



**专业项目设计**

**网站文件上传系统**

**学 院 计算机学院**

**专 业 计算机科学与技术**

**年级班别 2017级（4）班**

**学 号** **3117004593**

**学生姓名 钟惠明**

**指导教师 张伯泉**

**2020年 10月**

目录

[1. 选题方向 1](#_Toc59206016)

[2. 选题背景 1](#_Toc59206017)

[3. 专业项目设计描述 1](#_Toc59206018)

[4. 专业项目设计的创新性分析 1](#_Toc59206019)

[4.1时间不同 1](#_Toc59206020)

[4.2教学方式不同 1](#_Toc59206021)

[4.3空间距离优势 1](#_Toc59206022)

[5. 项目实现所采用的相关技术介绍及可行性分析 1](#_Toc59206023)

[5.1所采用的相关技术介绍 1](#_Toc59206024)

[5.2可行性分析 1](#_Toc59206025)

[6. 需求分析 1](#_Toc59206026)

[6.1功能要求 1](#_Toc59206027)

[6.2性能要求 1](#_Toc59206028)

[7. 概要设计 1](#_Toc59206029)

[7.1体系结构 1](#_Toc59206030)

[7.2功能组成 1](#_Toc59206031)

[8. 详细设计 1](#_Toc59206032)

[8.1团队招募功能过程设计 1](#_Toc59206033)

[8.2通信机制功能过程设计 1](#_Toc59206034)

[8.3协同过滤推荐算法设计 1](#_Toc59206035)

[8.4数据库设计 1](#_Toc59206036)

[9. 参考文献 1](#_Toc59206037)

[10.总结 1](#_Toc59206038)

# 选题方向

完成专业项目设计是在立足于广东社会发展需要，立足于学生的学业和生活的需要的基础上，实现进一步内化基础理论，提高与专业相符合的计算机工程实践能力。现结合《计算机网络》、《数据库系统概论》、《软件工程》等课程以及部分课外书籍的知识，选择移动智能信息系统作为专业项目设计方向并实现主要基于javaweb的在线课程学习系统。

# 选题背景

在当今这高速发展的社会中，人们面临着诸多方面的压力，面对工作和生活也需要不断的学习才能不被淘汰，但身处于现代社会，人们很难像在学生时代那样每天都有一整块一整块的时间去学习了，人们逐渐习惯了在互联网上，用碎片化的时间去学习去提升自己。网上听课、学习比传统方式更便捷更自由，也更能适应如今高速发展的时代节奏。疫情期间，因为都不能外出，所以所有学生都是在网上上课，更是驱动了在线教育、在线学习和蓬勃发展，让在线学习在某方面逐渐的取代了传统的教学、学习方式。

## 专业项目设计描述

3.1客户端需求

登录注册：用户可以在网页端登录注册，保存自己的信息

播放视频：用户需要在网页端浏览文档、观看视频

文件上传：用户可以上传文档、ppt、视频等文件

文件下载：用户可以下载相关文档

3.2服务端设计需求

和前端连接交互：用于接收前端登录注册、下载文档的请求以及接收前端上传上来的文件。

连接数据库：连接数据库进行用户信息的验证和保存等

## 专业项目设计的创新性分析

以网络为主的在线学习新型学习方式，去适应现代社会中人们的学习需求，相比于传统教育方式，有一下几种优势：

### 4.1时间不同

传统教学是在学校或者教育机构里进行教学和学习的，需要统一固定的时间，还需要在路程上花费时间，学习时间受限制，而在线教学对学习时间是没有限制的，学习者可以根据自己的时间自主安排学习时间。

### 4.2教学方式不同

传统教学是在学校或者教育机构里进行教学和学习的，以教师和课堂为中心，教学方式比较单一，而在线教学则是通过网络进行交互式学习，学习者可以利用网络工具向教师提问、与同学讨论问题，教学方式多样化。

### 4.3空间距离优势

在线学习让空间距离变得不再是问题，传统教学方式需要去到学校或者教育机构去上课，但是在线教学对空间基本没有限制，学习者可以随时随地的学习，只要有网络就可以了。

## 项目实现所采用的相关技术介绍及可行性分析

### 5.1所采用的相关技术介绍

在上面的分析中，我们主要归纳了两个问题：寻找、组织队伍难的问题以及难以认识志同道合的朋友和与其进行有效交流的问题。

寻找、组织队伍难的问题在于学生缺乏渠道去联系和了解交际圈外的优秀人才，以及优秀学生存在聚堆现象，这样就和普通学生之间存在一层看不见的隔膜。

难以认识和筛选志同道合的朋友和与其进行有效交流的问题在于学生在参加比赛活动和组织寻找团队的时候，缺少队友推荐和筛选过程，往往都是只能动员身边熟悉的同学一起参与。

以上问题是我们团队主要解决的方向，也是整个专业设计项目的灵魂所在。

### 5.2可行性分析

因本人主要负责专业设计项目的Android移动端开发，因此接下来探究在Android移动端的可行性。

如今的代码编写环境和代码能够实现的功能越来越完善，在Android开发中，使用原生的View和自定义View可以实现复杂的界面，另外，通过使用Android的四大组件和Java的设计思想、设计模式可以有效地完成功能的实现。也可通过Android官方推荐的Okhttp3网络框架来与后端进行有缓存、复用链接、重定向、桥接的数据交互。利用Glide框架来进行图片的加载、缓存和压缩。除此之外，也可使用很多网络上已有的框架和实例代码来实现诸如获取手机图片、获取手机参数、图片剪裁、状态栏变色等功能。因此，本项目在Android上开发是可行的。

下面进行实现的成本进行分析。本项目需要使用开发工具：三台电脑、一台Android系统手机、一个云服务器、一个腾讯云IM服务器等。三台电脑和一台Android系统手机可使用开发人员的日常电脑和手机，因此本项目的物质实现成本主要在于租用服务器、购买学习资料、出行费用等构成。在开发人员的经济能力承受范围之内。

最后进行开发周期的可行性分析。因我们是致力于打造一款具有登录、个人信息、团队信息、社交聊天、系统推荐、搜索筛选、账户管理、关于产品等几乎包含所有主流功能的App，因此完成整个App的界面和功能实现需要较长的开发周期，且本人此前Android开发基础薄弱，所以采用过边学边做的方式，这样就更加拖了开发周期。但是我们可以先实现重要的功能，一些不重要或者边缘的功能可以留在以后慢慢完善。加之离毕业设计还有一段时间，有机会在毕业答辩前完成大部分甚至全部功能。

## 需求分析

根据问题的描述和分析，现明确用户的功能要求和性能要求：

### 6.1功能要求

本次专业设计项目主要解决上述的两个问题：寻找、组织队伍难的问题和难以认识和筛选志同道合的朋友和与其进行有效交流的问题。因此我们设计了如下三个功能模块：推荐功能模块、团队招募功能模块、通信机制功能模块。

#### 6.1.1推荐功能模块

在推荐功能模块中，主要负责向用户推荐用户感兴趣的其他用户或者团队，实现有效推荐。帮助用户快速寻找朋友和组织队伍。

#### 6.1.2团队招募功能模块

在团队功能招募模块中，主要负责团队的招募工作，包括团队是否开启招募，团队的招募职位种类和数量，团队对于招募人员的要求等。

#### 6.1.3通信机制功能模块

在通信机制功能模块中，主要负责用户之间的交流通信，包括用户之间的私聊或者团队中的群聊等。

### 6.2性能要求

本项目应对启动速度、占用内存、与服务器的交互速度、兼容性四个方面进行性能上的要求。

#### 6.2.1启动速度

启动速度指用户从桌面点击图标到登录界面的速度。要求冷启动速度越快越好，最慢不能超过2s。热启动的启动速度不超过1s。

#### 6.2.2占用内存

要求App本身（不包含用户消息、图片缓存等）总内存占用少于200MB，此值是通过参考微信150MB、Bilibili230MB得出。

#### 6.2.3与服务器的交互速度

要求产品功能的每一个和后台的交互步骤在网络良好的情况下延迟小于200ms，如果按个功能和后台的交互延迟超过500ms则需要考虑拆分步骤分步完成。

要求产品的即时通讯系统在用户网络正常的情况下，用户从发送消息到消息被接收的单向传递过程延迟不超过1000ms。

#### 6.2.4兼容性

App的兼容性包括不同版本的安卓系统兼容，不同大小的屏幕的适配。

## 概要设计

根据需求分析现对设计问题进行体系结构以及功能组成的建模。

### 7.1体系结构

我们将体系结构设计为启动、主页、我的、发现、消息五个模块，每个模块有其对应的专属功能和逻辑，力求模块之间降低耦合性，模块内部提高内聚性。详细可见图7.1。

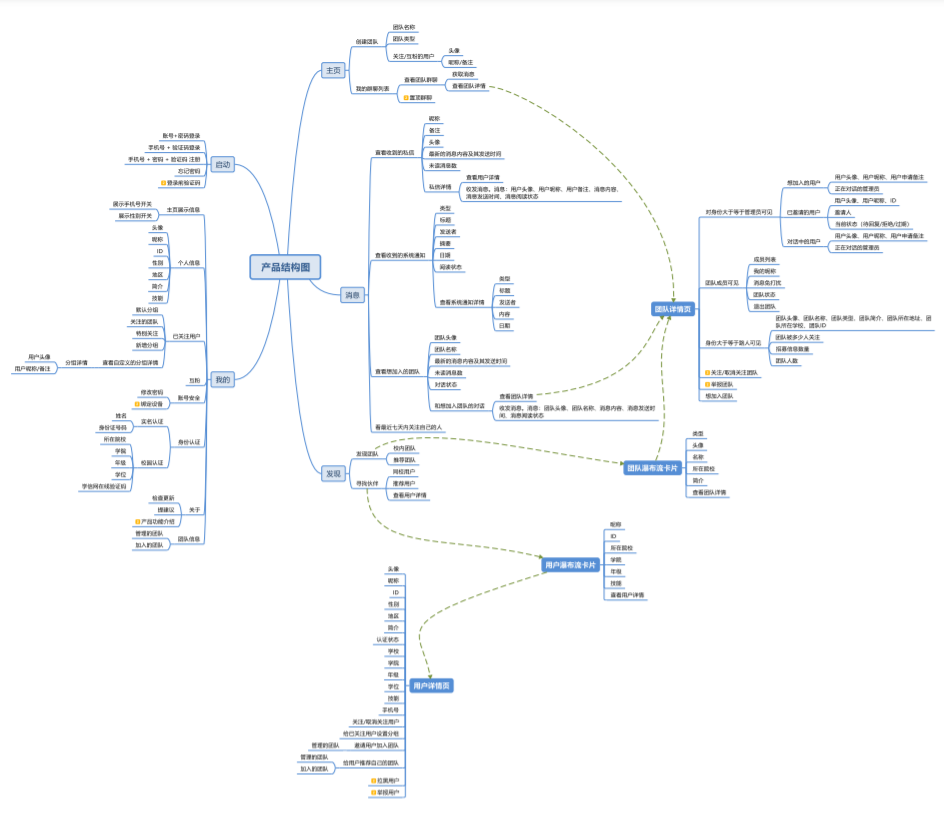


图7.1 体系结构

### 7.2功能组成

功能组成首先确定三大主要功能，然后将三大主要功能再分为不同的子功能，大功能之间互相独立，子功能之间也互相独立。

#### 7.2.1推荐功能组成

在推荐功能中，我们设计了使用非地理推进算法来对用户进行推荐，并明确将要使用协同过滤算法。系统会根据协同过滤算法来推荐用户或团队给用户。详细可见图7.2。

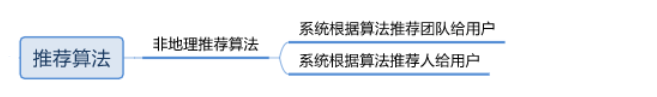
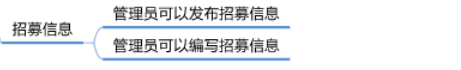


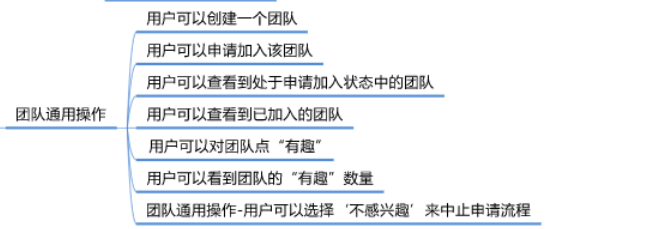
图7.2推荐功能组成

#### 7.2.2团队招募功能组成

在团队招募功能中，我们设计成在团队中，只有队长或管理员有权力编写和发布招募信息，并且对申请的用户进行筛选和录用操作。用户可以在团队的主页上查看该团队的招募信息，并且申请加入该队伍。用户可以在软件中查看其申请团队的状态和加入团队的状态。用户可以自由选择取消申请或退出团队。详细可见图7.3。



（a）团队招募部分功能组成



（b）团队招募部分功能组成

图7.3团队招募功能组成

#### 7.2.3通信机制功能组成

在团队招募功能中，我们设计了个人私聊和团队群聊两种通信方式。下面主要介绍团队群聊的通信方式。

在团队群聊中，设计了群聊管理功能和群聊通用功能。群聊管理功能包括团队管理员可以增加自定义的群聊、团队管理员可以编辑群聊信息、团队管理员可以编辑群聊公告等；群聊通用功能包括群聊可以@同一群聊内成员、群聊成员可以发送文字、群聊成员可以设置群聊免打扰等。详细可见图7.4。

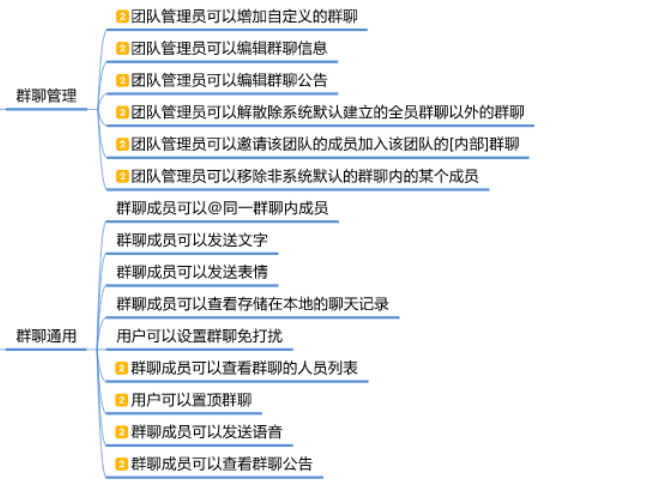


图7.4通信机制功能组成

## 详细设计

### 8.1团队招募功能过程设计

#### 8.1.1团队管理员发布招募信息过程设计

管理员在发起招募时，必须先开启团队中的自由招募功能。否则其他用户将无法从团队主页中查看团队的招募信息并申请加入该团队。在开启自由招募后，管理员可以编写并发布招募信息，如对已发布的招募信息不满意，管理员还可以重新修改或删除招募信息要求。

详细管理员发布招募信息过程可见图8.1。

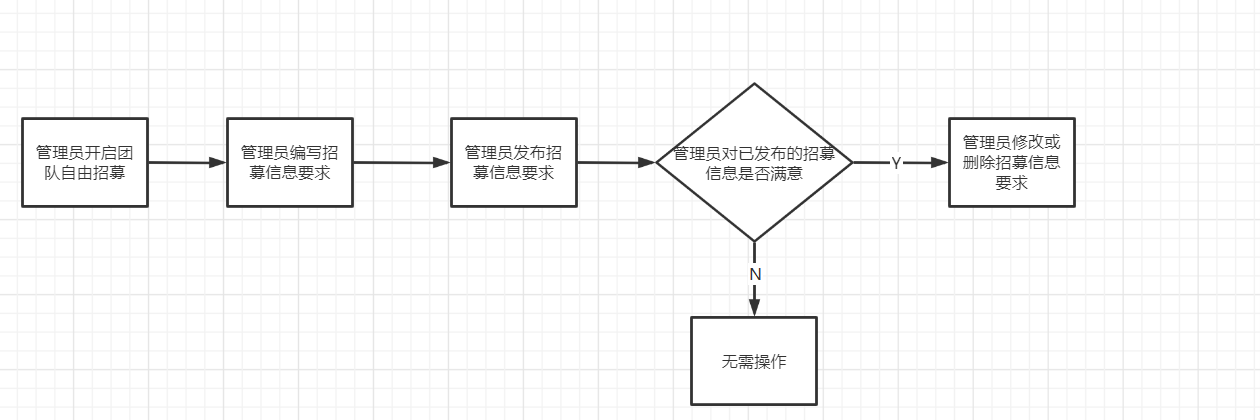


图8.1 管理员发布招募信息过程设计图

#### 8.1.2用户申请加入团队过程设计

我们设计了两种用户加入到团队的方式：团队队长或管理员邀请和用户主动申请加入团队，下面对用户主动申请加入团队这一过程进行设计。

用户在通过系统推荐或主动搜索进到团队主页后，可以选择加入该团队。点击选择加入团队的按钮后，客户端将判断该团队是否开启了自由招募状态，若团队未开启自由招募状态，则提示用户该团队不需要找人；若团队开启了自由招募状态，则进行下一步。接下来用户可以选择感兴趣的团队招募信息或者直接跳过该过程，最后则是用户进行完善备注和发送申请的操作。

详细用户主动申请加入团队的过程可见图8.2。

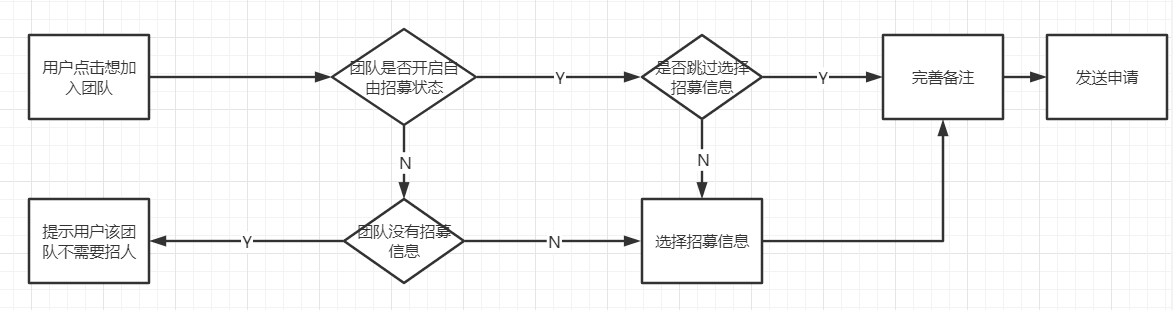


图8.2用户主动申请加入团队过程设计图

#### 8.1.3管理员处理申请过程设计

管理员可在团队主页中查看当前是否有用户的申请信息，若有，则可以先查看用户的信息和申请内容。查看用户信息后，若对用户十分满意可以直接接受用户的申请。若对用户抱着观察的心理，可选择与用户进行聊天，通过聊天来判断接受还是拒绝用户的申请。若对用户无感，可直接拒绝该用户的申请。

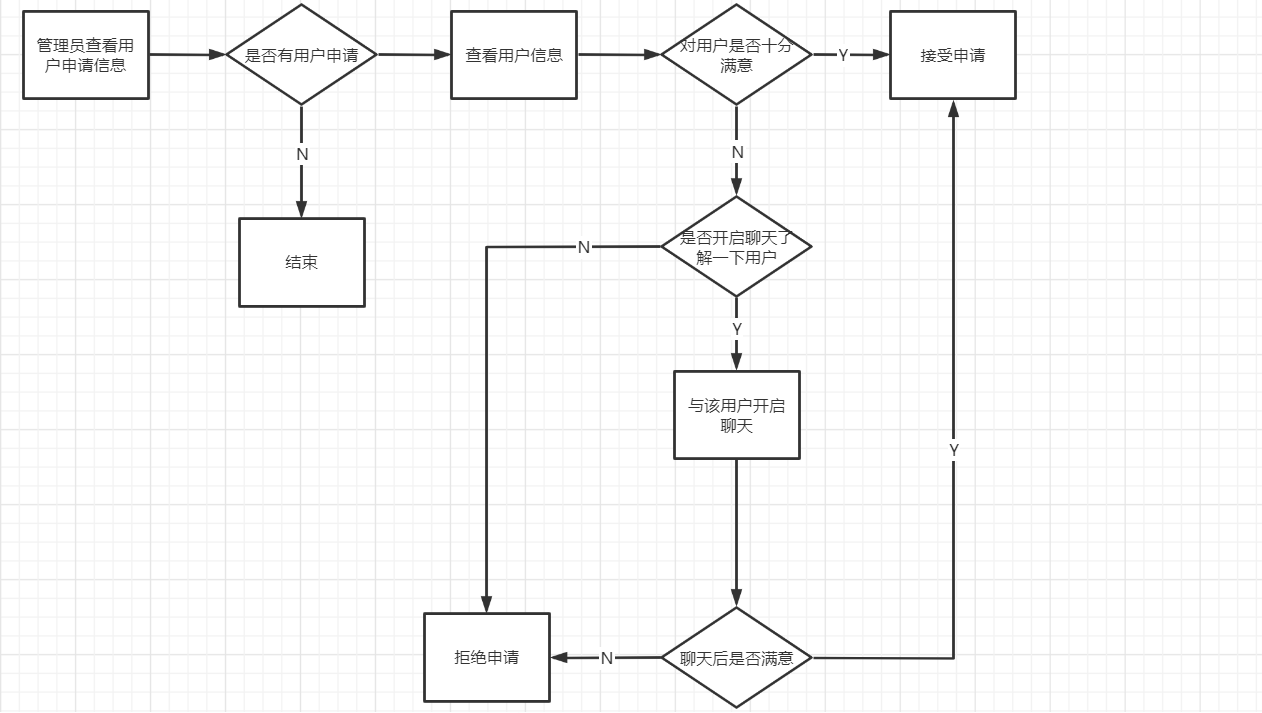


图8.3 管理员处理申请过程设计图

### 8.2通信机制功能过程设计

#### 8.2.1用户主动发起通信过程设计

若用户想对某用户发起聊天，可直接在该用户的主页中点击发起聊天，点击后会进入到双方的聊天界面中。在聊天界面中，用户可以发送文字、表情、文件等消息给目标用户，服务器在收到消息后，会将消息转发给目标用户。

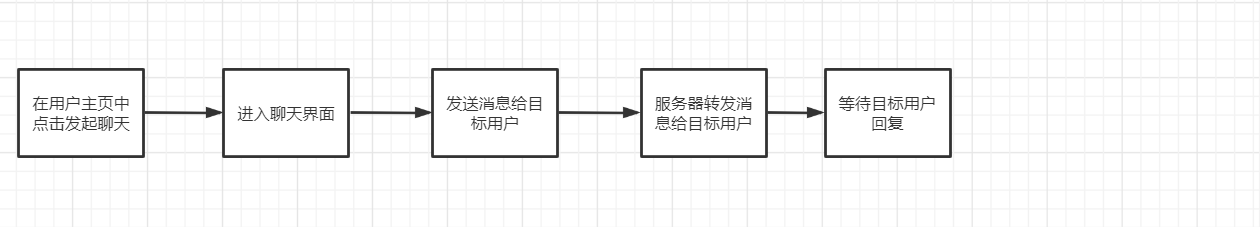


图8.4 用户主动发起通信过程设计图

#### 8.2.2用户接收消息并回复过程设计

若有其他用户向本用户发送消息，用户可以在主页或者消息界面中查看是否有新的消息，若有新的消息，可以点击聊天列表中的消息项来进入聊天界面，并且对消息进行回复。

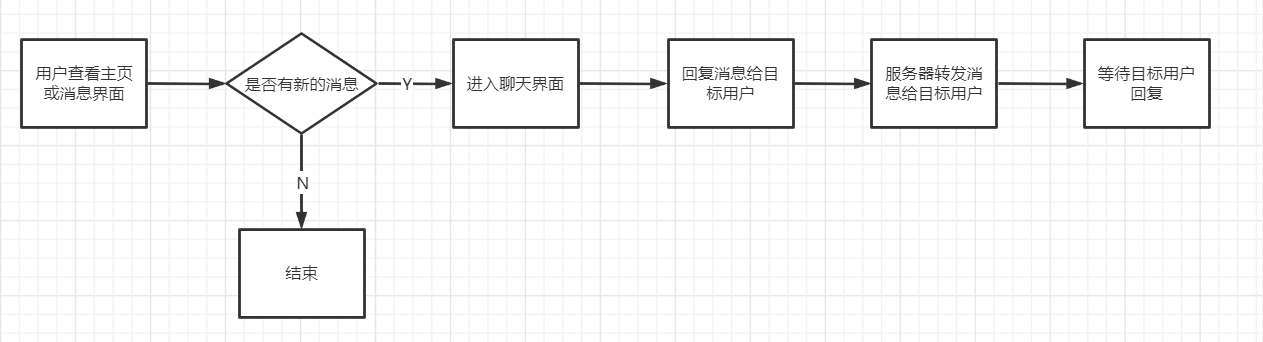


图8.5 用户接收消息并回复过程设计图

### 8.3协同过滤推荐算法设计

#### 8.3.1协同过滤推荐算法介绍

协同过滤推荐算法是目前推荐算法中最主流的推荐算法之一，它具有种类的多样性、使用的方便性等特点，具有十分广泛的应用。

它能够过滤机器难以自动内容分析的信息，如艺术品，音乐等；能够共用其他人的经验，避免了内容分析的不完全或不精确，并且能够基于一些复杂的，难以表述的概念进行过滤；具有推荐新信息的能力，可以发现内容上完全不相似的信息，用户对推荐信息的内容事先是预料不到的，可以发现用户潜在的但自己尚未发现的兴趣偏好；推荐个性化、自动化程度更高、能够有效的利用其他相似用户的回馈信息、加快个性化学习的速度。

#### 8.3.2协同过滤推荐算法原理

下面我们结合实际项目以及图8.6和表8.1来介绍协同过滤推荐算法的原理。

我们知道，一个用户可以同时加入多个团队，如用户1加入了团队1、团队2和团队3，用户2则加入了团队1、团队2和团队4，用户3则加入了团队2和团队4。从表8.1中，我们可以清晰地看到，用户1和用户2具有更高的相似性，那么，我们就可以把用户2加入的，而用户1未加入的团队4推荐给用户1。

通过上面描述的简单例子，相信已经阐述清楚了协同过滤推荐算法的主要原理。

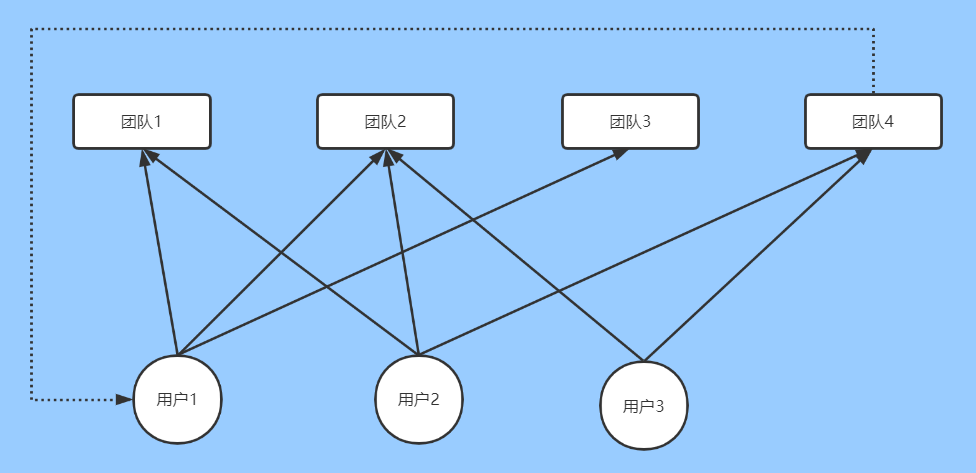


图8.6协同过滤推荐算法原理图

|  |  |
| --- | --- |
| 用户加入的团队 | |
| 用户1 | {团队1，团队2，团队3} |
| 用户2 | {团队1，团队2，团队4} |
| 用户3 | {团队2，团队4} |

表8.1 协同过滤推荐算法原理表

#### 8.3.3协同过滤推荐算法流程

在本项目中，我们使用了Mahout来实现协同过滤推荐算法。Mahout是Apache基金会基于Java的数据挖掘与机器学习类库，提供了关于推荐系统所需的分类、用户相似度计算、紧邻用户计算等工具类。

因此我们设计的协同过滤推荐算法流程也是在Mahout的基础上设计的。以下是我们的主要流程（可见图8.7）：

1. 搜集用户历史加入的所有团队信息。
2. 计算所有用户之间的相似度。
3. 针对每个用户user，获得用户user的相似用户排序表。
4. 选择与user最相似的k个用户。
5. 获得这k个用户加入过，但user未加入过的团队。
6. 推荐这些团队给用户user。

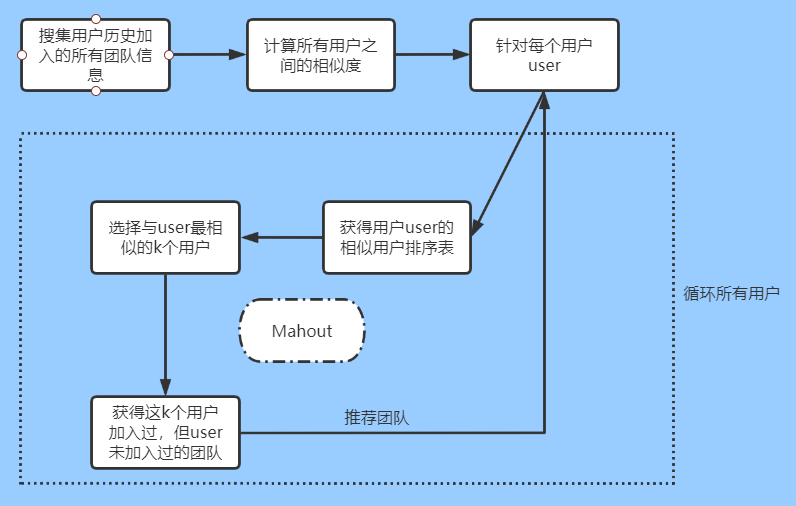


图8.7协同过滤推荐算法流程图

### 8.4数据库设计

#### 8.4.1团队招募功能相关数据库设计

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 属性名 | 类型 | 约束 |
| 项目组id | group\_id | int | 主键 |
| 组长id | group\_leader\_id | int | 非空 |
| 项目组名称 | group\_name | varchar(21) | 非空 |
| 成员数量 | group\_member\_num | int | 非空，默认为1 |
| 项目组类型 | group\_type\_id | int | 外键（表group\_type） |
| 项目组自由招募状态 | group\_recruit\_status | boolean | 非空,默认为true |
| 简介 | group\_introduction | varchar(600) |  |
| 关注数 | group\_funny | int | 非空，默认为0 |
| 点赞总数 | group\_like | int | 非空，默认为0 |
| 项目组状态 | group\_status | tinyint | 非空，默认为0  {0表示正常，  1表示解散} |
| 项目组头像 | group\_img\_path | varchar(255) |  |
| 所在地区名称 | group\_area\_name | varchar(255) |  |
| 学校名称 | group\_school\_name | varchar(255) |  |
| 创建时间 | create\_time | timestamp |  |
| 更新时间 | update\_time | timestamp |  |

表8.2 团队信息数据库表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 属性名 | 类型 | 约束 |
| 关系id | group\_application\_id | int | 主键 |
| 用户id | user\_id | int | 外键(表user\_account) |
| 团队id | group\_id | int | 外键(表group\_info) |
| 申请状态 | group\_application\_status | tinyint |  |
| 申请备注 | group\_application\_tip | varchar(200) |  |
| 招募信息id | group\_recruit\_id | int | 外键(表group\_recurit) |
| 创建时间 | create\_time | timestamp |  |
| 更新时间 | update\_time | timestamp |  |

表8.3 用户处于申请中的团队的数据库表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 属性名 | 类型 | 约束 |
| 招募信息id | group\_recruit\_id | int(11) | 主键 |
| 团队id | group\_id | int(11) | 外键（表group\_info） |
| 招募类型 | group\_recruit\_type | int(11) | 非空 |
| 职责描述 | group\_recruit\_description | varchar(600) |  |
| 标题 | group\_recruit\_title | varchar(150) | 非空 |
| 面向类型 | group\_recruit\_toward | int(11) | 非空 |
| 更新时间 | update\_time | timestamp |  |

表8.4 团队招募信息数据库表

#### 8.4.2通信机制功能相关数据库设计

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 属性名 | 类型 | 约束 |
| 群聊id | group\_chat\_id | int | 主键 |
| 项目组id | group\_id | int | 外键（表group\_info） |
| 群聊类型 | group\_chat\_type | int | 非空 |

表8.5 团队群聊信息数据库表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 属性名 | 类型 | 约束 |
| 关系id | group\_chat\_user\_relation\_id | int | 主键 |
| 群聊id | group\_id | int | 外键（表group\_chat） |
| 用户id | user\_id | int | 外键（表user\_accrount） |

表8.6 群聊与用户的关系数据库表

#### 8.4.3推荐算法功能相关数据库设计

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 属性名 | 类型 | 约束 |
| group\_log\_id | 团队日志id | int | 主键 |
| user\_id | 用户id | int | 外键（表user\_account） |
| group\_id | 团队id | int | 外键（表group\_info） |
| join\_time | 加入时间 | timestamp |  |

表8.6 用户加入团队历史数据库表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 属性名 | 类型 | 约束 |
| 推荐id | recommend\_group\_id | int | 主键 |
| 用户id | user\_id | int | 外键（表user\_account） |
| 团队id | group\_id | int | 外键（表group\_info） |
| 反馈情况 | feedback | bit(1) |  |
| 推导时间 | derive\_time | timestamp |  |

表8.7 团队推荐历史数据库表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 属性名 | 类型 | 约束 |
| 推荐id | recommend\_user\_id | int | 主键 |
| 用户id | user\_id | int | 外键（表user\_account） |
| 被推荐用户id | user\_recommended\_id | int | 外键（表user\_account） |
| 反馈情况 | feedback | bit(1) |  |
| 推导时间 | derive\_time | timestamp |  |

表8.8 用户推荐历史数据库表

## 参考文献

[1] 张海藩，牟永敏.软件工程导论（第6版）[M].北京：清华大学出版社，2013.

[2] 王珊，萨师瑄.数据库系统概论[M].北京：高等教育出版社，2014.

[3] 郭霖.第一行代码——Android[M].北京：人民邮电出版社，2016.

[4] 任玉刚.Android开发艺术探索[M].北京：电子工业出版社，2015.

[5] 启舰.Android自定义控件开发入门与实践[M].北京：电子工业出版社，2018.

## 10.总结

在实现专业项目设计的过程中，我选择结合自身所学，根据《软件工程导论（第6版）》一书中的内容如：可行性研究、需求分析、概要设计、详细设计等来进行软件的设计。

在项目设计完成后，开始寻找与推荐算法有关的博客或项目，了解当前较为流行的推荐算法如基于内容推荐算法，热点推荐算法，协同过滤推荐算法等。最终选择了协同过滤推荐算法作为本次专项设计的实现算法。在本次专项设计中，我不仅要学习与Android有关的前端知识，还要学习与推荐系统相关的后台知识，对于个人来说是一次十分巨大的挑战。

做专业项目设计的时候，发现了许多自己的不足，如对基础知识掌握的还是不够深，对于后台和数据库方面的知识十分薄弱；在面对整个屏幕的报错信息时，难以保持内心的平静；在做项目的时候常常难以保持专注等。这些都是在以后急需改正的地方。

尽管存在很多工作上的不足，但是对于本次专业项目设计的目标和目的是不变的，我一直保持着进行本次项目设计的初心：致力打造出一个可为学生提供完整且可靠的比赛活动信息来源的app，一个自由的组队和社交的app，一个符合用户需求的人才推荐匹配app，一个具有简洁且面面俱到的界面体验感的App。使学生不再为缺乏比赛活动信息而苦恼，不再为找不到志同道合的队友而烦恼，不再为简历上的一片空白而感到无力。一起遇见有趣的人，一起做有趣的事。