**移动应用开发课程设计**

**——Focus**

作 者 姓 名： 吴统琦 谢云鹏

学 号： 1750701 1751920

指 导 教 师： 刘岩

学院、 专业： 软件学院 软件工程

同济大学

2019.12.23

目录

1.开发背景·····························3

2.开发环境·····························3

3.基本需求·····························3

4.开发流程·····························3

5.成员贡献·····························7

6.项目心得·····························7

7.参考资料·····························7

一.开发背景

近年来手机成为了人们密不可分的东西，很多人每天睁眼的第一件事就是看手机，走路在看手机，吃饭在看手机，甚至睡觉都要手机相伴.不得不说手机的出现为我们的生活带来了极大的便利但是如今手机过分的使用量已经严重影响了人们的正常学习生活.甚至由此产生了一个新型的心理疾病-------网络成瘾.他们不是不想放下手机，而是他们的确无法做到.由此我们打算开发一款限制或者提醒人们使用时长的应用.

二.开发环境

Android studio bmob（后端云数据库）

三.基本需求

用户在登录注册后，设置当天想要使用手机的时长，有两种设置方式设置相对时间是输入浮点数据，表示他一共想要使用是时间;绝对时间是有两个时间段的输入，用户可以选择设置他们将在此时间段内不想使用手机.用户完成他所设定的目标后他可以得到相应的积分.他也可以查看自己的积分情况.然后用户可以添加好友来查看好友的积分情况和历史信息.

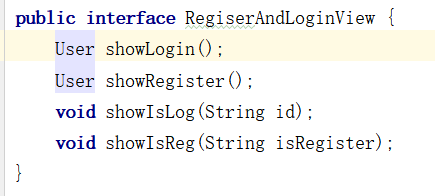
四.开发流程

在项目初期由于经验不足走了很多的误区.首先跑到了webSocket，我想在客户端收到的数据直接发到服务器并存进数据库.后来发现此路不通.然后又花了将近一个月学习Spring mvc.感觉这个做后端还可以.再后来又发现了Bmob后端数据库.如获至宝.我们的程序逻辑没有那么复杂，Bmob能在客户端直接读写云端的数据库简直完美无缺.然后又果断转移开始Bmob只旅.

在做前面的小测试时我只了解mvp的原理，感觉还不大会写mvp.然后就先试着搭了一个mvp的小demo.然后写完了所有的数据库连接类（Dao类）.然后开始基本的功能测试.

注册功能：注册是从view获取用户的信息给presenter，再给model并写进数据库，返会是否成功.

//注册和登录的view类接口.被注册登录的activity所实现.

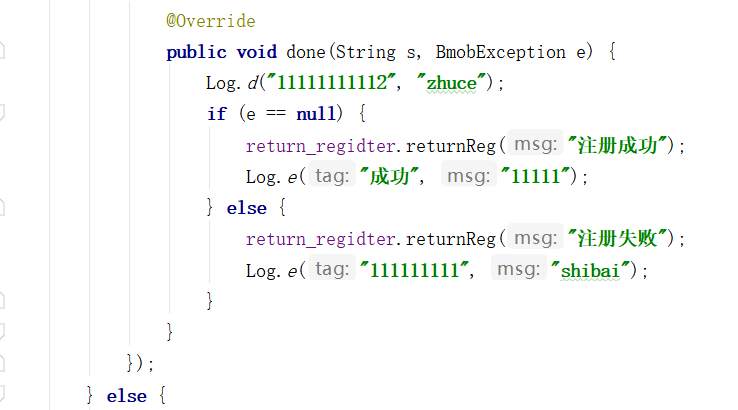


presenter层处理逻辑



model层处理

在注册的时候我首先为用户在好友表写入了一条数据，主键是用户的id其余的都是空值，然后向User表写入数据.



由于java基础不太好，在完成上述的功能的时候遇到了一个参数列表数据无法外传的问题.向Bmob数据库中写入信息后返回的失败与成功的状态值始终无法传到view界面.我尝试了使用Sharedpreferance.但是这个save方法在写数据的操作是异步执行的，sharedperfance里还没有被写进数据，前端就要展示注册的状态值，就会造成空值的闪退问题.

后来终于用接口回调解决这个问题.

在model类里声明内部接口，然后在model的实现类里调用此接口，将它传递到view层，并显示.

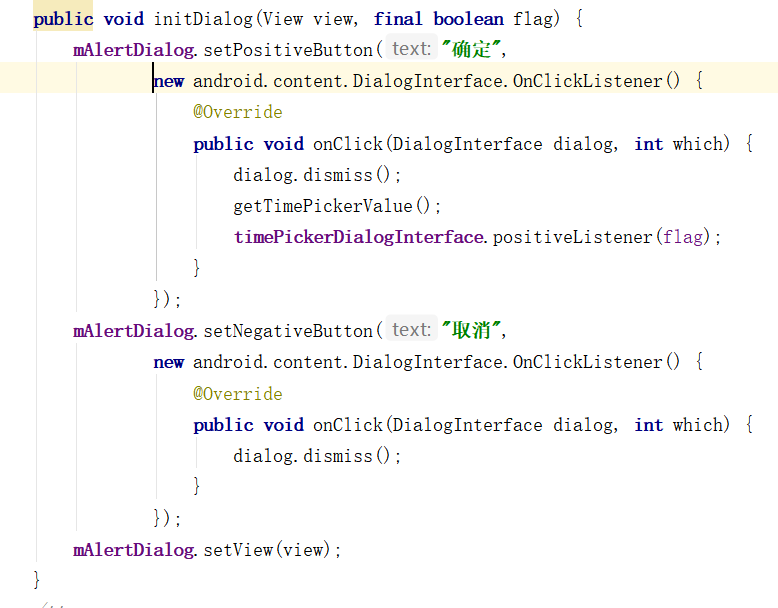
登录：读取用户的信息，匹配用户的电话或者邮箱和密码.如果成功返会用户的id，如果不成功，返会登录失败.由于其他的所有操作都是基于登录的账户的所以我用了Sharedperferance存储用户的id，这样就方便了后面的数据的读写操作.同样这个就类似本机的数据库直接写进本地文件中，保存了登录信息，免去了在后续使用的时候重复登录的麻烦.

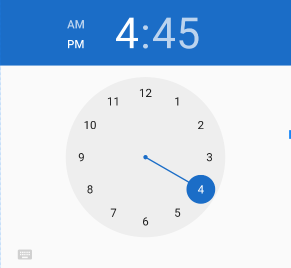
这里面遇到的问题也很莫名奇妙.在前一天我运行测试的时候一切都很正常，但是后一天软件打开就会闪退.然后连接Android studio看logcat发现是

**sharedPreferences** = getSharedPreferences(**"userid"**, ***MODE\_PRIVATE***);  
**id** = **sharedPreferences**.getString(**"userid"**, **"id is null"**);

这里是空对象.前面我是将它作为成员变量声明，出问题后我将它放在了oncreate方法里调用，发现又能运行.

设置时间：时间我采用了两种的输入方式.相对时间用户采用这种的输入直接用editview接受，写到数据库.绝对时间是需要用户输入两个之间值.我才用了timepicker这一组件，内部再使用一个接口，当用户点击确定后，将他选择的时间返回过去.然后在p层做了一下用户输入的检测，当相对时间小于0或者大于16 的时候提示输入有误，不会被写进数据库.



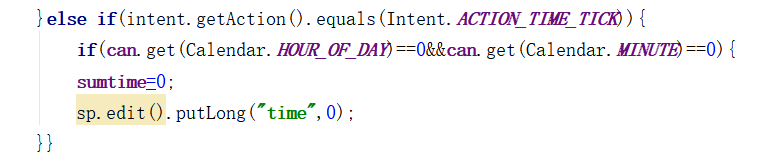


好友：没做这个功能的时候我觉得仅仅是在好友表里加上好友的id就可以了，但是在真正处理的时候我才发现好友的功能不是这个原理，首先需要一个请求添加，对方同意之后才能加进还有表.然后我又修改了数据库的设置.增加了一个request\_id表示请求添加好友的id，好友上线后检查自己有没有被添加，选择添加或者拒绝.在期初设计的时候我是想在添加的时候设置一个自己的历史记录是否好友可见的选择框.可后来一想，加了好友自己的信息不被好友可见的话，由于app没有其他的数据可被共享，然后就几乎和没有添加没啥区别，所以又删是否好友可见的功能.

添加好友首先在user表中寻找是否存在此好友的id，然后返会好友的昵称，点击添加后加入user表中的request\_id字段.实现好友的添加.

时间统计：在service服务里动态注册一个BroadcastReceiver，然后mainactivity的onCreate方法中启动，也就是在程序启动后他一直运行，在

00:00的时候改变为值为0;



五.成员贡献

吴统琦 60%

谢云鹏 40%

六.项目心得

在大一学习c，c++后我们在后面的课程学习中用到了python，C#语言，一直到学习这门课之前一直没有真正的接触过Java语言，仅仅懂一点点基础的语法.我在做东西过程中还是觉得c，c++挺好用的.但是昨完这个项目只后我深刻的感觉到Java语言的便利，具体地说也不知道如何描述，但是就是觉得Java在使用过程中相当的方便.还有各种编程语言的基础语法大多相同，没有真正的完成一个项目就永远无法渗透其中的精髓.我原本以为会了基础的语法就够了，但是自己着手去处理才会发现很多的东西并不是“相当然”的那种处理方式.例如上面提到的接口回调，本应就是个java中的基础的东西，但就是因为不会花费了好长的时间.当然主题还是安卓.以前的学习中一直做pc端的应用，终于接触到了移动应用很开心.项目结束后当然也加深了很多对安卓项目的理解.

七.参考资料

1.timepicker的使用

<https://blog.csdn.net/u012193089/article/details/48639381>

2.自定义dialog

<https://www.jianshu.com/p/f64888ca1468>

3.Bmob开发文档

<http://doc.bmob.cn/data/android/index.html>

4.mvp架构

<https://www.jianshu.com/p/242676fddc7e>

5.接口回调的简单使用

<https://www.jianshu.com/p/c182d0be0d20>

<http://www.imooc.com/article/42031>

6.server服务

<https://blog.csdn.net/cunteng008/article/details/77900576>

<https://blog.csdn.net/nixi_2017/article/details/80747312>

# 7.BoardcastReceiver广播服务

<https://blog.csdn.net/qq_21556263/article/details/82758550>