打开了侧栏。

问题五：Tab子页面更新延迟。

我把所有的钩子都挂了个输出，就没有一个合适的。之后点到另一个最边边的页面（也就是让当前页面被挂起或者销毁这样子），才能够看见这个输出。后来只好在上一级的页面（也就是在主页面）在这个tab被选中的时候调用这个fragment的一个函数来进行更新（当然，有可能这个的一些内容还没初始化好而失败，这个加个trycatch就好了，失败的都是没必要成功的，自然就没所谓了。

问题六：py爬虫的信息错误。

一开始学习爬虫是比较困难的，我也是通过一些简单的例子开始理解。掌握基础的html树的爬取，期间遇到了十分多的问题。对于爬取下来的json格式，我还要进行处理，这里我参考了不少的博客，吸取前人的经验。我在爬取的时候，忘记判断该英雄是否重复，导致爬下来的内容繁琐，且英雄重复率很高。于是我在写入文件的时候先判断该文件是否已经存在，再进行写入，否则则直接丢弃该数据。

还有就是爬取内容的时候，字符的编码格式没有设置，导致爬下来的都是乱码无法识别具体内容。后来才知道需要在reponse上加上encoding。
`response.encoding = 'GBK'。`

问题七：数据库返回的信息错误

数据库由于表的属性较多，读取的时候又是通过columnindex来读取，容易出现数字写重复的情况。我是我们小组首先开始工作的，我完成数据库的接口后，必须先自己测试过才上传给他们使用。在测试的过程我就发现有些返回的信息不准确，出现不匹配的情况，这需要回到数据库函数来查看。除此之外，我还需要对数据库中的字符串进行处理，分割装备id的一系列字符串来返回一个int的数组，里面包含装备id。不然直接返回string，不利于详情页面通过id来访问我的装备信息获取接口。

6. 思考及感想

这次期中项目的我认为完成的还是比较满意的，能做到基本的功能，在美观上也还过得去。即用到了之前课程学习的知识，自己也通过网上浏览资料用了几个新的开源控件。同时，自己对于基本的布局、事件处理等代码的编写也变得熟练，可以把心思花在如何把应用做的好看上。

在小组合作上，有了上学期UWP的开发经历，对git的使用比较熟练了。在开发过程中遇到的协作问题也很少了，就算有了冲突也知道如何解决了，我觉得这是一个很大的收获。

这个项目最终也基本达到预期的效果，起码在数据准确性上和逻辑操作上是没有什么问题的。我负责的部分还有一些可以优化的地方，比如说在信息显示上可以利用更多样化的布局，同时可以添加一些动态元素使得页面更加丰富。

感觉还好。就是一些莫名其妙的bug，有点磨人。而特别是那个androidx的版本兼容问题，搞了一天，真是莫名其妙。后来的话就是存粹的写bug-debug环节了，感觉好多了。收获的话，感觉是这个android的结构其实跟html的还是挺像的，特别是这个fragment和frame对比起来之后的话，感觉就是有些差不多的样子。另外在和队员们的使用git进行协作的时候，感觉对git有了更多的了解。其实感觉这个分支是应该是可以用在一个项目里面多个人的。每个人开个分支就好了，这样的话，相互间的操作起来也会比较的方便，而不是需要频繁的在项目之间跳动。

在期中项目开始之前，老师还没有讲到有关于数据库的内容。而关于爬虫的抓取更是没有尝试过，我一边学习一边尝试的来一步步构造数据库，其中也是颇有挑战性，做出来后成就感当然也是满满。作为基类的书写，接口的提供必须要根据前端页面的需求，这次我就忽略了一个英雄的背景图url，后面才进行补救。下次一定要在前期就做好计划，不需要东拼西凑的写代码。

7. 小组分工

组员	学号	负责内容	贡献百分比 (%)
梁颖霖	16340132	英雄装备数据的抓取，数据库的构建，提供数据库增删查改接口，外部数据库导入手机储存。	25
梁育诚	16340133	英雄详情页面的布局设计与逻辑实现。与主页进行通讯，通过使用数据库接口实现较为好看的详情界面，向用户反馈数据。	25
梁庭	16340131	主页面，侧栏，搜索筛选	25
余崇斌	13331217	英雄比较页面，测试	25

表 1: 小组分工

8. 参考资料

1. TabLayout+ViewPager实现Tab布局: <https://www.jianshu.com/p/adf7a994613a>
2. Fragment简介及使用: <https://www.jianshu.com/p/fce1e2b011ae>
3. TabLayout+ViewPager+Fragment搭配: <https://blog.csdn.net/IAFMAY/article/details/79537676>
4. TabLayout使用详解: <https://www.jianshu.com/p/7f79b08f5afa>
5. CircularProgressBar使用: <https://www.ctolib.com/brkckr-CircularProgressBar.html>
6. 修改TabLayout中的字体大小和颜色: https://blog.csdn.net/wode_dream/article/details/50424446
7. 设置TextView字体: <https://blog.csdn.net/huntersnail/article/details/48913331>
8. PopUpWindow使用详解: <https://blog.csdn.net/harvic880925/article/details/49272285>
9. PopupWindow显示在控件附近: <https://www.cnblogs.com/woaixingxing/p/5563171.html>
10. 设置View的背景透明: <https://blog.csdn.net/u011368551/article/details/80899838>
11. PopupWindow点击空白消失: <https://blog.csdn.net/gjy211/article/details/50858733>

12. 页面设计风格及布局参考: <https://pvp.qq.com/web201605/herodetail/508.shtml>
13. UI开源项目集: <https://github.com/wasabeef/awesome-android-ui>
14. MaterialDrawer: <https://github.com/mikepenz/MaterialDrawer>
15. 分页: https://blog.csdn.net/qq_16131393/article/details/50941884