

**毕 业 论 文**

**题 目 简易社区设计与实现**

**英文题目Design and Implementation of Simple Community**

**学生姓名**： **刘杰雨 申请学位门类**： **工学**

**学 号**： **201520180336**

**专 业**： **软件工程(嵌入式)**

**学 院**： **软件学院**

**指导教师**：  **王同罕**  **职称： 讲师**

**二 0 一 九 年 四 月 二 十 六 日**

**作 者 声 明**

本人郑重声明：所呈交的学位毕业设计（论文），是本人在王同罕老师的指导下，由本人独立完成的。没有剽窃、抄袭、造假等违反道德、学术规范和其他侵权行为。文中引用他人的文献、数据、图件、资料均已明确标注出，不包含他人成果及为获得东华理工大学或其他教育机构的学位或证书而使用过的材料。对本设计（论文）的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本毕业设计（论文）引起的法律结果完全由本人承担。

本毕业设计（论文）成果归东华理工大学所有。

特此声明。

毕业设计（论文）作者（签字）：

签字日期： 年 月 日

本人声明：该学位论文是本人指导学生完成的研究成果，已经审阅过论文的全部内容，并能够保证题目、关键词、摘要部分中英文内容的一致性和准确性。

学位论文指导教师签名：

年 月 日

**Design and Implementation of One-Card Inquiry System Based On RFID Technology**

**Jieyu Liu**

April 2019

# 摘 要

随着信息技术的飞速发展以及信息交流方式的改变，社区网站已经成为了一种新的沟通方式。

简易社区交流平台主要针对于小区的所有成员，一是将小区的管理一体化，以住户为中心，达到便捷、有序、高效的服务平台；二是增进小区成员之间的交流，在平台上可以相互交谈、意见表达、平等交流，增加相互之间的感情。

本文通过对部分小区的需求调研的基础上，根据调研结果做出了相对应的解决方案，通过选择相关适用的技术框架和显示界面样式，最终选定了两种框架，即前端layUI和后端Java的简易社区的设计与实现。

简易社区实现了简单、共用的方式，避免多套系统的复杂使用，以增进用户的体验感。

**关键词**： Spring； Mybatis； LayUI； 社区；

# ABSTRACT

With the rapid development of information technology and the change of information communication mode, people's communication mode has changed with the development of the Internet. Therefore, the significance of community website is to enhance the communication between people.

The simple community communication platform is mainly aimed at all members of the community. First, it integrates the management of the community and centers on residents to achieve a convenient, orderly and efficient service platform. The second is to enhance the communication between community members, in the platform can talk to each other, opinion expression, equal exchange, increase mutual feelings.

Based on the demand survey of some communities, this paper makes corresponding solutions according to the survey results, and then selects relevant and applicable technical framework and display interface style. Finally, the simple community design and implementation based on layUI and Java is selected.

Simple community to achieve a simple, common way to avoid the complex use of multiple systems, in order to enhance the user experience.

**Key words**: Spring；Mybatis； LayUI； Community;

**目 录**

[摘 要 1](#_Toc7028440)

[ABSTRACT 2](#_Toc7028441)

[第1章 绪论 1](#_Toc7028442)

[1.1 论文选题背景与意义 1](#_Toc7028443)

[1.2 研究现状与趋势 1](#_Toc7028444)

[1.3 研究方式 1](#_Toc7028445)

[1.3.1 设计思想 1](#_Toc7028446)

[1.3.2 技术简介 2](#_Toc7028447)

[1.4 本章小结 3](#_Toc7028448)

[第2章 模块以及功能设计介绍 4](#_Toc7028449)

[2.1 用户层级和模块介绍 4](#_Toc7028450)

[2.1.1 用户权限分级介绍 4](#_Toc7028451)

[2.1.2 模块介绍 4](#_Toc7028452)

[2.2 各模块及功能需求介绍 4](#_Toc7028453)

[2.2.1 交流模块 4](#_Toc7028454)

[2.2.2 管理模块 6](#_Toc7028455)

[2.2.3 超级管理模块 7](#_Toc7028456)

[2.3 运行环境介绍 8](#_Toc7028457)

[2.4 本章小结 8](#_Toc7028458)

[第3章 可行性分析 9](#_Toc7028459)

[3.1 市场可行性分析 9](#_Toc7028460)

[3.2 技术可行性分析 9](#_Toc7028461)

[3.3 经济可行性分析 9](#_Toc7028462)

[3.4 安全可行性分析 9](#_Toc7028463)

[3.5 社会可行性分析 9](#_Toc7028464)

[第4章 系统数据库设计 10](#_Toc7028465)

[4.1 需求分析 10](#_Toc7028466)

[4.2 概念结构设计 11](#_Toc7028467)

[4.3 逻辑结构设计 12](#_Toc7028468)

[4.4 物理结构设计 12](#_Toc7028469)

[4.5 运行和维护设计 12](#_Toc7028470)

[4.6 数据库表结构以及E-R图 12](#_Toc7028471)

[4.6.1 主要实体E-R图 12](#_Toc7028472)

[4.6.2 非主要实体类数据库表结构介绍 15](#_Toc7028473)

[4.7 本章小结 16](#_Toc7028474)

[第5章 技术及核心代码阐述 17](#_Toc7028475)

[5.1 Spring简介 17](#_Toc7028476)

[5.2 核心代码阐述 17](#_Toc7028477)

[5.3 本章小结 19](#_Toc7028478)

[第6章 系统测试 20](#_Toc7028479)

[6.1 测试目标 20](#_Toc7028480)

[6.2 预期测试结果 20](#_Toc7028481)

[6.3 测试出异常以及修复规则 20](#_Toc7028482)

[6.4 运行并记录测试结果 20](#_Toc7028483)

[6.4.1 交流模块测试结果 20](#_Toc7028484)

[6.4.2 管理模块测试结果 26](#_Toc7028485)

[6.4.3 超级管理模块测试结果 27](#_Toc7028486)

[6.5 本章小结 28](#_Toc7028487)

[总 结 29](#_Toc7028488)

[致 谢 30](#_Toc7028489)

[参考文献 31](#_Toc7028490)

# 第1章 绪论

## 1.1 论文选题背景与意义

在互联网飞速发展的今天，人与人之间的通过形式早已发生改变。人们对生活的需求已经从简单的物质追求慢慢的向着追求质量、功能和服务进行转变、并且对互联网有一定的依赖性。而社区网站[18,19]----就变成了人与人之间沟通的桥梁，并且会随着时代的发展不断的完善。它提供了自由的交流平台和便捷服务。

在这个交流的平台上，可以增进社区成员之间的关系，增进彼此的互动、互助以及相互体谅。这个平台目前是针对小区管理及住户之间的一个交流平台，每个社区成员都可以便捷的获取通知等告急信息，也可以提出针对性的建议和诉求。同时，遇见问题也能在上面寻找相对应的解决方法。

## 1.2 研究现状与趋势

在大多数国内互联网的平台上，社交作为互联网应用发展的必备要素，社区的发展较快，种类繁多，单从服务类型的社区也大可分为公共服务、博客显示、兴趣交流、商业社区等，无论哪种都是以服务大众为主旨；社区特色以及文化底蕴还有待提升。

社区网站作为互联网信息化的平台，通过社区网站[7]，认识更多志同道合的人；了解别人的想法、观点，同时可以帮助他人或者提出问题需要其他来解答；人们可以通过大部分的社交软件来阅读或者欣赏了解最新的娱乐八卦，看新闻资讯；社区网站越来越趋向于一种娱乐方式。

## 1.3 研究方式

### 1.3.1 设计思想

本次课题采用结构化生命周期作为系统开发的基本思想：

（1）系统开发准备阶段：

首先明确简易社区网站的需求以及针对使用人群。保证各方面的工作质量和今后各阶段开发的正确性，削减系统开发的盲目。

（2）调查研究及可行性分析阶段：

浏览当前大部分的社区网站，将体验结果数据统计，进行系统性的分析；去样本选中的小区进行调研，采集人群对简易社区网站的需求；综上两点，达到保留适应当前形式的需求，并加入新的需求。

（3）系统分析阶段：

将需求整理成功能文档，理清业务与数据流程的思路，将公用的流程提取出来，确定好最终的总体项目设计方案。

（4）系统设计阶段：

将最终的设计方案进行按层次、阶级、权限等进行功能的划分，将公用的功能额外的提取；对每个功能进行详细设计，数据库设计时，确定好数据字典，表字段名称以及含义，确认程序编码，选定开发工具，选择适用当前社区界面显示的框架，为一下开发阶段以及实施阶段做好准备。

（5）系统实施阶段：

确定实施计划和方案，录入测试数据，对系统各个模块进行调试，对主要模块进行抗压测试，遇见问题应立即修复，测试完毕后，应编写程序使用说明书。

### 1.3.2 技术简介

简易社区网站[10]用Java作为开发语言，采用MVC设计模式以及较为成熟的SpringMVC，Mybatis、Spring框架进行开发，使用Maven管理项目，Maven可以将一个项目拆分成多个工程，利于开发人员的工作分配，如果需要某些置顶的jar包，就直接引用依赖，不需要复制粘贴文件过来避免占用不必要的空间。

开发工具选用Intelli IDEA 工具，效率高并能提供便捷式的功能，数据库采用MySQL5.7，它能够处理大量数据并且开源，无需任何费用，相对于其他而言速度明显提升。

## 1.4 本章小结

本章主要是介绍课题来源的背景及意义，针对当前互联网社交平台的现状，提出相对应的发展趋势；并且将当前形势以及其未来的发展趋势作为本次课题的新需求，并提出研究课题的方式，介绍了开发项目所采用的技术进行介绍。

# 第2章 模块以及功能设计介绍

## 2.1 用户层级和模块介绍

### 2.1.1 用户权限分级介绍

因简易社区针对居住小区的所有成员，包括住户、小区管理员以及物业人员，所以在权限这一块，大致分为三种：

1）普通用户，这层用户属于社区基层，所以在数量上占绝大部分，简易社区的主旨是以服务群众为中心，拥有言论自由化的权利；在制作该整套社区网站都是以普通用户为基础尽心尽力打造最好的服务；

2）管理员一般是帮助超级管理员分担社区压力，在面对普通通用户的建议和要求时，可能一个管理用户忙不过来，这时就需要有人帮助分担压力，这样为了能够将普通用户的要求和建议能得到解决，也能缓解管理层的压力

3）超级管理员拥有整个社区最高的权限，管理整个社区，可选中部分普通用户帮忙分担自己的压力。

### 2.1.2 模块介绍

根据不同层级的用户，所拥有权限也不一样，共分为三个界面模块：交流模块；普通管理；超级管理。每一个模块都有相对应的功能，以交流模块为主，其他模块为次，进行分配工作。

1）交流模块：

基层用户使用的模块，因该模块的用户数量基数大，所以在这个模块上花费的时间百分比最高，功能也相对应的较为齐全；功能主要包括：可浏览其他用户发布的帖子、文章，既能评论留言，交流心得，也能收藏、保证帖子不会被其他帖子挤下来，能看到本周最活跃的用户，最火的文章，也能获取管理员发布的相关公告及告急信息，发布帖子等其他快捷按钮。也能对自己在社区的信息作出相对应的修改，管理自己发布收藏的帖子以及收到别人评论与留言。

2）普通管理模块：

管理员相对于普通用户多的一个权限模块，因时间都花费在交流模块上面，所以该模块以及下一个模块的功能相对而言部分功能缺失，该模块仅包括：管理需要置顶以及较为精炼的帖子或文章；帮助超级管理员发布小区公告等其他较为重要的信息。

3）超级管理模块：

超级管理员：顾名思义、具有在社区中最高权限的成员，但一般都是社区管理高层；比管理员多出一个超级管理模块，功能包括：修改用户的权限，新增、修改帖子或文章所属类型的名称。

## 2.2 各模块及功能需求介绍

### 2.2.1 交流模块

该社区网站将交流模块作为访问的首页，采用了LayUI经典化框架作为主要前端页面，显示效果能达到一面极简、一面丰盈；极简是是视觉的所在的外在，丰盈是倾情雕琢的内在，简而重，以此达到双面体验的效果，更加的平易近人；简易社区以怪异文字的形式做为logo标签，logo主要的设计缘由是用一个独特视觉符号代替社区品牌，这样既直观又具有比较高的识别性，单击logo图标即可返回主界面，不管是已登录或未登录都能浏览页面，页面能看到的按钮和链接都能使用和访问，并且不包含任何广告和恶意链接，以这种简洁的形式突显出社区的氛围，让浏览的游客感受社区带来的舒适感。



图 2.2.1.1 首页展示

首页主要共划分六个小版块：快捷按钮、置顶、所有帖子、公告、回帖周榜以及本周热议；快捷按钮每个都有相对应功能，既方便快速使用，又能不占界面显示面积，也不会影响美观；每个版块都显示相对应的数据，以所有帖子的版块为主，其他为次；单击相对应的帖子或文章都可以查询帖子或文章的详细信息，单击头像便可看到这条帖子的发帖人的详细信息。

对于详细信息页面，不仅能够看到对应的帖子信息，比如：发帖人、发布时间、头像、帖子类型等，而且在界面顶端和右侧保留了大部分快捷按钮，但还是以帖子详细信息为主，如果该帖子的发布人是当前用户，便可增加请求置顶和请求加精的特殊按钮，一旦管理员通过了请求便可在首页的置顶版块显示您的帖子，如果觉得自己发布的这条帖子觉得不太好，我们提供了删除按钮；查看时，在一定时间内停留该帖子便能记录阅读次数，在显示详细信息的偏上部，而右侧以小图标的形式，更能贴近当前市面上的视觉显示效果，单击小图标可收藏（注：保证文章不会因刷新而流失）、可评论（注：说出对当前文章的想法以及建议）快速跳转到评论框；在评论框说出您的想法并提交上去，发帖人便可收到消息并给予回复，在评论区域，如果存在恶意评论或攻击，即可删除这条评论。



图 2.2.1.2 详细信息展示

单击右上角即可登录与注册，注册时，按照简易社区的宗旨，并不需要填写那么多的信息，如若在登录时忘记密码了，这是不要慌张，作为社区最基本的要求，我们开启了通过邮箱找回密码的功能（目前仅仅支持QQ邮箱找回），我们将重置新的密码发送至您的邮箱（如若失败请联系管理员）；单击上方头像便可设置自己信息和退出当前账号，而在右下角便能快捷的进入到尕布帖子的界面。

进入到信息管理界面，又分为四大板块：我的主页、用户中心、基本设置、我的消息；主页包括发布帖子全部信息、回复信息以及个人信息的展示；用户中心含有发布帖子以及收藏的帖子，之前浏览文章或帖子收藏，这时会在这个版块显示；基本设置含有个人信息填写、换个别具一格的头像、怕密码泄露设置新密码的功能；消息管理便可收到回复帖子以及@你的消息，这个版块需要实时更新，才能达到时时提醒的功能。

发布帖子的文本输入框，使用了LayUI的富文本，支持表情、超链接、粗体、斜体等多种对字体的操作。

以上便是交流模块基本的功能所在，因这个模块使用的对象的人群是社区的大多数成员，所以在功能的使用上会更加的舒适，并且对后台的代码开发也是比较严谨。

### 2.2.2 管理模块

该模块主要功能偏向于管理，所以只有管理员用户才能访问，普通用户是看不到该页面的，因该社区网站主旨偏向于服务群众，所以在管理页面部分功能有些欠缺，但最基本的功能还是可以使用的。



图 2.2.2.1 普通管理展示

本模块主要功能共分为三个：管理置顶、管理加精以及发布公告。

管理置顶：功能主要是将用户需要请求置顶的帖子或文章进行审批，是否能置顶，让其他用户看到，这这个版块我们添加了快速查询某个范围内的帖子，这样可以达到快速、高效操作，减少不必要的时间；

管理加精的功能大致与管理置顶模块是一样的，加精模块只不过是将帖子设置为精贴；

发布公告将小区的一些重要、紧急的事情在社区里面发布一下，在首页的公告版块中能看到发布的信息。

### 2.2.3 超级管理模块

该模块仅属于超级管理员使用，能管理所有帖子的种类，对其能进行修改、新增；修改用户的权限，能快速查询到某些用户，提升工作效率。



图 2.2.3.1 超级模块管理展示

## 2.3 运行环境介绍

本套社区系统隶属于web应用系统，兼容手机终端浏览器，使用IDEA进行开发，IntelliJ在业界被公认为便捷、高效的java程序开发工具(现在流行使用eclipse、myeclipse和IDEA)，尤其在代码自动提示等方面的功能可以说是比较好用的；Java语言是Sun公司开发的，如今已经成为普遍存在的技术，相对啊语言市场占比率大，在业界也有较为丰富的职员可使用；数据库使用的是Sun公司开源的MySQL数据库5.7版本，该数据库大量数据时查询速度是业界较好的，也能保障数据的安全性；而中间件精简而强悍的Apache tomcat8.5版本。

## 2.4 本章小结

本章较为详细的介绍了整套简易社区网站系统的各个模块、功能，阐述了不同社区成员的权限所使用的模块各个功能的特点，；也谈到系统运行的环境，对使用到的工具进行一个简洁的介绍。

# 第3章 可行性分析

## 3.1 市场可行性分析

简易社区针对的市场坐落于城市住宅区，根据对城市中的部分居住小区的调研结果，发现大部分的社区的公告栏上贴满了各种广告、通知等等；这样不仅通知不到位而且，一旦错过了什么重要的事情，会增加住户与小区管理员的误会；所以简易社区针对的人群还是有较大的市场的。

## 3.2 技术可行性分析

在使用的技术上使用目前较为主流的的ssm框架，性能相对于其他框架有一定提升，在开发，运维都有着较高的效率，使用Java开发在安全性上面也有一定的保障，使用tomcat作为中间件能承受较多的用户对系统进行同时操作和使用，数据库采用MySQL能承受百万级数据的涌入；在社区网站的界面使用及借鉴了LayUI框架，简约而丰富，让用户能感受到两种不同的体验。

## 3.3 经济可行性分析

简易社区服务于小区的住户以及管理员，能够提升小区管理的效率，也能减少住户与管理员之间误会，使用MySQL数据库进行数据储存以及加上开发工具IDEA，在成本上面虽然有一定的开销，但是一旦将整套系统开发出来后，不仅仅能够让一个小区进行使用，拓展一下市场范围，一旦普及多个城市，获得到收益折算成本后，也有一定的可观。

## 3.4 安全可行性分析

在个人信息的保障，使用md5加盐加密传输，能保障每个人信息不被盗取，再加上重置密码能关联QQ邮箱，在QQ邮箱这个腾讯保护下面，很难盗取用户个人信息。

## 3.5 社会可行性分析

简易社区制作的初衷是服务于群众，参考当前大部分社区，总结其他社区的利弊，精心制作而成，该社区不仅能增加服务质量，得到用户的好评，也可以推动其他类型社区的发展。

# 第4章 系统数据库设计

## 4.1 需求分析

在设计数据库之前，应先做好对系统的需求分析，因为数据库可以说是系统之根本，修改数据库就会对系统造成大小不一的影响；所以应该弄清楚了每个模块、每个功能的流程、种类、范围、数量以及数据交互中会出现可能的情况。确定好系统对数据库的使用要求以及约束，避免在后面的开发过程中，倒回来再去修改数据库，这样不仅低效，而且制作出来的系统也不太稳定。

总结之前的调研结果，设计以及确定好所有实体、实体的属性以及实体之间的联系，并绘制数据流图：



图4.1 数据流图

绘制好数据流图后，将系统数据流程图绘制，可以让开发人员更好的理解功能之间的流程，如下图4.2所示：



图4.2 系统整体数据流程图

## 4.2 概念结构设计

概念结构设计就是以需求分析为基础，抽离出用户需求，转换成信息结构的一个过程，也称为概念模型。

它具有能清楚表达出需求，方便交流和理解，能随时修改和扩展，很容易向其他数据模型转换并且易使开发人员记住整体流程等特点。

经常用到的一种概念模型就是E-R模型，它将现实世界的信息结构统一以绘制不同图形来描述它们之间的联系，不同的图形分别代表不同的含义。

简易社区设计的数据库概念模型用的是自底向下的方法，先将每个实体的概念结构以绘制E-R图的形式将它描述出来，然后将它们集成系统整体E-R结构图，根据实体与实体之间的联系，逐步生成其他的概念结构，直至总体概念结构。

## 4.3 逻辑结构设计

逻辑结构设计就是将概念结构设计转换为DBMS支持的数据模型[4]，一般使用的模型为关系模型，将实体的属性以及实体与实体之间进行联系，然后可以根据需求或其他的考虑，建立相对应的视图、存储过程和函数等等。

## 4.4 物理结构设计

根据DBMS特点和处理的需要，进行物理存储安排，建立索引，形成数据库内模式[8.13]。在简易社区数据库的物理结构设计中，采用了聚集和索引，不但节省了存储空间还提高了查询效率，同时还确保了数据的完整性。

## 4.5 运行和维护设计

数据库的运行和维护主要任务是：维护数据库的安全性与完整性、监测并改善数据库性能、重新组织和构造数据库[14]。

## 4.6 数据库表结构以及E-R图

### 4.6.1 主要实体E-R图

（1）用户基本信息表（唯一ID，邮箱，密码，昵称，城市，性别，签名，头像名称，权限）：

表4-1 表名

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| v\_user | | | | |
| 名称 | 表字段名称 | 字段类型 | 键 | 备注 |
| 唯一表示ID | u\_id | nvarchar2(32) | 主键 | 使用UUID获取 |
| 邮箱 | email | nvarchar2(20) |  | 登录账户(唯一的) |
| 密码 | upwd | nvarchar2(32) |  | 可用Md5加密 |
| 昵称 | username | nvarchar2(20) |  | 网名 |
| 城市 | city | nvarchar2(20) |  | 可以搜同城 |
| 性别 | ugender | number(1) |  | 0是男，1是女 |
| 签名 | usignature | nvarchar2(100) |  |  |
| 头像名称 | uimage | nvarchar2(32) |  |  |
| 权限 | urole | number(1) | 外键 | 0普通,1 管理员,2超级管理员 |

用户表的E-R图：



图4.3 用户E-R图

（2）帖子信息表(唯一ID，发布人邮箱，标题，内容，发布时间，评论数量，点赞数量，浏览次数，帖子类型，帖子所属模块，删除状态，精贴状态)：

表4-2 表名

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **v\_article** | | | | |
| **名称** | 表字段名 | 字段类型 | 键 | 备注 |
| **帖子唯一ID** | aid | nvarchar2(32) | 主键 | 唯一标识id |
| **发布人的邮箱** | email | nvarchar2(20) |  |  |
| **标题** | atitle | nvarchar2(30) |  |  |
| **内容** | acontent | nvarchar2(2000) |  |  |
| **发布时间** | currtetime | date |  |  |
| **评论的数量** | aspnum | number(8) |  |  |
| **点赞的数量** | adznum | number(8) |  |  |
| **浏览次数** | anum | number(8) |  |  |
| **帖子的类型** | atype | number(1) |  | 0是普通，1是置顶，2是公告 |
| **贴子的所属模块** | acolumn | number(3) | 外键 | 请看模块表 |
| **是否被删除** | Isdelete | number(1) |  | 0是存在，1是删除 |
| **是否为精贴** | Isfine | number(1) |  | 0为普通，1是精贴 |

帖子E-R图：



图4.4 帖子E-R图

（3）帖子模块表(模块ID，模块名称)：

表4-3 表名

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **v\_module** | | | | |
| **名称** | 表字段名 | 字段类型 | 键 | 备注 |
| **ID** | mid | number(3) | 主键 |  |
| **模块名称** | mname | nvarchar2(10) |  | 帖子所属模块名称 |

帖子模块E-R图：



图4.5 帖子模块E-R图

（4）将所有实体类的E-R图汇集到一起，构成了整体E-R图，如下：



图4.6 系统整体E-R图

### 4.6.2 非主要实体类数据库表结构介绍

（1）回复评论信息表：

回复评论表（ID，回复帖子ID，回复人邮箱，被回复邮箱，回复内容，时间，被回复人未已读，点赞数量，楼主未已读，删除状态）：

表4-4 表名

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| v\_atrinfo | | | | |
| 名称 | 表字段名 | 字段类型 | 键 | 备注 |
| ID | aiid | nvarchar2(32) | 主键 | 唯一ID |
| 回复贴子的ID | Aid | nvarchar2(32) | 外键 |  |
| 回复人邮箱 | Replyemail | nvarchar2(20) |  |  |
| 被回复人邮箱 | Ireplyemail | nvarchar2(20) |  |  |
| 回复内容 | Replycontent | nvarchar2(1000) |  |  |
| 回复时间 | replytime | date |  |  |
| 被回复人是否已读 | Replyisread | number(1) |  | 0是未读，1是已读,2为删除 |
| 点赞的数量 | Replythumbs | number(1) |  | 点赞数量 |
| 楼主是否已读 | Lordisread | number(1) |  | 0是未读，1是已读，2是已删除 |
| 回复是否被删除 | isdelete | number(1) |  | 0是存在,1是删除 |

（2）帖子点赞表(唯一ID，点赞的帖子，点赞的回复ID，点赞人邮箱)：

表4-5 表名

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| v\_tags | | | | |
| 名称 | 表字段名 | 字段类型 | 键 | 备注 |
| ID | tagid | nvarchar2(32) | 主键 |  |
| 点赞的帖子 | Aid | nvarchar2(32) | 外键 |  |
| 点赞的回复ID | aiid | nvarchar2(32) | 外键 |  |
| 点赞人邮箱 | email | nvarchar2(20) |  |  |

（3）帖子收藏表：

表4-6 表名

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| v\_collect | | | | |
| 名称 | 表字段名 | 字段类型 | 键 | 备注 |
| ID | vcid | nvarchar2(32) | 主键 |  |
| 收藏帖子id | aid | nvarchar2(32) | 外键 |  |
| 收藏人邮箱 | email | nvarchar2(20) |  |  |
| 是否删除 | Iddelete | number(1) |  | 0为存在，1是删除 |

（4）置顶精贴表：

表4-7 表名

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| v\_porary | | | | |
| 名称 | 表字段名 | 字段类型 | 键 | 备注 |
| ID | Pid | nvarchar2(32) | 主键 |  |
| 请求帖子ID | Aid | nvarchar2(32) | 外键 |  |
| 请求格式 | Format | number(1) |  | 0是请求精贴，1是请求置顶 |
| 是否通过 | Isadopt | number(1) |  | 0是未通过，1是已通过 |
| 请求时间 | Createtime | date |  |  |

## 4.7 本章小结

本章主要介绍了简易社区底层数据库设计，按照规范的数据库设计步骤，首先在需求分析里面，应该确定好用户的需求，需要细致到每个功能，这样才能打好整套系统的基础，并且在数据库设计表这一块，可以根据需求在需要的表建立相对应的索引、存储过程等数据库提供快速查询的方法。

绘制好E-R图更能直观表现系统的主体功能以及实体与实体之间联系，便于开发者更好的理清除思路。这样不仅仅提升开发效率，一旦在开发后期出现新的需求，需要改动数据库表结构、迁移数据库或者优化的时候，开发人员将在维护、优化方面省心省力。

# 第5章 技术及核心代码阐述

## 5.1 Spring简介

Spring框架[6,12]是由于软件开发过于复杂而创建的一种轻量级框架，便捷程序开发，并且使用方便，并且大部分都是以注解的形式将类注入到spring容器。然而，Spring的用途不仅仅局限于java服务器端的开发。从简单性、可测试性和松耦合性角度而言，市场是大部分的Java web项目都可以从Spring中受益；所包含的特点总结如下四点：

1.方便解耦，简化开发

通过Spring提供的IoC容器，我们可以将对象之间的依赖关系交由Spring进行统一管理，避免编写程序所造成的过度程序耦合，这样能够减少冗余代码的书写。有了Spring，用户不必再为单实例模式类、属性文件解析等这些很底层的需求编写代码，也就是说没必要太关注于整套系统的结构，可以更专注于在业务逻辑代码上的开发，提升开发效率。

2.AOP编程的支持

通过Spring提供的AOP功能，方便进行面向切面的编程，因为AOP功能开启了，程序所执行的某个方式，便一定能走到AOP类，这时可以进行记录程序日志等等，而且许多用传统OOP实现不了的功能可以通过AOP实现。

3.声明事务的支持

在Spring中，对事务管理SPI也能依旧使用，也可以描述事务的隔离级别，通过注解声明式方式灵活地进行事务的管理，提高开发效率和质量。

4.方便集成各种优秀框架

Spring不排斥各种优秀的开源框架，包括spring自己的衍生框架，并且Spring官方也做出对每种框架相对应的适配jar包，相反，Spring可以降低各种框架的使用难度，Spring提供了对各种优秀框架（如Struts,Hibernate，Mybatis）、NoSql数据库[17]的直接支持。

## 5.2 核心代码阐述

介绍完主要使用的框架后，可以谈谈简易社区的业务模块，以及部分功能的代码介绍：

（1）帖子评论：

在之前的数据库设计当中我们能了解到，在你评论这一块，并没有像市场是大部分的论坛之类楼中楼形式，该系统使用@的形式展现在我们面前，因为简易社区有消息提醒的功能，所以没必要有楼中楼的形式，这也是简易社区的特色所在。

回复包括两种，一种是直接回复楼主，第二种是@其他人：

核心代码：

V\_atrInfo atrInfo=new V\_atrInfo();  
atrInfo.setAiid(UUid.getUUID());  
atrInfo.setAid(a\_id);  
atrInfo.setReplyemail(reply\_id);  
if(ireply\_id!=null && !ireply\_id.equals(""))  
 atrInfo.setIreplyemail(ireply\_id);  
atrInfo.setReplycontent(content);  
atrInfo.setReplyisread(0);  
atrInfo.setReplythumbs(0);  
if(articleService.insertAtrInfoContent(atrInfo) && articleService.writeComment(a\_id))  
 return "success";  
else  
 return "fail";

（2）详细信息按钮及小图标：

虽然查看信息不难，但是对上方的功能按钮权限控制还是相对来说比较困难的，前端利用jstl标签库进行控制，下列代码中有部分代码有注释，以便于理解：

核心代码：  
ModelAndView mv=new ModelAndView();  
ArticleDetail detail=articleService.getArticleDetail(aid);  
V\_UserInfo userInfo=(V\_UserInfo) session.getAttribute("userInfo");  
  
List<Map<String,Object>> list;  
if(userInfo==null)  
 list=null;  
else  
 list=articleService.selectCollects(userInfo.getEmail());  
//遍历当前帖子是否为当前登录用户收藏  
boolean isContain=false;  
Object isDelete=-1;  
if(list!=null) {  
 for (Map<String, Object> map : list) {  
 if (map.get("AID").equals(detail.getAid())) {  
 isContain = true;  
 isDelete = map.get("ISDELETE");  
 break;  
 }  
 }  
}  
mv.addObject("detail",detail);  
mv.addObject("isContain",isContain);  
mv.addObject("isDelete",isDelete);  
mv.setViewName("/html/jie/detail");  
return mv;

（3）找回密码：

找回密码采用QQ邮箱的形式进行发送重置后的密码，因需要使用腾讯QQ邮箱，所以相对来说比较复杂，所花费的时间也比较长；因为是类似的，所以写个了SpringBoot版本的QQ邮箱发送信息的webservice，放置在远程服务器上，任何人想要是要QQ邮箱发送信息，直接调用该端口即可：

//创建MIME邮件对象  
mimeMsg=new MimeMessage(mailSession);  
String name=MimeUtility.encodeText(FormName);  
mimeMsg.setFrom(new InternetAddress(name+"<"+username+">"));  
mimeMsg.setRecipients(Message.RecipientType.TO, InternetAddress.parse(MailTo));  
//设置邮件主题  
mimeMsg.setSubject(Subject,"utf-8");  
//设置邮件内容，将邮件body部分转化为HTML格式  
mimeMsg.setContent(Body,"text/html;charset=utf-8");  
//发送邮件  
try{  
 Transport.send(mimeMsg);  
 return true;  
}catch(Exception e){  
 return false;  
}

创建好Utils类后，在找回密码的方法中直接请求该链接便能返回相对的值，url: <http://47.102.207.63:8086/SendMessage>这个接口是我在远程服务器上部署的一个springboot的jar包，带上对应参数便能请求发送QQ邮件了：

URL url = new URL("http://47.102.207.63:8086/SendMessage"+param);  
URLConnection urlConnection = url.openConnection();  
urlConnection.setRequestProperty("accept", "\*/\*");  
urlConnection.setRequestProperty("connection", "Keep-Alive");  
urlConnection.setRequestProperty("user-agent", "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1;SV1)");  
urlConnection.connect();  
//System.out.println("返回的内容"+urlConnection.getContent());  
Map<String, List<String>> map = urlConnection.getHeaderFields();  
// 遍历所有的响应头字段，获取到cookies等  
for (String key : map.keySet()) {  
 System.out.println(key + "--->" + map.get(key));  
}  
// 定义 BufferedReader输入流来读取URL的响应  
read = new BufferedReader(new InputStreamReader(urlConnection.getInputStream(), "UTF-8"));  
String line;//循环读取  
while ((line = read.readLine()) != null) {  
 result += line;  
}

return (Map)JSON.parse(result);

简易社区这套系统难点已介绍完毕，部分功能的核心代码也一并展示了。

## 5.3 本章小结

本章主要介绍在开发过程中需要用的技术简介，以及部分功能的核心代码介绍，其实Spring框架在市面上大部分企业都用到了，Spring也发展的越来好，各种各样的衍生框架慢慢开始发展了；所以时代在变化，我们要跟上时代的变迁，框架的更新迭代才能跟上潮流。

# 第6章 系统测试

## 6.1 测试目标

在简易社区网站系统开发完毕后，我们需要部署、测试，避免在用户在使用的过程的中系统出现较为严重的问题，导致社区网站崩溃等等；所以测试是系统上线的最后一个过程，不仅能够将开发者未发现的错误进行反馈，也能发现部分功能需要优化或者改进等等。

需要明确好需要进行测试的系统功能。对主要功能需要进行压力测试，分功能、分模块和分用户权限层级等等分别进行测试。

简易社区主要是以普通用户、交流模块为主，测试就应该将这两个作为比较重要的模块进行测试，需测试社区常用功能，比如：浏览首页、发布、评论、收藏，点赞，个人信息管理等等；

一旦发现问题，需立即修改，直到未测出问题，测试才算结束。

## 6.2 预期测试结果

进入简易社区网站，用户先看到的是网站首页，如果用户对该社区网站感兴趣的话，就会注册登录，紧接着就是发布，评论等一系列的操作，包括完善个人信息，即交流模块的大部分功能；而我们预期的测试结果为：上述谈到的功能不仅页面不能报错，后台请求也许保证正常运行。

如若是普通用户测试因基本差不多了，要是管理员以及超级管理员，那就不仅仅需要测试交流模块了，还得加上管理模块以及超级管理模块了，测试功能包括对需要置顶或者加精的帖子进行管理、并且能正常发布公告；而超级管理模块需要测试权限给予是否正确，这个功能相对来说比较重要，因为事关权限大小问题，避免因程序异常造成用户权限混乱，所以这个功能比较重要；管理用户虽然在交流模块上要求不怎么严格，但是需要偏重管理这一块；所以预期的结果为权限管理功能需进行压力测试，而且其他能正常使用即可。

## 6.3 测试出异常以及修复规则

在测试的过程中，难免会出现程序报错，导致页面显示异常，碰见这样的问题，应该要按照规则记录问题所在。

所以出现异常的话，将页面报错信息记录，如果是单击了某个按钮导致报错的话，将按钮名称以及报错结果信息记录，有必要的话可以截图。

测试人员将测试结果制作成一个问题反馈文档发送至研发人员，开发人员收到问题反馈文档后，分析问题所在，找到对应解决办法，修改源程序，将修复问题的文档记录保存，修复后保存一个版本，可以推送新的版本给测试人员进行第二轮测试；

在第二轮测试应将前面的问题所在重新测试。以此反复，直至测试没有测试出问题，才能算测试完毕。

## 6.4 运行并记录测试结果

### 6.4.1 交流模块测试结果

（1）登录与注册：

登录和注册时一套系统常用的功能，测试要点为：注册完毕并能正常登录后台打印日志以及前端页面不报错即可；注册以及登录测试结果如下：

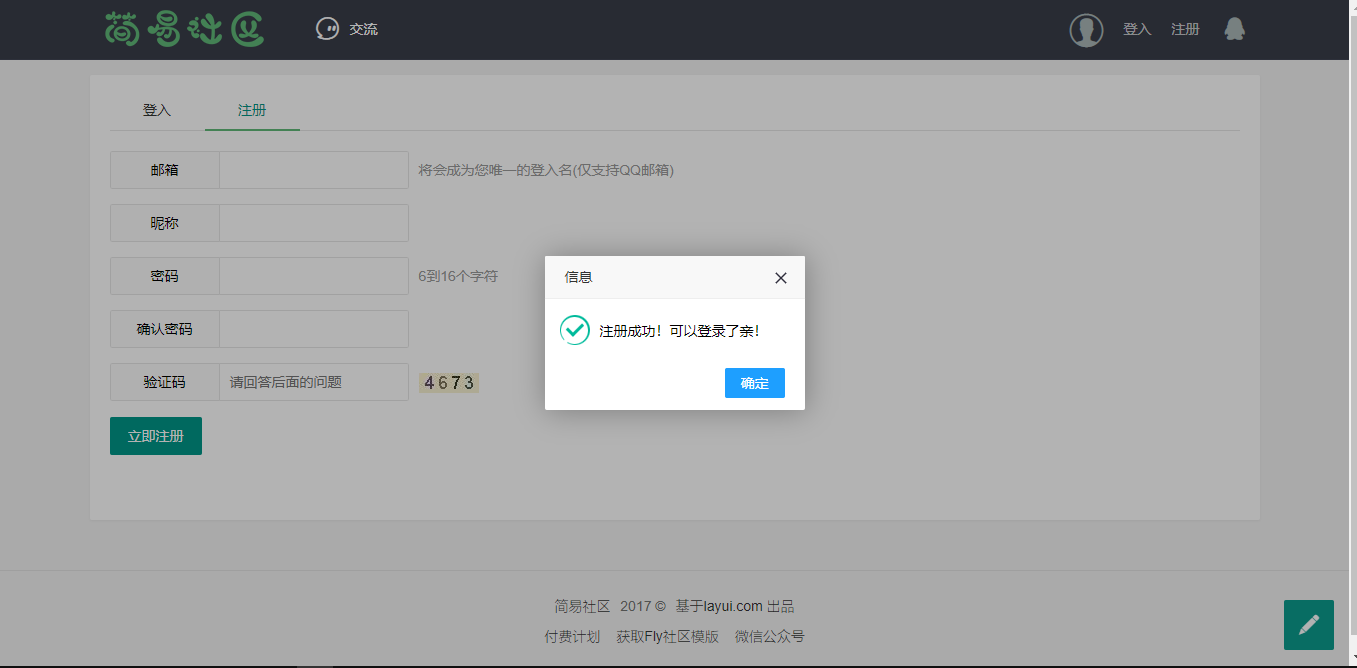


图6.1 图名

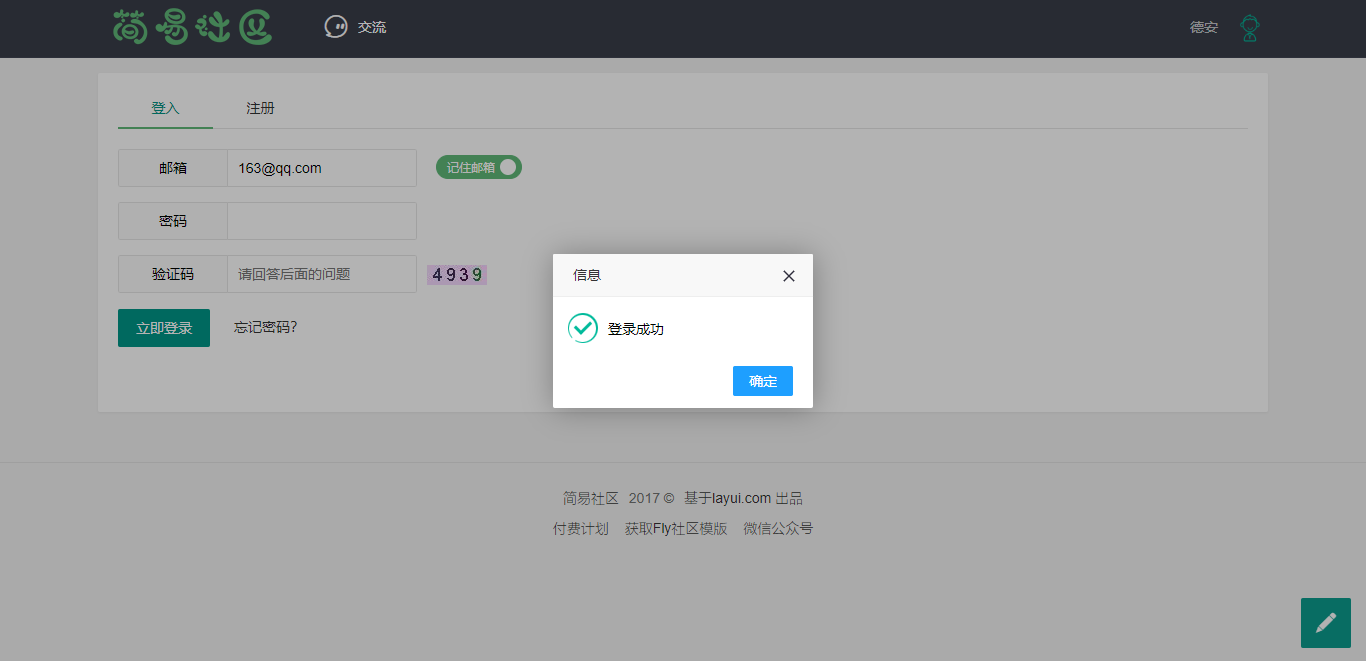


图6.4.1.1登录与注册测试结果图

以上测试结果均为正常写入数据的方式，我们不仅仅应该按照正确的使用方式去进行测试，测试应该将任何输入数据的可能结果都进行测试，以下为错误提示：

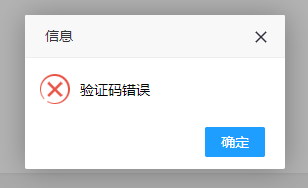


图6.3 验证码提示结果

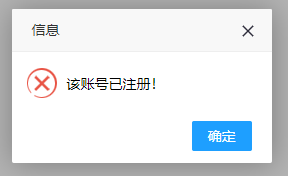


图6.4注册失败结果

经测试后，无论正常执行还是异常运行后台日志以及前端页面均正常；该功能测试完毕。

（2）登录后发布文章以及浏览首页，我们先测试发布帖子，单击首页右下角类似于笔一样的图标即可进入到发布帖子的界面，右上角可以选择帖子所属类型，发布后：

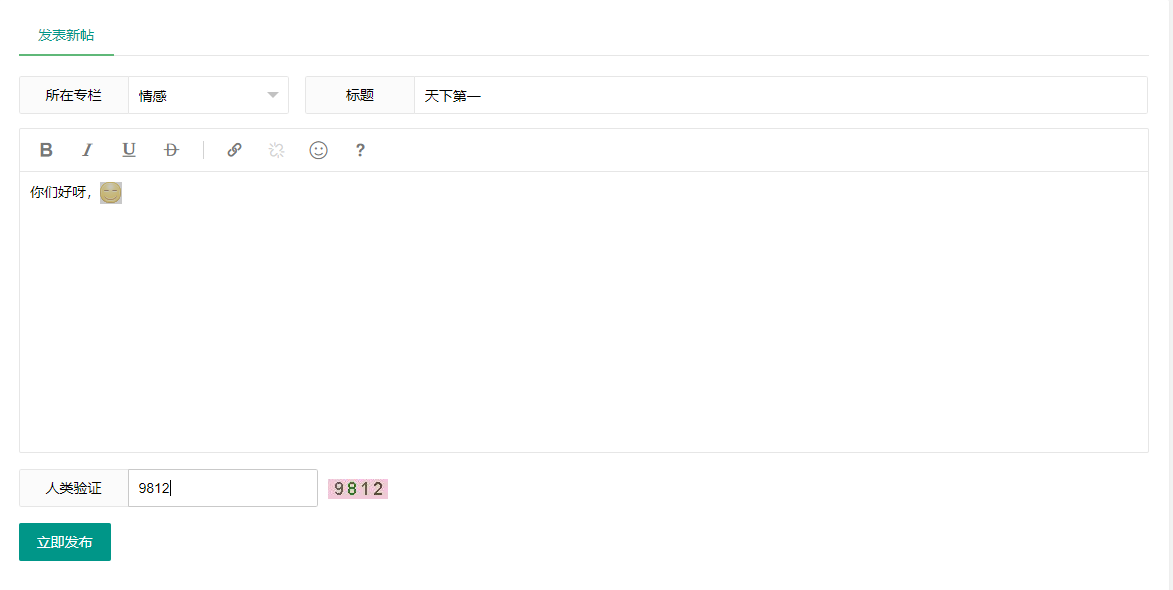


图6.5 图名



图6.6 发布及首页浏览

输入标题，写入内容，输入正确的验证码，即可发布，会自动默认跳转到首页，如果验证码验证失败即会刷新当前页面；在首页功能按钮比较齐全。现在是数据量比较少，展示界面不太好看，下面是数据量较为丰富的时候：



图6.7 数据丰满

（3）现在测试评论、点赞、收藏功能，以下为测试结果：



图 6.8 评论测试



图 6.9 点赞测试



图 6.10 收藏帖子测试

（4）个人信息管理：包括修改个人基本信息比如：昵称、性别、签名等，也能更换比较适合自己的个性头像，加上定期修改密码也是一个良好的习惯；当别人评论了你发布的帖子会出现消息提示，在右上角头像旁边会有红点提示你有新的消息：



图 6.11 个人主页



图 6.12 消息页面

其他的功能测试就不一一展示了。

（5）忘记密码功能测试，这个功能需要用药QQ邮箱，目前使用测试账号测试并未发现太大问题：

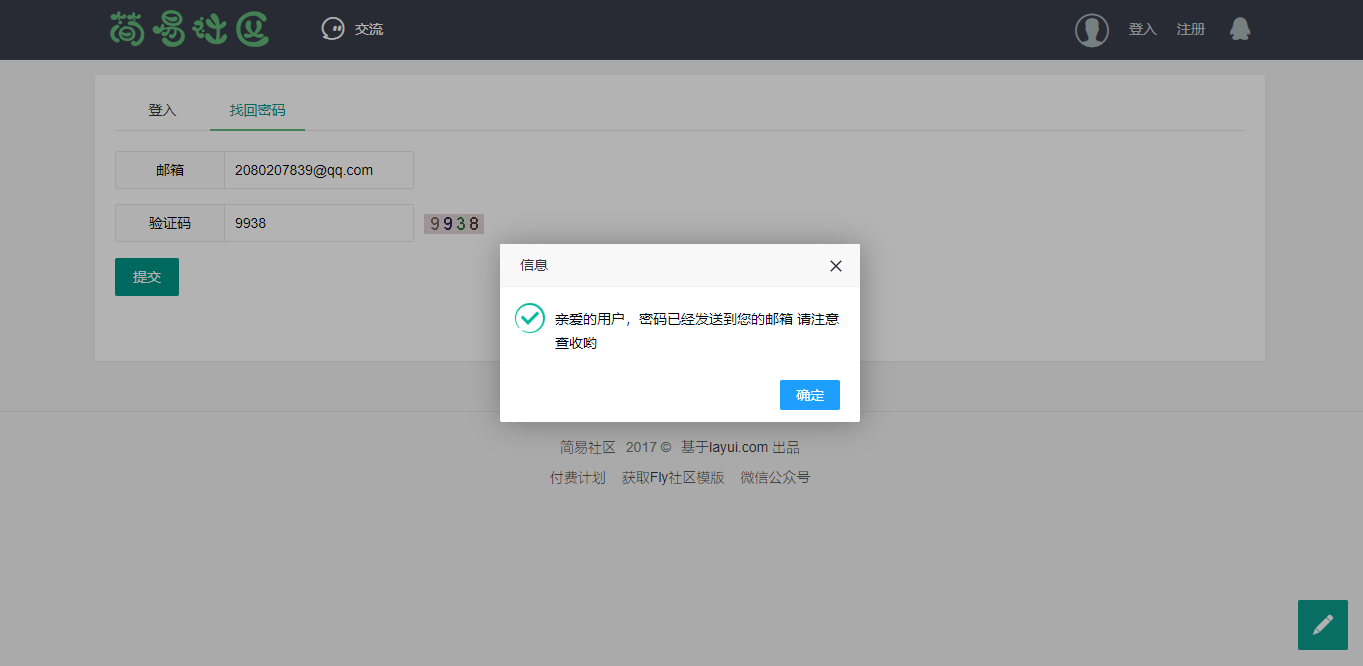


图 6.13 找回密码测试

去邮箱可以找到简易社区发送过来的密码邮件啦！

交流模块的主要功能测试基本完毕。

### 6.4.2 管理模块测试结果

管理员虽然有两个大模块的权限，但交流模块基本不怎么使用，主要管理普通管理模块，普通管理分为三个功能：置顶、加精、发布公告：



图6.14 图名？



图6.15 图名？



图 6.16 置顶、加精、发布公告功能测试截图

测试结果一切正常。

### 6.4.3 超级管理模块测试结果

超级管理员与管理员差不多，主管超级管理模块，该模块分为两个功能：权限人员管理，以及管理帖子类型，能够提升或者降低用户的权限，也有快速查询功能：



图 6.17 修改人员权限测试截图

紧接着就是最后一个功能的测试了----帖子类型管理，我们先进入测试：

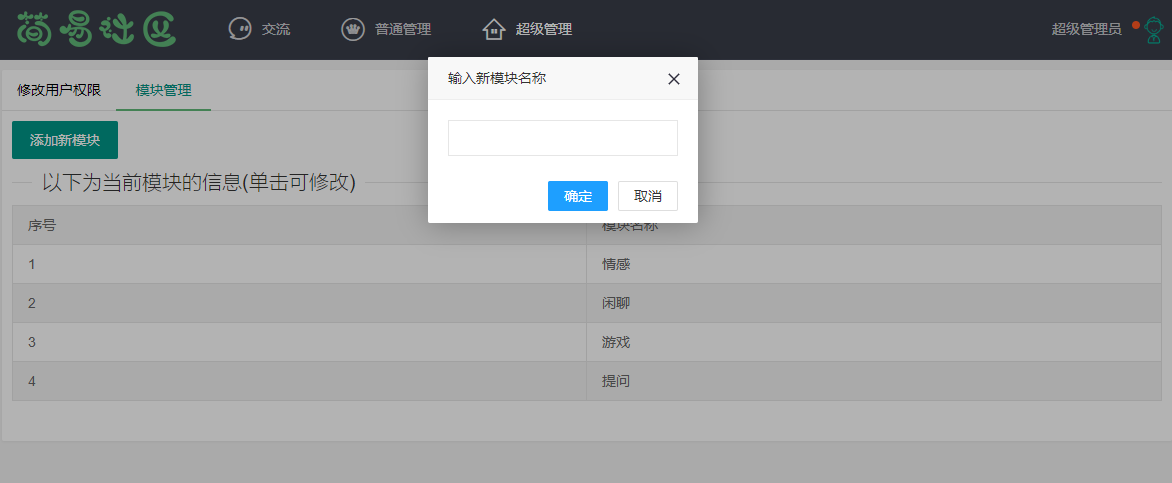


图 6.18 添加类型测试截图

超级管理模块测试完毕。

## 6.5 本章小结

本章主要是介绍测试环节的目的，制定好测试的规范，以供测试人员测试以及开发人员进行修改能够更好的交接工作。

经过几轮的测试后，终于可以将最终版本推送给用户使用了，但这并不代表该系统就没有问题，一旦用户反馈了问题，还是得进行修改，修改完后再进入到测试，不断的反反复复完善系统bug，提供最好的用户体验。

# 总 结

整体论文主要目标是满足于小区的管理、服务、交流的日常需求，虽然现如今社区种类繁多，但每个社区网站都有利有弊，我们需要吸纳优良社区的特色，剔除社区的隐患，总结当前形式，打造出属于自己特色的社区网站。

简易社区在总体的设计开发方案，需要考虑周全，在社区完成初期，主要功能已经完成，但是还差一些辅助功能，这时在进行开发的时候就会发现，之前写的代码都很死，固定了数据、框架的灵活性，一个好的设计方案应该要满足任何条件下都能使用的，比如显示的界面，在分辨率不同的情况显示的界面应该正常才能算的上一个好的界面；就拿数据库来说吧，最开始选择的数据库是Oracle，大家都知道Oracle数据库在性能方面在业界也是数一数二，但是唯一就是得交一定的授权费用，一旦Oracle数据库不能使用，那么整个简易社区就已经凉凉了，这时我们需要拓展另外一个数据库，选择mysql作为备用数据库，以备不时之需；不能说这个项目完成了就算了，对于程序员来说，自己开发的一个项目就像自己的孩子一样，需要不断地去维护它，拓展它的能力。

到此，简易社区网站基本已经完成，虽然目前能正常使用，但是还存在很多问题，在未来时间充裕的情况下，能够拓展业务，不仅仅局限于住宅小区，希望能够提供给大部分年轻人使用。

# 致 谢

在这临近毕业之际、论文完成之时，我要向我的导师以及同学表示感谢，他们在我毕业设计给我提供了很大的帮助，也指点了我毕业项目的很多问题的解决方法，这样才能使我的毕业设计更加完善；遇到问题我一般都是先百度在网上寻找答案，也会去找一些相关的书籍，但是坚持一会儿实在是找不到问题所在，就会去求助我的同学，他们都会很热情的帮助我并且他们会尽力的帮我解决问题。

在做的过程中，我也学会了好多知识，明白了一个项目不仅仅靠的是对技术有多了解，而是靠的实践经验，将技术运用到现实环境中，与在理论课上的理想环境是不一样的，而且不仅仅技术单独靠一个来支撑项目；所以，在大学通过大学课程的学习，我不仅仅学到了理论知识，更重要的是学会了实践，有各种课题设计实践方式来提升我们在现实环境的经验。

感谢我的母校---东华理工大学，让我学会了很多东西，掌握了很多知识，也交到很多朋友，但时间飞逝，眼看马上就要毕业了，在这里要表示真诚的感谢。

# 参考文献

[1]殷彤丽. 基于HTML语言的网页制作方法[J]. 赤峰学院学报(自然科学版), 2011(11):40-42.

[2]田军. 浅谈jQuery技术在Web开发中的应用[J]. 电子世界, 2012(24):8-8.

[3]王艳清, 陈红. 基于SSM框架的智能web系统研发设计[J]. 计算机工程与设计, 2012, 33(12):4751-4757.

[4]佚名. Oracle数据库应用开发基础[M]. 2003.

[5]葛萌,黄素萍,欧阳宏基.基于Spring MVC框架的Java Web应用[J].计算机与现代化,2018(08):97-101.

[6]杨瑞东,王云峰,张海英.Spring新特性之Java Config在Web开发中的应用[J].微型机与应用,2017,36(18):26-29.

[7]周述文, 郭晓军, 孙爱平,等. 电子商务网站的分类及评价初探[J]. 中国管理科学, 2000(s1):748-754.

[8]曾明霏, 刘强. 基于SQL和表设计的Oracle数据库开发审计研究[J]. 软件导刊, 2016, 15(12):136-138.

[9]杨新艳,徐伶伶.基于Java EE的多层软件架构的研究与搭建[J].电子制作,2014(11):87-88.

[10]陈威. 基于JAVA的社区管理和交流平台设计与实现[D].云南大学,2012.

[11]宋佳,范伟,黄鹏.基于Spring的Java平台程序架构研究[J].计算机与网络,2016,42(22):68-71.

[12]马楠. 基于Spring的J2EE安全框架的研究与应用[D].北京交通大学,2009.

[13]王彰忠.Oracle数据库应用系统性能优化技术及其集群技术的实施[J/OL].电子技术与软件工程,2019(05):178-179[2019-03-21].http://kns.cnki.net/kcms/detail/10.1108.TP.20190320.1145.260.html.

[14]翁平.Oracle数据库性能及优化分析[J].信息与电脑(理论版),2019(03):163-164.

[15]赵芬,雷珍臻,杨晓云,苏鹏举,王顺晔.基于百度贴吧大学生网络舆情分析[J].电脑知识与技术,2018,14(28):227-229.

[16]孙思阳,张海涛,任亮,尹慧子.虚拟学术社区用户知识交流行为研究综述[J].情报科学,2019,37(01):171-176.

[17]范凯.Spring Application Platform——SpringSource迈向应用服务器的第一步[J].程序员,2008(06):20.

[18]朱红.“微”时代社区居民的信息交流现状及分析[J].美与时代(上),2017(11):63-65.

[19]姜禾. 以嵌入式社区建设促进民族交往交流交融[N]. 新疆日报(汉),2016-07-07(011).

[20]王茂钢.Java内存模型描述及变量运用分析[J].现代信息科技,2019,3(04):98-99.

[21]Sun, Fengdong, and W. Lan. "Paging query optimization of massive data in oracle 10g database." International Conference on Computer Science & Service System 2011.

[22]Freeman A, Sanderson S. jQuery[J]. Pro Asp Net Mvc Framework, 2011, 8(8):págs. 54-60.

[23]Arthur J, Azadegan S. Spring framework for rapid open source J2EE Web application development: a case study[C]// International Workshop on International Conference on Software Engineering, Artificial Intelligence, Networking & Parallel/distributed Computing. 2005.

[24] Arthur J, Azadegan S. Spring framework for rapid open source J2EE Web application development: a case study[M]. 2005.

[25] Cosmina I. Spring Bean LifeCycle and Configuration[M]// Pivotal Certified Professional Spring Developer Exam. 2017.

[26] Chen Q, Shi H. Design and implementation of domain knowledge service platform based on spring framework[J]. Proceedings of the Association for Information Science & Technology, 2017, 54(1):635-636.

[27] Sharif A, Forouraghi B. evoGraphs — A jQuery plugin to create web accessible graphs[C]// Consumer Communications & Networking Conference. 2018.

# 附 录

附录一：

Spring Framework官方资料：官方网站：[spring.io](https://spring.io)

1. Overview:

* The Spring Framework provides a comprehensive programming and configuration model for modern Java-based enterprise applications - on any kind of deployment platform.
* A key element of Spring is infrastructural support at the application level: Spring focuses on the "plumbing" of enterprise applications so that teams can focus on application-level business logic, without unnecessary ties to specific deployment environments.

1. Features

* [Core technologies](https://docs.spring.io/spring-framework/docs/current/spring-framework-reference/core.html): dependency injection, events, resources, i18n, validation, data binding, type conversion, SpEL, AOP.
* [Testing](https://docs.spring.io/spring-framework/docs/current/spring-framework-reference/testing.html): mock objects, TestContext framework, Spring MVC Test, WebTestClient.
* [Data Access](https://docs.spring.io/spring-framework/docs/current/spring-framework-reference/data-access.html): transactions, DAO support, JDBC, ORM, Marshalling XML.
* [Spring MVC](https://docs.spring.io/spring/docs/current/spring-framework-reference/web.html) and [Spring WebFlux](https://docs.spring.io/spring/docs/current/spring-framework-reference/web-reactive.html) web frameworks.
* [Integration](https://docs.spring.io/spring-framework/docs/current/spring-framework-reference/integration.html): remoting, JMS, JCA, JMX, email, tasks, scheduling, cache.
* [Languages](https://docs.spring.io/spring-framework/docs/current/spring-framework-reference/languages.html): Kotlin, Groovy, dynamic languages.

1. SpringOnePlatform 2017 talks

* [Spring Framework 5.0 on JDK 8 & 9](https://content.pivotal.io/springone-platform-2017/spring-framework-5-0-on-jdk-8-9-juergen-hoeller)
* [Servlet or Reactive Stacks: The Choice is Yours. Oh No…​ The Choice is Mine!](https://content.pivotal.io/springone-platform-2017/servlet-or-reactive-stacks-the-choice-is-yours-oh-no-the-choice-is-mine-rossen-stoyanchev)
* [New in Spring Framework 5.0: Functional Web Framework](https://content.pivotal.io/springone-platform-2017/new-in-spring-framework-5-0-functional-web-framework-arjen-poutsma)
* [Why Spring Loves Kotlin](https://content.pivotal.io/springone-platform-2017/why-spring-loves-kotlin-s%C3%A9bastien-deleuze)