**[同城校友网]**

**毕业论文**

专业：[计算机科学与技术]

班级：[2011级1班]

学生姓名：[张沛]

指导教师：[李磊军]

完成时间： 2015年4月8日



[1.摘要 5](#_Toc416299119)

[2.绪论 5](#_Toc416299120)

[2.1选题背景 5](#_Toc416299121)

[2.2国内外发展情况 6](#_Toc416299122)

[2.3研究意义 8](#_Toc416299123)

[2.4论文的组织结构 8](#_Toc416299124)

[2.5小结 8](#_Toc416299125)

[3系统需求分析 9](#_Toc416299126)

[3.1系统总体目标 9](#_Toc416299127)

[3.2功能需求 9](#_Toc416299128)

[3.3性能需求 9](#_Toc416299129)

[3.3.1数据精确度 9](#_Toc416299130)

[3.3.2页面设计 9](#_Toc416299131)

[3.3.3页面自适应性 10](#_Toc416299132)

[3.4：系统功能模块结构 10](#_Toc416299133)

[3.4.1功能结构图 10](#_Toc416299134)

[3.4.2后台管理系统 10](#_Toc416299135)

[3.4.3前台 11](#_Toc416299136)

[后台系统管理的模块设计（删除） 12](#_Toc416299137)

[前台模块设计（删除） 18](#_Toc416299138)

[3.5数据需求（根据实体类来决定） 21](#_Toc416299139)

[3.5.1用户信息 21](#_Toc416299140)

[3.5.3学校信息 21](#_Toc416299141)

[3.5.3地址信息 21](#_Toc416299142)

[3.5.4群组信息 21](#_Toc416299143)

[3.5.5帖子信息 21](#_Toc416299144)

[3.7小结 22](#_Toc416299145)

[4数据库设计 22](#_Toc416299146)

[4.1数据库的概念 22](#_Toc416299147)

[4.2数据库的选择 22](#_Toc416299148)

[4.3数据库表的设计 23](#_Toc416299149)

[4.3.1概念数据模型 23](#_Toc416299150)

[4.3.2物理数据模型 23](#_Toc416299151)

[4.3.3表格清单 24](#_Toc416299152)

[5.系统开发要点 24](#_Toc416299153)

[5.1系统开发环境及开发工具的使用及介绍 24](#_Toc416299154)

[5.1.1eclipse 24](#_Toc416299155)

[5.1.2mysql 24](#_Toc416299156)

[5.1.3SVN 25](#_Toc416299157)

[5.1.4Maven 25](#_Toc416299158)

[5.1.5JDk1.7以上 25](#_Toc416299159)

[5.1.6tomcat7.0以上 25](#_Toc416299160)

[5.2系统开发相关技术 25](#_Toc416299161)

[5.2.1Html 25](#_Toc416299162)

[5.2.2Css 25](#_Toc416299163)

[5.2.3javascript 26](#_Toc416299164)

[5.2.4Servlet 26](#_Toc416299165)

[5.2.5Jsp 26](#_Toc416299166)

[5.2.6Mysql 27](#_Toc416299167)

[5.2.7JQuery 27](#_Toc416299168)

[5.2.8：Bootstrap 28](#_Toc416299169)

[5.2.9：nutz 28](#_Toc416299170)

[5.3小结 29](#_Toc416299171)

[6系统实现 29](#_Toc416299172)

[6.1后台管理系统 30](#_Toc416299173)

[6.1.1用户管理 30](#_Toc416299174)

[6.1.2学校管理 33](#_Toc416299175)

[6.1.3地址管理 35](#_Toc416299176)

[6.1.4群组管理 38](#_Toc416299177)

[6.1.5帖子类型管理 39](#_Toc416299178)

[6.1.6帖子管理 41](#_Toc416299179)

[6.2前台 42](#_Toc416299180)

[6.2.1首页 43](#_Toc416299181)

[6.2.2全部群组 43](#_Toc416299182)

[6.2.3群组详情页 43](#_Toc416299183)

[6.2.4帖子详情页 43](#_Toc416299184)

[6.2.5注册页 43](#_Toc416299185)

[6.2.6登录页 43](#_Toc416299186)

[6.2.7个人中心 44](#_Toc416299187)

[6.1.2帖子管理 44](#_Toc416299188)

[6.3公共技术组件 44](#_Toc416299189)

[6.3.1分页组件 44](#_Toc416299190)

[6.3.2提示组件 44](#_Toc416299191)

[6.3.2基础的服务类 44](#_Toc416299192)

[5.3管理员管理用户页面设计（删除） 46](#_Toc416299193)

[5.4用户登录之后根据不同身份有不同的操作权限设计（删除） 47](#_Toc416299194)

[**普通管理员** 47](#_Toc416299195)

[**超级管理员** 48](#_Toc416299196)

[7系统安装配置 48](#_Toc416299197)

[结束语 48](#_Toc416299198)

[参考文献 48](#_Toc416299199)

[附录 48](#_Toc416299200)

[致谢 48](#_Toc416299201)

# 1.摘要

随着中国的社会迅速发展，越来越多的人走出家门，奔向各大城市努力奋斗，除了能获得更多的经济收入之外，还获得更多的工作机会和见识，基于这样的趋势，越来越多的学校同学加入这个洪流，在新的环境里，衣食住行都会在一个新的陌生的环境中慢慢成型，在新的环境中，由于各方面的不熟悉，身边也没有什么可以信赖的群体，导致会遇到各种问题，影响自己的发展。

本项目的开发目的，就是为了解决这样的学生群体遇到的问题，校友网主要是用于已毕业的校友之间的人际交流，尤其是同城之间的学习，工作经验交流，及生活等沟通，通过校友网这个社交网站达到日常交流。

**【关键词】：学生，陌生，交流，新环境，信赖，社交网站**

# 2.绪论

## 2.1选题背景

**【社交网络服务（亦称社交网站）】**主要作用是为一群拥有相同兴趣与活动的人创建在线社区。这类服务往往是基于互联网，为用户提供各种联系、交流的交互通路，如电子邮件、实时通信服务等。此类网站通常通过朋友，一传十、十传百地把网络展延开去，就像树叶的脉络，华语地区一般称之为“社交网站”。

多数社交网络会提供多种让用户交互起来的方式，可以为聊天、寄信、影音、文件分享、博客、新闻组等。

社交网络为信息的交流与分享提供了新的途径。作为社交网络的网站一般会拥有数以百万的登记用户，使用该服务已成为了用户们每天的生活。社交网络服务网站当前在世界上有许多，知名的包括Google+、Myspace、Plurk、Twitter、Facebook等。在中国大陆地区，社交网络服务为主的流行网站有人人网、QQ空间、百度贴吧、微博等。

## 2.2国内外发展情况

**【社交网络服务的发展】**在早期的互联网上，有许多提供用户间交互交谈的服务，例如：BBS，新闻组等。早期社交网络的服务网站呈现为在线社区的形式。用户多通过聊天室进行交流。

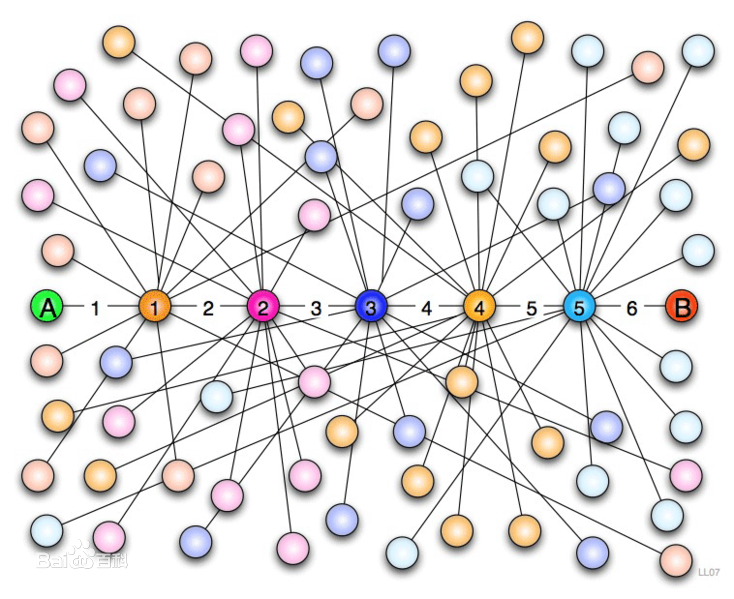
随着博客等新的网上交际工具的出现用户可以通过网站上创建的个人主页来分享喜爱的信息。

2002年至2004年间，世界上三大最受欢迎的社交网络服务类网站是Friendster、MySpace、Bebo。

在2005年之际MySpace成为了世上最巨大的社交网络服务类网站。传闻当时其页面访问量超越了作为著名搜索引擎的Google。

2006年第三方被允许开发基于Facebook的网站API的应用，使得Facebook随后一跃成为全球用户量增长最快的网站。有更多的网站随后加入自己或是第三方开发者的API。

社交服务网站的发展验证了**【六度分隔理论】**，即“人际关系脉络方面你必然可以通过不超出六位中间人间接与世上任意先生女士相识”。个体的社交圈会不断地扩大和重叠并在最终形成大的社交网络。在此类通过对“朋友的朋友是朋友”原则的实现而得到发展的线上社交网络中，Friendster具备一定的代表性。



但“朋友的朋友”，只是社交拓展的一种方式，而并非社交拓展的全部。因此，一般所谓的SNS，则其含义还远不止“朋友的朋友”这个层面。比如根据相同话题进行凝聚（如贴吧）、根据爱好进行凝聚（如Fexion网）、根据学习经历进行凝聚（如Facebook，人人网）、根据工作经历进行凝聚（如尚班族、大街网）、根据周末出游的相同地点进行凝聚、根据中国农民应用网络的方式凝聚（如农享网）等，都被纳入“SNS”的范畴。

社交服务类网站最早出现商业盈利目的是在2005年3月雅虎对雅虎360°的推出。在2005年6月新闻集团成功收购MySpace。随后在2005年12月，英国ITV购得Friends Reunited。此后在世界的各地涌现出各种不同语言的社交网络服务类网站。

**【社交网络的基本功能】**

* 纪录个人数据
* 私讯功能
* 用户相互链接的功能
* 用户检索的功能
* 日记（博客）的功能
* 社区的功能（包括公开的社区(open group)、不公开的社区(not open group)、秘密社区(closed group)等。）

**【社交网络的高级功能】**

* 提供地点打卡的功能。
* 可以针对某一议题设立可讨论的专页，惟其权限只限于开设版面的创版人与其所允许的管理员。

## 2.3研究意义

随着人们的沟通方式的多样化，信息化水平的不断提高，网络上出现了越来越多的各种各样的交流平台，有专业技术交流，也有日常生活的点点滴滴，例如，58同城，赶集类的网站侧重解决越来越多的人们工作难，就业难的问题，饿了么，最近火热的订餐网站，。。。。我们会发现越来越多的交流网站的目标在不断的细化，不断的切合实际，解决人们日益出现的问题，对于毕业生找工作的，或者马上也毕业，准备实习的同学来说，他们的压力也很大，除了要面临更大的毕业生多，就业岗位少的现实外，还要顶着各种没有工作经验，没有实践能力的公司苛刻，所以缓解大学生就业压力很重要，虽然有人人网等一些校内网站在解决各自问题，但是对于大部分毕业生的就业问题的的解决需要全面的把握！此社交网站在解决毕业生矛盾的同时，对于维护社会的安定也有潜在的意义

## 2.4论文的组织结构

**用visio来描述**

## 2.5小结

本绪论先从国内国外关于社交网络的发展做了概要的分析，然后抓住当前社会中部分社交网站涉及到的问题，和社会中日益凸显的毕业生就业难的矛盾关系，基于此，抓住这个矛盾，针对于这个问题，设计了这个面向解决这样的问题的社交网站，以此来弥补社会上已经出现的类似网站的不足，突出解决这样的社会问题！

# 3系统需求分析

## 3.1系统总体目标

实现用户在某个地区可以在所在的学校的群组中进行信息共享，和信息交流沟通，达到通过同校的较高信任度来完成在新的环境中快速适应和更好的工作和生活的目的，通过同校之间的互帮互助，让学校的温暖在需要的地方为需要的校友送去幸福，以此增进校友之间的友谊。

## 3.2功能需求

前台功能：用户可以注册登录此校友网，可以加入群组，退出群组，发表帖子，上传文件

后台功能：实现全国主要城市的学校的信息的采集和维护(lucene),角色管理，权限管理，用户管理，群组管理，帖子管理，文件管理

## 3.3性能需求

### 3.3.1数据精确度

在搜索时，可以进行模糊查询

省市显示的时候异步处理数据

### 3.3.2页面设计

操作界面简单明了，易于操作，对格式和数据类型限制的数据，进行验证，包括客户端验证和服务器验证，并采用错误提醒机制，提示用户输入正确数据和密码的 安全保密性

只有合法用户才能登录使用系统，对每个用户都有权限设置。对登录名、密码、以及用户重要信息进行加密，保证账号信息安全。 可维护性

### 3.3.3页面自适应性

采用自适应框架，页面在浏览器窗口大小改变的情况下，可以做出相应的调整，满足页面效果人性化的需求

## 3.4：系统功能模块结构

### 3.4.1功能结构图

### 3.4.2后台管理系统

后台管理系统，由系统管理员登陆后使用，主要针对系统各个模块数据进行统一维护、查看，包括用户管理，学校管理，地址管理。群组管理，帖子管理，帖子类型管理。

#### 3.4.2.1用户管理

查看已经注册用户列表信息，点击单个用户查看用户详细信息。统计总的注册用户数。

#### 3.4.2.3学校管理

查看所有学校列表信息，提供初始化学校功能，点击初始化学校，去新浪网站（<http://edu.sina.com.cn/gaokao/2014-08-08/1903430534.shtml>）采集全国高效名单：

#### 3.4.2.4地址管理

查看各个省份信息，包括编码，名称，经纬度，点击省份名称，查看该省份下的所有地市信息。

#### 3.4.2.5群组管理

查看所有群组列表信息，对群组进行审批操作，包括审批为活动群组，解散群组，关闭群组，解散和关闭群组的时候需要填写操作原因，并发送消息给群组创建人。

#### 3.4.2.6帖子管理

查看所有帖子列表信息，点击可以查看帖子详细内容。

#### 3.4.2.7帖子类型管理

查看所有帖子类型列表信息，增加、修改、删除帖子类别。

### 3.4.3前台

#### 3.4.3.1首页

网站首页，展示最新创建的若干群组，展示最热（按照帖子回复数量作为帖子热度的依据）的若干群组。显示最新发布的若干帖子。显示评论量最大的若干帖子。

#### 3.4.3.2群组列表首页

点击帖子的“全部群组”进入群组列表页面。该列表页，以列表形式展示所有群组。点击群组链接可以进入群组详情页面

#### 3.4.3.3群组详情页面

展示群组详细信息，包括群组创建信息，群组成员。该群组下的帖子列表。并提供发布帖子和加入（或者退出）该群组的链接功能。

#### 3.4.3.4帖子详情页

在任何地方点击帖子链接，进入该页面，该页面展示帖子详细信息，以及帖子的所有回复信息，并提供帖子的回复功能。

#### 3.4.3.5用户注册

用户在该页面输入详细注册信息，即可注册为网站用户。

#### 3.4.3.5用户登录

用户输入正确的登陆账号和密码，即可成功登录该网站，登录后举报发布帖子，回复帖子，加入群组等功能。

#### 3.4.3.6个人中心

用户登陆后可以看到个人中心链接，进入该页面，可以看到该用户在网站的所有信息，包括加入的群组，创建的群组，发布的帖子，查看以及完善用户个人信息。并具备修改密码和查看收件箱功能。

### 后台系统管理的模块设计（删除）

#### 用户管理

|  |  |
| --- | --- |
| 功能1名称 | 用户列表页 |
| 功能描述 | 以分页形式查看用户列表信息 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：用户表。  2.点击系统管理首页的用户管理连接，进入用户列表页面。  3.用户列表页以分页形式显示用户信息。  4.用户名称以链接显示，点击可以查看用户详细信息 |
| 补充说明 |  |
| 功能2名称 | 用户详情页 |
| 功能描述 | 在用户列表页面，点击用户名链接，打开用户详情页面 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：用户表。  2.显示用户详细信息，用户加入的群组列表，用户发送的帖子列表（因为帖子可以根据帖子日期和帖子话题和发帖人坐在的群组分类，所以这里可以加入搜索链接，选择搜索方式（日期，话题，群组）来模糊查询具体的帖子列表。 |
| 补充说明 |  |

#### 学校管理

|  |  |
| --- | --- |
| 功能1名称 | 数据初始化 |
| 功能描述 | *通过抓取网上较全面的学校信息入库到学校表* |
| 数据结构与算法 | 1.该功能使用数据表结构包括：学校表。  2.学校列表页面增加初始化按钮，点击初始化即清空学校所有数据，并从网上抓取所有学校数据插入到学校表中。 |
| 补充说明 |  |
| 功能2名称 | 学校列表页 |
| 功能描述 | 以分页形式展示学校信息 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：\_学校表。  2.列表页面学校名称为超链接，点击进入学校详情页面 |
| 补充说明 |  |
| 功能3名称 | 学校详情页 |
| 功能描述 | 以分页形式展示学校信息 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：\_学校表。  2.列表页面学校名称为超链接，点击进入学校详情页面（学校名称，学校地址，学校简介 |
| 补充说明 |  |

#### 地址管理（地址管理表存在的意义）

|  |  |
| --- | --- |
| 功能1名称 | 地址列表页 |
| 功能描述 | 以分页形式展示地址列表信息 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：地址表。  2.在系统管理界面点击地址管理按钮，进入地址列表页  3.地址列表页面分页展示地址信息，右上角具备，增加，删除，两个按钮，点击分别进入地址增加和删除页面  4.点击地址名称超链接，进入地址编辑页面 |
| 功能2名称 | 地址添加页 |
| 功能描述 | 添加地址信息 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：地址表。  2.在地址列表页，点击添加按钮，进入该界面。  3.输入地址基本信息，点击保存，保存成功后跳转到地址列表页面，保存失败后，重新跳转至地址添加页面。 |
| 补充说明 |  |
| 功能3名称 | 地址删除功能 |
| 功能描述 | 在地址列表页面，选中要删除的地址，点击删除按钮，删除地址 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：地址表。  2.选中要删除的一个或者多个地址，点击删除按钮，删除成功，跳转至地址列表页  3，如果一个地址都没有选择，点击删除按钮，则提示，必须至少选择一个地址。 |
| 补充说明 |  |
| 功能4名称 | 地址编辑功能 |
| 功能描述 | 更改地址基本信息 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：地址表。  2.在地址列表页，选择地址名称，打开地址编辑界面。  3.输入新的地址信息，点击保存，保存成功跳转至地址列表页。 |
| 补充说明 | 地址管理从（学校，用户，同城，三个方面来考虑） |

#### 群组管理

|  |  |
| --- | --- |
| 功能1名称 | 群组列表页 |
| 功能描述 | 以分页形式展示群组列表信息 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：群组表。  2.在系统管理界面点击群组管理按钮，进入群组列表页  3.列表页面分页展示群组基本信息  4.点击群组名称超链接，进入群组详情页面 |
| 功能2名称 | 群组详情页 |
| 功能描述 | 查看群组详情信息 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：群组表。  2.在群组列表页，点击群组名称超链接，进入群组详情界面。  3.群组详情页面，显示群组基本信息，显示该群组成员列表，显示封禁按钮， |
| 补充说明 |  |
| 功能3名称 | 群组封禁功能 |
| 功能描述 | 针对非法的群组进行封禁 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：群组表。  2.在群组详情页面，点击封禁按钮，弹出封禁群组对话框  3，输入封禁群组原因，点击确认封禁按钮，讲该群组置为封禁状态，并自动发送站内消息给该群组的管理员 |
| 补充说明 |  |
| 功能4名称 | 群组解封功能 |
| 功能描述 | 解封群组 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：群组表。  2.在群组满足解封条件后，系统管理员在群组详情界面，点击解封群组，讲群组置为正常状态。 |
| 补充说明 |  |

#### 帖子类型管理

|  |  |
| --- | --- |
| 功能1名称 | 帖子类型列表页 |
| 功能描述 | 以分页形式展示帖子类型列表信息 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：帖子类型表。  2.在系统管理界面点击帖子类型管理按钮，进入帖子类型列表页  3.帖子类型列表页面分页展示帖子类型信息，右上角具备，增加，删除，两个按钮，点击分别进入帖子类型增加和删除页面  4.点击帖子类型名称超链接，进入帖子类型编辑页面 |
| 功能2名称 | 帖子类型添加页 |
| 功能描述 | 添加帖子类型信息 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：帖子类型表。  2.在帖子类型列表页，点击添加按钮，进入该界面。  3.输入帖子类型基本信息，点击保存，保存成功后跳转到帖子类型列表页面，保存失败后，重新跳转至帖子类型添加页面。 |
| 补充说明 |  |
| 功能3名称 | 帖子类型删除功能 |
| 功能描述 | 在帖子类型列表页面，选中要删除的帖子类型，点击删除按钮，删除帖子类型 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：帖子类型表。  2.选中要删除的一个或者多个帖子类型，点击删除按钮，删除成功，跳转至帖子类型列表页  3，如果一个帖子类型都没有选择，点击删除按钮，则提示，必须至少选择一个帖子类型。 |
| 补充说明 |  |
| 功能4名称 | 帖子类型编辑功能 |
| 功能描述 | 更改帖子类型基本信息 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：帖子类型表。  2.在帖子类型列表页，选择帖子类型名称，打开帖子类型编辑界面。  3.输入新的帖子类型信息，点击保存，保存成功跳转至帖子类型列表页。 |
| 补充说明 |  |

#### 帖子管理

|  |  |
| --- | --- |
| 功能1名称 | 帖子列表页 |
| 功能描述 | 以分页形式展示帖子列表信息 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：帖子表。  2.在系统管理界面点击帖子管理按钮，进入帖子列表页  3.列表页面分页展示帖子基本信息  4.点击帖子名称超链接，进入帖子详情页面 |
| 功能2名称 | 帖子详情页 |
| 功能描述 | 查看帖子详情信息 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：帖子表。  2.在地址列表页，点击帖子名称超链接，进入帖子详情界面。  3.帖子详情页面，显示帖子基本信息，显示该帖子回复列表，显示封禁按钮， |
| 补充说明 |  |
| 功能3名称 | 帖子封禁功能 |
| 功能描述 | 针对非法的帖子进行封禁 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：帖子表。  2.在帖子详情页面，点击封禁按钮，弹出封禁帖子对话框  3，输入封禁帖子原因，点击确认封禁按钮，讲该帖子置为封禁状态，并自动发送站内消息给该帖子的管理员 |
| 补充说明 |  |
| 功能4名称 | 帖子解封功能 |
| 功能描述 | 解封帖子 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：帖子表。  2.在帖子满足解封条件后，系统管理员在帖子详情界面，点击解封帖子，讲帖子置为正常状态。 |
| 补充说明 |  |

#### 广告管理

|  |  |
| --- | --- |
| 功能1名称 | 广告列表页 |
| 功能描述 | 以分页形式展示广告列表信息 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：广告表。  2.在系统管理界面点击广告管理按钮，进入广告列表页  3.广告列表页面分页展示广告信息，右上角具备，增加，删除，两个按钮，点击分别进入广告增加和删除页面  4.点击广告名称超链接，进入广告编辑页面 |
| 功能2名称 | 广告添加页 |
| 功能描述 | 添加广告信息 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：广告表。  2.在广告列表页，点击添加按钮，进入该界面。  3.输入广告基本信息，点击保存，保存成功后跳转到广告列表页面，保存失败后，重新跳转至广告添加页面。 |
| 补充说明 |  |
| 功能3名称 | 广告删除功能 |
| 功能描述 | 在广告列表页面，选中要删除的广告，点击删除按钮，删除广告 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：广告表。  2.选中要删除的一个或者多个广告，点击删除按钮，删除成功，跳转至广告列表页  3，如果一个广告都没有选择，点击删除按钮，则提示，必须至少选择一个广告。 |
| 补充说明 |  |
| 功能4名称 | 广告编辑功能 |
| 功能描述 | 更改广告基本信息 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：广告表。  2.在广告列表页，选择广告名称，打开广告编辑界面。  3.输入新的广告信息，点击保存，保存成功跳转至广告列表页。 |
| 补充说明 |  |

#### 系统参数维护

|  |  |
| --- | --- |
| 功能1名称 | 系统参数列表页 |
| 功能描述 | 以分页形式展示系统参数列表信息 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：系统参数表。  2.在系统管理界面点击系统参数管理按钮，进入系统参数列表页  3.系统参数列表页面分页展示系统参数信息，右上角具备，增加，删除，两个按钮，点击分别进入系统参数增加和删除页面  4.点击系统参数名称超链接，进入系统参数编辑页面 |
| 功能2名称 | 系统参数添加页 |
| 功能描述 | 添加系统参数信息 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：系统参数表。  2.在系统参数列表页，点击添加按钮，进入该界面。  3.输入系统参数基本信息，点击保存，保存成功后跳转到系统参数列表页面，保存失败后，重新跳转至系统参数添加页面。 |
| 补充说明 |  |
| 功能3名称 | 系统参数删除功能 |
| 功能描述 | 在系统参数列表页面，选中要删除的系统参数，点击删除按钮，删除系统参数 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：系统参数表。  2.选中要删除的一个或者多个系统参数，点击删除按钮，删除成功，跳转至系统参数列表页  3，如果一个系统参数都没有选择，点击删除按钮，则提示，必须至少选择一个系统参数。 |
| 补充说明 |  |
| 功能4名称 | 系统参数编辑功能 |
| 功能描述 | 更改系统参数基本信息 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：系统参数表。  2.在系统参数列表页，选择系统参数名称，打开系统参数编辑界面。  3.输入新的系统参数信息，点击保存，保存成功跳转至系统参数列表页。 |
| 补充说明 |  |

#### 日志管理

|  |  |
| --- | --- |
| 功能1名称 | 日志列表页 |
| 功能描述 | 以分页形式显示系统操作日志信息 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：日志表。  2.分页显示日志信息。  3.增加搜索框，可以根据日志内容关键字进行检索 |

### 前台模块设计（删除）

#### 首页（黑板报）

|  |  |
| --- | --- |
| 功能1名称 | 黑板报 |
| 功能描述 | 网站首页 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：群组表、帖子表。  2.该页面是网站首页  3.页面上面显示导航，右侧显示注册和登录按钮。  4.接下来显示4-6个新建立的群组  5.接下来显示4-6个新发布的帖子。 |
| 补充说明 |  |
| 功能2名称 | 群组详情页面 |
| 功能描述 | 点击首页的某个群组打开群组详情页面 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：群组表\_。  2.群组详情页面，显示群组基本信息，显示该群组成员列表。  3.分页显示帖子列表。  4.点击帖子标题超链接，进入帖子详情页面 |
| 补充说明 |  |
| 功能3名称 | 帖子详情页面 |
| 功能描述 | 在任何地方打开帖子，跳转到帖子详情页 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：帖子表\_。  2.显示帖子详细信息，包括标题，内容，发帖者，发帖日期，回复列表。  3.底部有回复帖子功能，登录用户输入回复内容即可。未登录用户需要首先登录才能进行回复。 |
| 补充说明 |  |

#### 校友志

|  |  |
| --- | --- |
| 功能1名称 | 校友志 |
| 功能描述 | 该页面即群组列表页面 |
| 数据结构与算法 | 1.该功能使用数据表结构包括：群组表\_。  2.默认按照热度（帖子发布数量）进行排序，显示群组列表，显示信息为：群组名称，学校，专业，所在城市，成员数目，以及最新的3个帖子。  3.群组以瀑布六形式展示，鼠标向下滚动后，自动刷新12条群组出来，每行4个，刷出来3行即可。  4.点击具体某个群组进入群组详情页。  5.点击具体某个帖子进入帖子详情页。 |
| 补充说明 |  |

#### 注册

|  |  |
| --- | --- |
| 功能1名称 | 注册 |
| 功能描述 | *提供注册页面以便用户注册* |
| 数据结构与算法 | 1.该功能使用数据表结构包括：用户表  2.打开注册页面  3.输入注册信息，包括登录名，密码，注册邮箱等。登录名必须唯一。  4.如果登录名不唯一，则提醒并允许用户重新注册。  5.注册成功后跳转至登录页面。 |
| 补充说明 |  |

#### 登录

|  |  |
| --- | --- |
| 功能1名称 | 登录 |
| 功能描述 | *系统登录功能* |
| 数据结构与算法 | 1.该功能使用数据表结构包括：用户表。  2.在用户需要进行登录操作的功能除都提供登录链接。  3.在登录界面输入用户名和密码，验证码后，提交登录。  4.登录成功后跳转至登录之前的页面  5.登录失败后，重新跳转至登录页面。 |
| 补充说明 |  |

#### 个人中心

|  |  |
| --- | --- |
| 功能1名称 | 我的群组 |
| 功能描述 | 列出自己加入的所有群组（原则上每个人只能加入一个群组） |
| 数据结构与算法 | 1.该功能使用数据表结构包括：群组表。  2.显示群组详情信息，显示退出按钮，提供退出群组功能  3.分页显示自己在该群组发的所有帖子。 |
| 补充说明 |  |
| 功能2名称 | 重置密码 |
| 功能描述 | *重置登录密码* |
| 数据结构与算法 | 1.该功能使用数据表结构包括：用户表。  2.在重置密码页面输入旧密码和新密码即可重置密码 |
| 补充说明 |  |
| 功能3名称 | 个人资料完善 |
| 功能描述 | *完善个人资料* |
| 数据结构与算法 | 1.该功能使用数据表结构包括：用户表  2.显示个人详细信息，并提供编辑功能。  3.限制对学籍信息的修改，如果修改学籍信息，则会自动退出之前加入的群组。 |
| 补充说明 |  |
| 功能4名称 | 消息中心 |
| 功能描述 | *显示自己收到所有的系统消息和好友消息* |
| 数据结构与算法 | 1.该功能使用数据表结构包括：消息表。  2.分页理出系统和好友发送的所有消息，消息内容包括消息内容，发送者，发送日期  3.可以回复每个消息。  4.可以删除消息。  5.可以发送一个消息。 |
| 补充说明 |  |

## 3.5数据需求（根据实体类来决定）

### 3.5.1用户信息

用户信息包括用户名，密码，昵称，真实姓名，Email，联系电话，QQ,所属学校，所属学院。

### 3.5.3学校信息

学校信息包括学校名称，地址。

### 3.5.3地址信息

地址信息包括地址编码，名称，经度，纬度。

### 3.5.4群组信息

群组信息包括群组名称，群组描述，创建者，床架日期，群组状态（审核中，活动群组，已解散，已关闭）。

### 3.5.5帖子信息

帖子信息包括标题，创建日期，发帖人、帖子类型。

## 3.7小结

通过对系统的整体把握，系统的大致框架和数据来源即最终的系统功能效果基本成型，为后期的系统实现绘制蓝图

# 4数据库设计

## 4.1数据库的概念

数据库（Database）是按照数据结构来组织、存储和管理数据的仓库，它产生于距今六十多年前，随着信息技术和市场的发展，特别是二十世纪九十年代以后，数据管理不再仅仅是存储和管理数据，而转变成用户所需要的各种数据管理的方式。数据库有很多种类型，从最简单的存储有各种数据的表格到能够进行海量数据存储的大型数据库系统都在各个方面得到了广泛的应用。

在信息化社会，充分有效地管理和利用各类信息资源，是进行科学研究和决策管理的前提条件。数据库技术是管理信息系统、办公自动化系统、决策支持系统等各类信息系统的核心部分，是进行科学研究和决策管理的重要技术手段。

## 4.2数据库的选择

本系统采用MySql搭建关系型数据库

MySQL是一个关系型数据库管理系统，由瑞典 MySQL AB 公司开发，目前属于 Oracle 公司。MySQL 是最流行的关系型数据库管理系统，在 WEB 应用方面 MySQL 是最好的 RDBMS (Relational Database Management System，关系数据库管理系统) 应用软件之一。MySQL 是一种关联数据库管理系统，关联数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性。MySQL 所使用的 SQL 语言是用于访问数据库的最常用标准化语言。MySQL 软件采用了双授权政策（本词条“授权政策”），它分为社区版和商业版，由于其体积小、速度快、总体拥有成本低，尤其是开放源码这一特点，一般中小型网站的开发都选择 MySQL 作为网站数据库。由于其社区版的性能卓越，搭配 PHP 和 Apache 可组成良好的开发环境。

与其他的大型数据库例如 Oracle、DB2、SQL Server等相比，MySQL 自有它的不足之处，但是这丝毫也没有减少它受欢迎的程度。对于一般的个人使用者和中小型企业来说，MySQL提供的功能已经绰绰有余，而且由于 MySQ L是开放源码软件，因此可以大大降低总体拥有成本。Linux作为操作系统，Apache和 Nginx作为 Web 服务器，MySQL 作为数据库，PHP/Perl/Python作为服务器端脚本解释器。由于这四个软件都是免费或开放源码软件（FLOSS)，因此使用这种方式不用花一分钱（除开人工成本）就可以建立起一个稳定、免费的网站系统，被业界称为“LAMP“或“LNMP”组合。

## 4.3数据库表的设计

### 4.3.1概念数据模型

简称概念模型(Conceptual Data Model)，是面向数据库用户的现实世界的模型，主要用来描述世界的概念化结构，它使数据库的设计人员在设计的初始阶段，摆脱计算机系统及DBMS的具体技术问题，集中精力分析数据以及数据之间的联系等，与具体的数据库管理系统（Database Management System，简称DBMS）无关。概念数据模型必须换成逻辑数据模型，才能在DBMS中实现。

其只描述信息的特征和强调语义，而不涉及信息在计算机中的表示，是现实世界到信息世界的第一层抽象。最常用的是实体联系模型（Entity Relationship Model)。

### 4.3.2物理数据模型

物理数据模型（Physical Data Model）PDM，提供了系统初始设计所需要的基础元素，以及相关元素之间的关系。即用于存储结构和访问机制的更高层描述，描述数据是如何在计算机中存储的，如何表达记录结构、记录顺序和访问路径等信息。使用物理数据模型，可以在系统层实现数据库。数据库的物理设计阶段必须在此基础上进行详细的后台设计，包括数据库的存储过程、操作、触发、视图和索引表等；至2013年，还没有太多的物理数据模型，最常用的物理数据模型如下：

统一模型。

框架存储模型。

### 4.3.3表格清单

* 考虑到数据库是在项目的开发中不断的完善的，所以就没有采用powerdesigner工具来设计数据库，后期可以在实体关系完全确定后用powerdesigner来设计数据库
* 这里数据库表的设计几乎都是和项目中的Java实体Bean对象一一对应的，所以以下表的内容会把数据库中的表和实体对象纳入到一起来对比显示
* 注：是否为空列项：(0表示不允许为空，-1允许为空）

使用pd生成的报告直接粘贴到这里即可。

# 5.系统开发要点

## 5.1系统开发环境及开发工具的使用及介绍

### 5.1.1eclipse

开源，轻便，可以以插件的方式方便加入和去除需要的插件功能

### 5.1.2mysql

开源，小巧，功能较齐全的数据库

### 5.1.3SVN

项目版本控制管理工具

Svn（Subversion）是近年来崛起的版本管理工具，在当前的开源项目里(J2EE)，几乎95%以上的项目都用到了 SVN。Subversion 项目的初衷是为了替换当年开源社区最为流行的版本控制软件 CVS，在 CVS的功能的基础上有很多的提升同时也能较好的解决 CVS 系统的一些不足。

### 5.1.4Maven

翻译为“专家”，“内行”

Maven是跨平台的项目管理工具。主要服务于基于Java平台的项目构建，依赖管理和项目信息管理。

### 5.1.5JDk1.7以上

### 5.1.6tomcat7.0以上

## 5.2系统开发相关技术

### 5.2.1Html

超文本标记语言或超文本链接标示语言（标准通用标记语言下的一个应用）HTML（HyperText Mark-up Language）是一种制作万维网页面的标准语言，是万维网浏览器使用的一种语言，它消除了不同计算机之间信息交流的障碍。

它是目前网络上应用最为广泛的语言，也是构成网页文档的主要语言。HTML文件是由HTML命令组成的描述性文本，HTML命令可以说明文字、图形、动画、声音、表格、链接等。HTML文件的结构包括头部（Head）、主体（Body）两大部分，其中头部描述浏览器所需的信息，而主体则包含所要说明的具体内容。

优点：打开速度快，操作简单，无需下载其他阅读器。

缺点：改变格式，无法直接引用，界面不友好，阅读不方便。

### 5.2.2Css

级联样式表是一种用来表现HTML（标准通用标记语言的一个应用）或XML（标准通用标记语言的一个子集）等文件样式的计算机语言。

CSS目前最新版本为CSS3，是能够真正做到网页表现与内容分离的一种样式设计语言。相对于传统HTML的表现而言，CSS能够对网页中的对象的位置排版进行像素级的精确控制，支持几乎所有的字体字号样式，拥有对网页对象和模型样式编辑的能力，并能够进行初步交互设计，是目前基于文本展示最优秀的表现设计语言。CSS能够根据不同使用者的理解能力，简化或者优化写法，针对各类人群，有较强的易读性。

### 5.2.3javascript

JavaScript一种直译式脚本语言，是一种动态类型、弱类型、基于原型的语言，内置支持类型。它的解释器被称为JavaScript引擎，为浏览器的一部分，广泛用于客户端的脚本语言，最早是在HTML（标准通用标记语言下的一个应用）网页上使用，用来给HTML网页增加动态功能。

在1995年时，由Netscape公司的Brendan Eich，在网景导航者浏览器上首次设计实现而成。因为Netscape与Sun合作，Netscape管理层希望它外观看起来像Java，因此取名为JavaScript。但实际上它的语法风格与Self及Scheme较为接近。

为了取得技术优势，微软推出了JScript，CEnvi推出ScriptEase，与JavaScript同样可在浏览器上运行。为了统一规格，因为JavaScript兼容于ECMA标准，因此也称为ECMAScript。

是一种解释性脚本语言（代码不进行预编译）。

主要用来向HTML（标准通用标记语言下的一个应用）页面添加交互行为。

可以直接嵌入HTML页面，但写成单独的js文件有利于结构和行为的分离。

跨平台特性，在绝大多数浏览器的支持下，可以在多种平台下运行（如Windows、Linux、Mac、Android、iOS等）。

### 5.2.4Servlet

[servlet](http://baike.baidu.com/view/25169.htm) 是在[服务器](http://baike.baidu.com/view/899.htm)上运行的小[程序](http://baike.baidu.com/view/17674.htm)。这个词是在 Java [applet](http://baike.baidu.com/view/150834.htm) 的环境中创造的。虽然后者已很少被使用，但 servlet 却发展的很好。是一般面试都会常考的知识

### 5.2.5Jsp

JSP全名为Java Server Pages，中文名叫java服务器页面，其根本是一个简化的Servlet设计，它[1] 是由Sun Microsystems公司倡导、许多公司参与一起建立的一种动态网页技术标准。JSP技术有点类似ASP技术，它是在传统的网页HTML（标准通用标记语言的子集）文件(\*.htm,\*.html)中插入Java程序段(Scriptlet)和JSP标记(tag)，从而形成JSP文件，后缀名为(\*.jsp)。 用JSP开发的Web应用是跨平台的，既能在Linux下运行，也能在其他操作系统上运行。

它实现了Html语法中的java扩张（以 <%, %>形式）。JSP与Servlet一样，是在服务器端执行的。通常返回给客户端的就是一个HTML文本，因此客户端只要有浏览器就能浏览。

JSP技术使用Java编程语言编写类XML的tags和scriptlets，来封装产生动态网页的处理逻辑。网页还能通过tags和scriptlets访问存在于服务端的资源的应用逻辑。JSP将网页逻辑与网页设计的显示分离，支持可重用的基于组件的设计，使基于Web的应用程序的开发变得迅速和容易。 JSP(JavaServer Pages)是一种动态页面技术，它的主要目的是将表示逻辑从Servlet中分离出来。

Java Servlet是JSP的技术基础，而且大型的Web应用程序的开发需要Java Servlet和JSP配合才能完成。JSP具备了Java技术的简单易用，完全的面向对象，具有平台无关性且安全可靠，主要面向因特网的所有特点。

### 5.2.6Mysql

MySQL是一个关系型数据库管理系统，由瑞典 MySQL AB 公司开发，目前属于 Oracle 公司。MySQL 是最流行的关系型数据库管理系统，在 WEB 应用方面 MySQL 是最好的 RDBMS (Relational Database Management System，关系数据库管理系统) 应用软件之一。MySQL 是一种关联数据库管理系统，关联数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性。MySQL 所使用的 SQL 语言是用于访问数据库的最常用标准化语言。MySQL 软件采用了双授权政策（本词条“授权政策”），它分为社区版和商业版，由于其体积小、速度快、总体拥有成本低，尤其是开放源码这一特点，一般中小型网站的开发都选择 MySQL 作为网站数据库。由于其社区版的性能卓越，搭配 PHP 和 Apache 可组成良好的开发环境。

### 5.2.7JQuery

#### 5.2.7.1 JQuery简介

Jquery是继prototype之后又一个优秀的Javascript库。它是轻量级的js库 ，它兼容CSS3，还兼容各种浏览器（IE 6.0+, FF 1.5+, Safari 2.0+, Opera 9.0+），jQuery2.0及后续版本将不再支持IE6/7/8浏览器。jQuery使用户能更方便地处理HTML（标准通用标记语言下的一个应用）、events、实现动画效果，并且方便地为网站提供AJAX交互。jQuery还有一个比较大的优势是，它的文档说明很全，而且各种应用也说得很详细，同时还有许多成熟的插件可供选择。jQuery能够使用户的html页面保持代码和html内容分离，也就是说，不用再在html里面插入一堆js来调用命令了，只需要定义id即可。

jQuery是一个兼容多浏览器的javascript库，核心理念是write less,do more(写得更少,做得更多)。jQuery在2006年1月由美国人John Resig在纽约的barcamp发布，吸引了来自世界各地的众多JavaScript高手加入，由Dave Methvin率领团队进行开发。如今，jQuery已经成为最流行的javascript库，在世界前10000个访问最多的网站中，有超过55%在使用jQuery。

jQuery是免费、开源的，使用MIT许可协议。jQuery的语法设计可以使开发者更加便捷，例如操作文档对象、选择DOM元素、制作动画效果、事件处理、使用Ajax以及其他功能。除此以外，jQuery提供API让开发者编写插件。其模块化的使用方式使开发者可以很轻松的开发出功能强大的静态或动态网页。

jQuery，顾名思议，也就是JavaScript和查询（Query），即是辅助JavaScript开发的库。

#### 5.2.7.2 特点

1. 动态特效
2. AJAX
3. 通过插件来扩展
4. 方便的工具 - 例如浏览器版本判断
5. 渐进增强
6. 链式调用
7. 多浏览器支持，支持Internet Explorer6.0+、Opera9.0+、Firefox2+、Safari2.0+、Chrome1.0+（在2.0.0中取消了对Internet Explorer6,7,8的支持）

### 5.2.8：Bootstrap

**Bootstrap简介**

Bootstrap，来自 Twitter，是目前最受欢迎的前端框架。Bootstrap 是基于 HTML、CSS、JAVASCRIPT 的，它简洁灵活，使得 Web 开发更加快捷。它由Twitter的设计师Mark Otto和Jacob Thornton合作开发，是一个CSS/HTML框架。Bootstrap提供了优雅的HTML和CSS规范，它即是由动态[CSS](http://baike.baidu.com/view/15916.htm)语言Less写成。Bootstrap一经推出后颇受欢迎，一直是GitHub上的热门开源项目，包括NASA的MSNBC（微软全国广播公司）的Breaking News都使用了该项目。

### 5.2.9：nutz

#### 5.2.9.1 nutz简介

Nutz是对于Java程序员来说，除SSH之外的另一个选择。当然，它是开源的，并且是完全免费的。同时也是商业友好的(Licensed under the Apache License, Version 2.0)。

#### 5.2.9.2特点

* **轻**

当前最新版，整个jar文件共910kB -- 针对JDBC的薄封装，无缓存。

* **全**

提供了Dao(ORM, SQL管理)，Ioc, Aop, Mvc, Json解析等必要功能。

* **活**

各个部分可以独立使用，比如在Spring里采用Nutz.Dao，又比如在Nutz.Ioc里使用Hibernate等。

* **整**

它所有功能均不依赖第三方jar文件。这就意味着：如果一个Web应用，你在WEB-INF/lib下只需要放置一个nutz.jar就够了。当然，你要使用其它的连接池、数据库驱动、打印PDF支持等功能，还需要自行添置jar包。

### 5.3小结

好的框架和技术如果很好利用的话，总能达到很好的效果，这里面的技术多多少少都用到过！

# 6系统实现

整个系统分为后台管理系统和前台系统。后台管理系统负责对系统的相关模块、数据进行统一的维护，由网站的管理员角色来完成。前台主要向相关用户展示功能页。

## 6.1后台管理系统

后台管理系统负责对系统的相关模块、数据进行统一的维护，由网站的管理员角色来完成。主要包括用户管理，学校管理，地址管理，群组管理，帖子管理，帖子类型管理。

### 6.1.1用户管理

#### 6.1.1.1用户列表页

* 设计思路

用户在后台管理页面点击“用户管理”链接，进入用户列表页面，页面展示系统总注册用户数，分页展示用户列表信息。

* 效果图



* 关键代码

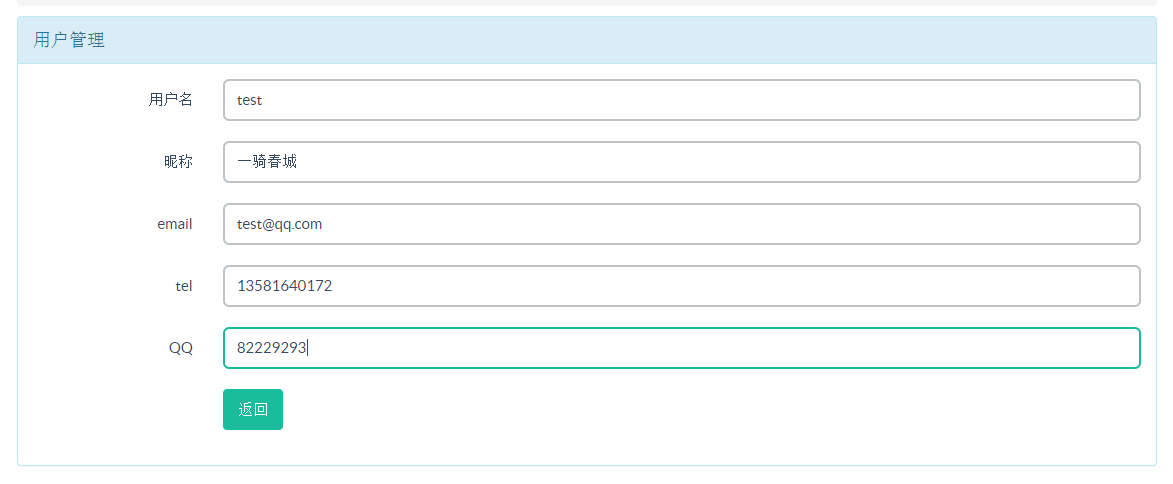
|  |
| --- |
| @At("/admin/user")  @Ok("jsp:/page/admin/user/user.jsp")  **public** **void** index(@Param("ps") Integer pageSize,  @Param("pn") Integer pageNumber, HttpServletRequest request) {  Pager pager = **new** Pager();  **if** (pageNumber == **null**) {  pageNumber = 1;  }  **if** (pageSize == **null**) {  pageSize = 10;  }  pager.setPageNumber(pageNumber);  pager.setPageSize(pageSize);  List<User> list = userService  .searchByPage(User.**class**, Cnd.*NEW*(), pager);  **int** count = userService.searchCount(User.**class**, Cnd.*NEW*());  pager.setRecordCount(count);  request.setAttribute("pager", pager);  request.setAttribute("list", list);  } |

#### 6.1.1.2用户详情页

* 设计思路

在用户列表页面，点击用户名称链接，进入用户详情页面，用户详情页根据用户id查询用户详细信息并展示在页面上

* 效果图



* 关键代码

|  |
| --- |
| 后台代码：  @At("/admin/userform")  @Ok("jsp:/page/admin/user/userForm.jsp")  **public** User form(@Param("id") Long id) {  User user = userService.find(id, User.**class**);  **return** user;  } |
| 前台代码：  <form class=*"form-horizontal"* role=*"form"* >  <input type=*"hidden"* name=*"user.id"* value=*"*${obj.id}*"*/>  <div class=*"form-group"*>  <label class=*"col-lg-2 control-label"*>用户名</label>  <div class=*"col-lg-10"*>  <input type=*"text"* name=*"user.uname"* class=*"form-control"* value=*"*${obj.uname}*"*>  </div>  </div>  <div class=*"form-group"*>  <label class=*"col-lg-2 control-label"*>昵称</label>  <div class=*"col-lg-10"*>  <input type=*"text"* name=*"user.nickName"* class=*"form-control"* value=*"*${obj.nickName}*"*>  </div>  </div>  <div class=*"form-group"*>  <label class=*"col-lg-2 control-label"*>email</label>  <div class=*"col-lg-10"*>  <input type=*"text"* name=*"user.email"* class=*"form-control"* value=*"*${obj.email}*"*>  </div>  </div>  <div class=*"form-group"*>  <label class=*"col-lg-2 control-label"*>tel</label>  <div class=*"col-lg-10"*>  <input type=*"text"* name=*"user.email"* class=*"form-control"* value=*"*${obj.tel}*"*>  </div>  </div>  <div class=*"form-group"*>  <label class=*"col-lg-2 control-label"*>QQ</label>  <div class=*"col-lg-10"*>  <input type=*"text"* name=*"user.email"* class=*"form-control"* value=*"*${obj.qq}*"*>  </div>  </div>  <div class=*"form-group"*>  <div class=*"col-lg-offset-2 col-lg-10"*>  <a href=*"*${ctx }*/admin/user"* class=*"btn btn-primary"*>返回</a>  </div>  </div>  </form> |

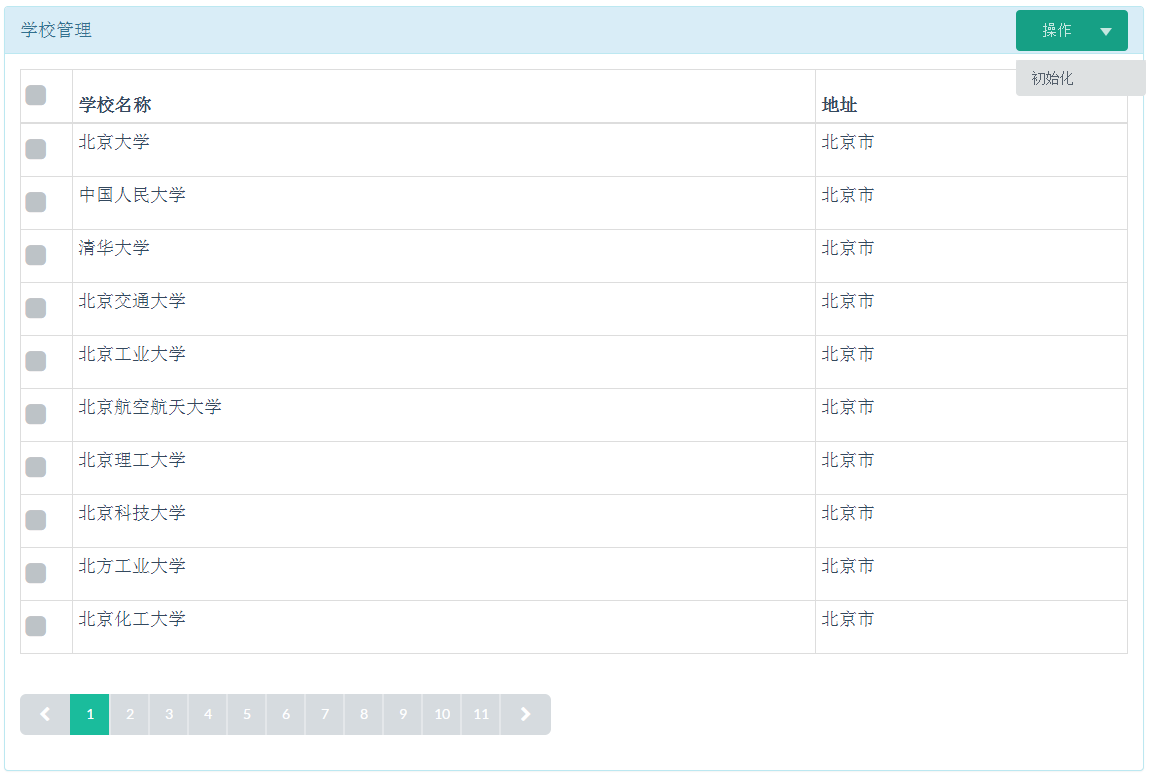
### 6.1.2学校管理

#### 6.1.2.1学校列表页

* 设计思路

在学校列表页，分页查询学校信息，并展示在页面，并提供了初始化学校功能。初始化的逻辑为：先清空学校表中的所有学校数据，然后从新浪教育频道讲全国几千所高效抓取下来并解析后存放到学校表中。

* 效果图



* 关键代码

采集学校使用到了开源的Jsoup组件，该组件可以方便的根据url从网上抓去数据，并提供了强大、易用的api对抓取到的页面文档进行解析、处理。

|  |
| --- |
| 初始化学校：  **public** **void** collect() {  // 清空学校表中的原有学校信息  **this**.clear();  String url = "http://edu.sina.com.cn/gaokao/2014-08-08/1903430534.shtml";  **try** {  Document doc = Jsoup.*connect*(url).timeout(100000).get();  Elements es = doc.getElementsByTag("tbody").get(0)  .getElementsByTag("a");  **for** (Element e : es) {  Document doc2 = Jsoup.*connect*(e.attr("href")).timeout(100000)  .get();  Elements es2 = doc2.getElementsByTag("tbody").get(0).children();  **for** (**int** i = 2; i < es2.size(); i++) {  Element e2 = es2.get(i);  Elements es3 = e2.children();  College c = **new** College();  c.setCname(es3.get(1).text().trim());  Region r = findByCondition(Region.**class**, "name", es3.get(3)  .text().trim());  c.setAddrCode(r != **null** ? r.getId() : **null**);  c.setAddr(es3.get(3).text().trim());  save(c);  }  }  } **catch** (Exception e2) {  e2.printStackTrace();  }  } |

### 6.1.3地址管理

#### 6.1.3.1省份列表页

* 设计思路

分页展示省份信息，包括省份的编码、名称，经度、纬度。

* 效果图



* 关键代码

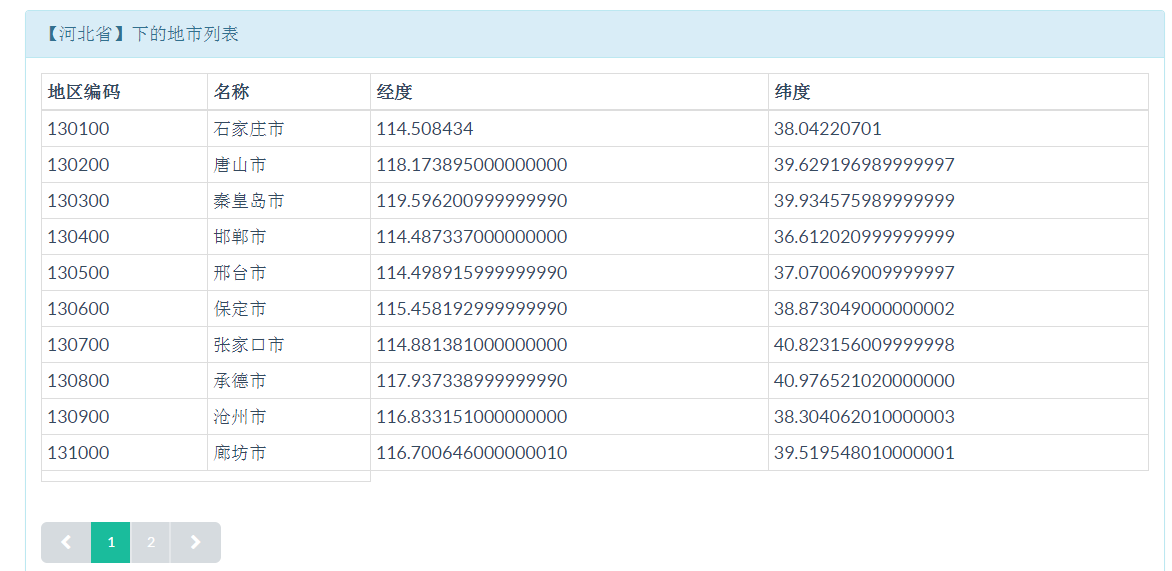
|  |
| --- |
| @At("/admin/region")  @Ok("jsp:/page/admin/region/province.jsp")  **public** **void** index(@Param("ps") Integer pageSize,  @Param("pn") Integer pageNumber, HttpServletRequest request) {  Pager pager = **new** Pager();  **if** (pageNumber == **null**) {  pageNumber = 1;  }  **if** (pageSize == **null**) {  pageSize = 10;  }  pager.setPageNumber(pageNumber);  pager.setPageSize(pageSize);  List<Region> list = regionService.searchByPage(Region.**class**,  Cnd.*where*("id", "like", "%0000"), pager);  **int** count = regionService.searchCount(Region.**class**,  Cnd.*where*("id", "like", "%0000"));  pager.setRecordCount(count);  request.setAttribute("pager", pager);  request.setAttribute("list", list);  } |

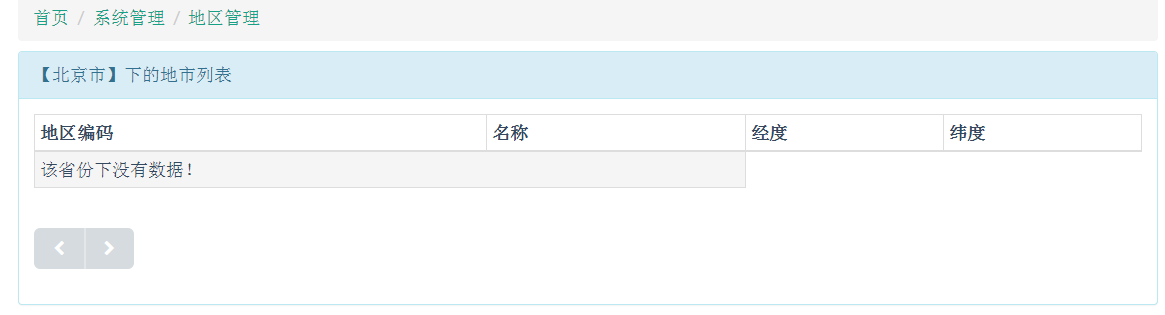
#### 6.1.3.1城市列表页

* 设计思路

在省份页面点击省份，进入城市列表，该页面根据根据省份编码查询该省份下所有城市列表并分页展示，如果该省份为直辖市，则列表中显示“该省份下没有数据！”

* 效果图





* 关键代码

|  |
| --- |
| @At("/admin/region/cities")  @Ok("jsp:/page/admin/region/cities.jsp")  **public** **void** index(@Param("ps") Integer pageSize,  @Param("pn") Integer pageNumber, @Param("pid") String provinceId,  HttpServletRequest request) {  Pager pager = **new** Pager();  **if** (pageNumber == **null**) {  pageNumber = 1;  }  **if** (pageSize == **null**) {  pageSize = 10;  }  pager.setPageNumber(pageNumber);  pager.setPageSize(pageSize);  List<Region> list = regionService.searchByPage(  Region.**class**,  Cnd.*where*("id", "like", provinceId.substring(0, 2) + "%").and(  "id", "<>", provinceId), pager);  **int** count = regionService.searchCount(  Region.**class**,  Cnd.*where*("id", "like", provinceId.substring(0, 2) + "%").and(  "id", "<>", provinceId));  pager.setRecordCount(count);  request.setAttribute("province",  regionService.find(Long.*valueOf*(provinceId), Region.**class**));  request.setAttribute("pid", provinceId);  request.setAttribute("pager", pager);  request.setAttribute("list", list);  } |

### 6.1.4群组管理

#### 6.1.4.1群组列表页

* 设计思路

分页展示群组列表，并提供群组的操作功能，包括针对新建群组的审核通过，解散群组，关闭群组功能。解散或者关闭群组的时候需要通知群组的创建者。并且如果群组的状态为非活动状态（审核中，已解散，已关闭），则前台不能展示该群组。

* 效果图



* 关键代码

|  |
| --- |
| /\*\*  \* 更新群组状态  \*  \* **@param** id  \* id  \* **@param** status  \* 要更改为的状态  \* **@param** statusMsg  \* 更改群组原因  \*/  **public** **boolean** updateGroupStatus(**long** id, **int** status, String statusMsg) {  Group entity = find(id, Group.**class**);  entity.setStatus(status);  **try** {  update(entity);  entity.getCreator();  MsgBox msg = **new** MsgBox();  msg.setSendUser("系统管理员");  msg.setDescript(statusMsg);  msg.setSendDate(**new** Date());  msg.setIfRead(MsgBox.***NO\_READ***);  msg.setReceiveUser(entity.getCreatorId());  save(msg);  **return** **true**;  } **catch** (Exception e) {  e.printStackTrace();  }  **return** **false**;  } |

#### 6.1.4.2群组详情页

* 设计思路

群组详情页和前台的群组相同，具体设计思路、效果图，关键代码请参考章节[6.2.3](#_6.2.3群组详情页)

### 6.1.5帖子类型管理

#### 6.1.5.1帖子类型列表页

* 设计思路

分页展示帖子类型信息，显示帖子的名称和描述

* 效果图



* 关键代码

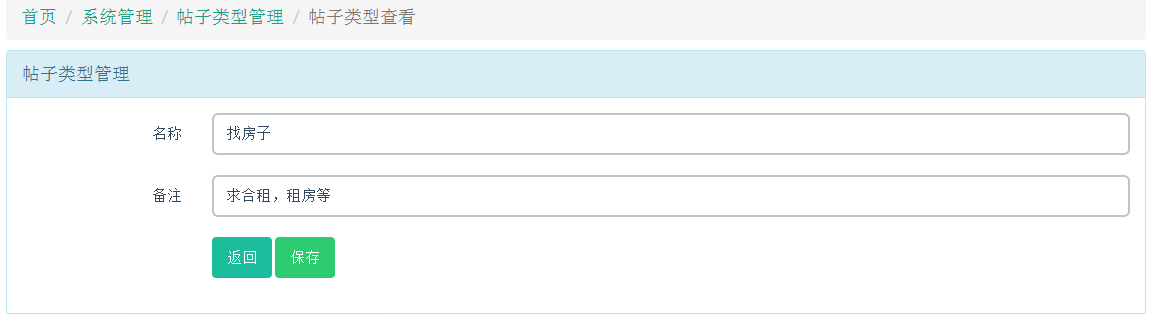
|  |
| --- |
| @At("/admin/msgtype")  @Ok("jsp:/page/admin/msgtype/msgtype.jsp")  **public** **void** index(@Param("ps") Integer pageSize,  @Param("pn") Integer pageNumber, HttpServletRequest request) {  Pager pager = **new** Pager();  **if** (pageNumber == **null**) {  pageNumber = 1;  }  **if** (pageSize == **null**) {  pageSize = 10;  }  pager.setPageNumber(pageNumber);  pager.setPageSize(pageSize);  List<MsgType> list = msgTypeService.searchByPage(MsgType.**class**,  Cnd.*NEW*(), pager);  **int** count = msgTypeService.searchCount(MsgType.**class**, Cnd.*NEW*());  pager.setRecordCount(count);  request.setAttribute("pager", pager);  request.setAttribute("list", list);  } |

#### 6.1.5.2帖子类型详情页

* 设计思路

在帖子列表页，点击帖子名称超链接，进入帖子详情页，该页面展示帖子的名称和描述信息，并可以进行编辑，并保存以更新帖子类型信息。

* 效果图

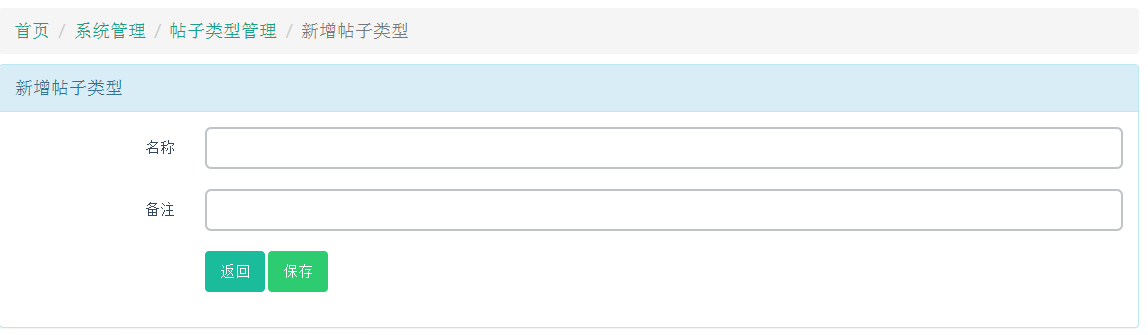


#### 6.1.5.3帖子类型添加页

* 设计思路

在帖子类型列表页，打开帖子类型添加页面，输入帖子名称和描述，点击保存按钮，保存一条帖子类型信息。

* 效果图



### 6.1.6帖子管理

#### 6.1.6.1帖子列表页

* 设计思路

分页显示帖子列表信息，并提供封帖功能。针对违反网站规定的帖子，对其进行封帖，封帖的时候需要输入封帖原因，并且发送一条消息给发帖者，告知其封帖原因，封的帖子。

* 效果图



* 关键代码

|  |
| --- |
| @At("/admin/message")  @Ok("jsp:/page/admin/message/message.jsp")  **public** **void** index(@Param("ps") Integer pageSize,  @Param("pn") Integer pageNumber, HttpServletRequest request) {  Pager pager = **new** Pager();  **if** (pageNumber == **null**) {  pageNumber = 1;  }  **if** (pageSize == **null**) {  pageSize = 10;  }  pager.setPageNumber(pageNumber);  pager.setPageSize(pageSize);  List<Message> list = messageService.searchByPage(Message.**class**,  Cnd.*where*("replayId", "=", 0), pager);  **int** count = messageService.searchCount(Message.**class**,  Cnd.*where*("replayId", "=", 0));  pager.setRecordCount(count);  request.setAttribute("pager", pager);  request.setAttribute("list", list);  } |

#### 6.1.6.1帖子详情页

* 设计思路

帖子详情页和前台的帖子详情页相同，具体设计思路、效果图，关键代码请参考章节[6.2.4](#_6.2.4帖子详情页)

## 6.2前台

网站前台即针对普通用户提供的页面功能，包括首页，群组列表页，群组详情页，帖子详情页，注册页，登录页，个人中心页。

### 6.2.1首页

### 6.2.2全部群组

### 6.2.3群组详情页

### 6.2.4帖子详情页

### 6.2.5注册页

### 6.2.6登录页

### 6.2.7个人中心

### 6.1.2帖子管理

## 6.3公共技术组件

### 6.3.1分页组件

### 6.3.2提示组件

### 6.3.2基础的服务类

主页信息显示区主要分为四大模块（删除）

最热群组：

根据帖子数量

最新群组：

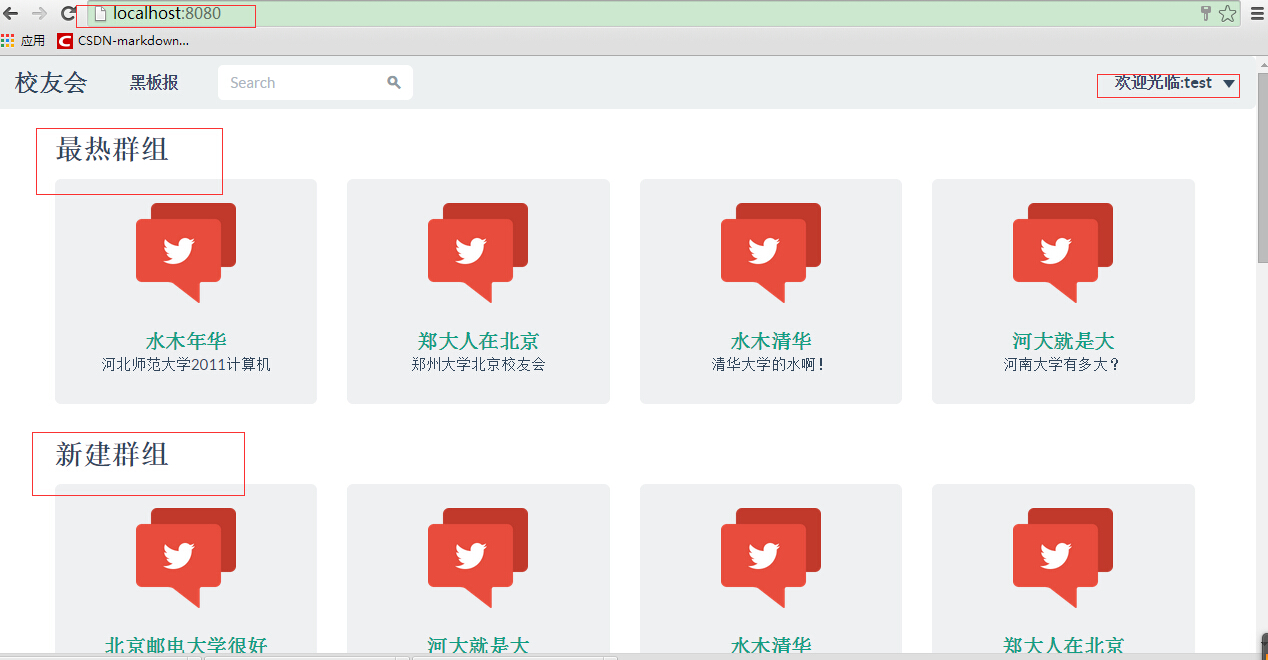
根据群组的创建日期

最近帖子列表：

根据帖子的创建日期

最热帖子列表：

根据帖子的回复量



点击进入某个群组后：可以看到群组详情及成员，最近的帖子信息

在这里，可以查看成员信息，回去帖子，加群，发帖，也可以选择，加载更多来显示更多群组



**群组的创建目的和帖子的内容没有限制**

**后期根据学习，生活，娱乐，等各个方面划分群组所在的范围，这样，方便后期用户查找所需要的群组**

**5.2.1:学习交流**

**5.2.2:生活交流**

**5.2.3....**

**吃喝玩乐**

**租房，求租，合租**

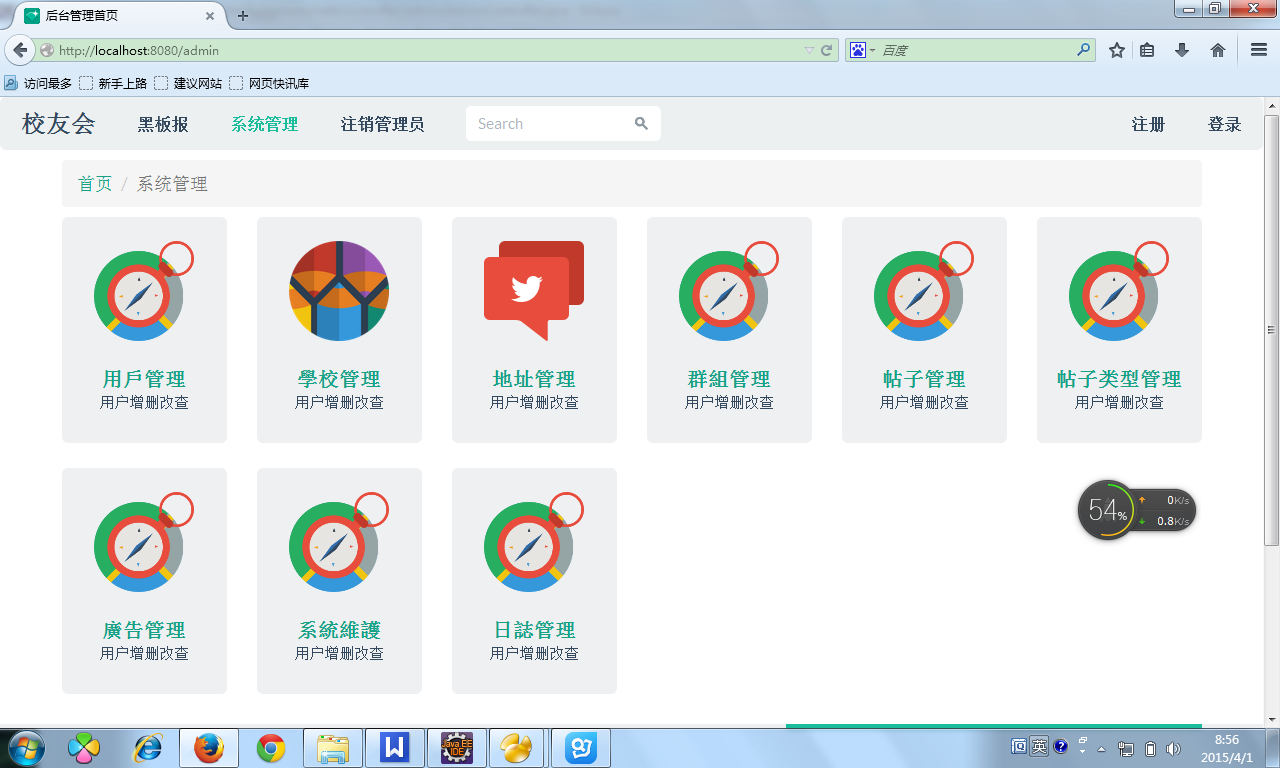
**找工作，发布，找**

## 5.3管理员管理用户页面设计（删除）

登录页面查acct\_user表，密码加密



登录成功后的界面：



此时可以点击注销管理员，也可以进行后台管理的相关操作

## 5.4用户登录之后根据不同身份有不同的操作权限设计（删除）

**普通管理员**

普通管理员只能做部分数据的查看，普通管理员至少有一个

**超级管理员**

超级管理员可以进行所有的增删该查，超级管理员有且又能有一个

# 7系统安装配置

参考《SysExpertIT运维监管系统安装配置维护手册V3.0(windows).doc》文档

# 结束语

# 参考文献

# 附录

# 致谢