

**目录**

[摘 要 4](#_Toc8924211)

[第1章 前 言 6](#_Toc8924212)

[1.1 项目的背景和意义 6](#_Toc8924213)

[1.2 研究开发现状分析 6](#_Toc8924214)

[1.3 研究目标及范围 6](#_Toc8924215)

[1.4 论文结构简介 6](#_Toc8924216)

[第2章 技术与原理 7](#_Toc8924217)

[2.1 系统开发技术 7](#_Toc8924218)

[2.2 系统开发工具介绍 9](#_Toc8924219)

[第3章 需求建模 10](#_Toc8924220)

[3.1 系统可行性分析 10](#_Toc8924221)

[3.2 系统需求分析 10](#_Toc8924222)

[3.3 系统角色定义 15](#_Toc8924223)

[第4章 架构设计 29](#_Toc8924224)

[4.1系统类图 29](#_Toc8924225)

[4.2 系统技术架构 32](#_Toc8924226)

[4.3系统功能结构图 33](#_Toc8924227)

[4.4数据库ER图 34](#_Toc8924228)

[第5章 模块设计 38](#_Toc8924229)

[5.1 系统设计 38](#_Toc8924230)

[5.2 登陆注册模块 38](#_Toc8924231)

[5.3 个人信息管理模块 41](#_Toc8924232)

[5.4 数据查询模块 44](#_Toc8924233)

[5.5 题目收藏模块 45](#_Toc8924234)

[5.6 题目评论消息模块 47](#_Toc8924235)

[5.7 练习模块 48](#_Toc8924236)

[第6章 部署与应用 50](#_Toc8924237)

[6.1 系统运行环境和系统部署 50](#_Toc8924238)

[6.2 系统的主要应用(用户手册) 50](#_Toc8924239)

[第7章 结 论 57](#_Toc8924240)

**岩生网在线答题系统设计与实现**

摘 要

随着5G时代的到来，人们进入了一个快速发展的网络时代。传统的纸质训练、考试等手段已显得落后，人们在学习与训练时常希望不需要携带繁多的书籍与试题，不受时空限制就能够完成，因此，建立基于网络的答题系统有重要的现实意义。岩生网在线答题系统具有做题、记录、讨论、统计等功能，学习者在使用系统做题训练时，能够自动统计用户做题的对错情况，在做题的时候与其他人一起交流，发表自己的不同见解，并在做完题目之后能够和其他学习者一起讨论做题的心得体会，提高学习效率与练习的效果。系统基于Spring、MyBatis、SpringMvc框架结合MySql数据建库完成，同时使用Redis进行缓存管理，使用Java语言进行开发，结合运用软件工程的开发理论，进行需求分析、设计、实现和测试。论文首先详细分析在线答题系统的需求，明确系统所要实现的功能和系统逻辑；再进行详细设计，分别对各个模块的功能进行描述和设计详细的类图、结构图等；最后，进行系统测试。

关键词：在线答题系统；SSM；Redis；MySql

Online Answering System Of YanSheng Net

Pick to

With the advent of the 5G era, people have entered a rapidly evolving network era. Traditional paper training and examinations have become backward. People often hope that they do not need to carry a large number of books and test questions during their study and training. They can be completed without time and space restrictions. Therefore, establishing a network-based answering system has important practical significance. . Yansheng Online's online answering system has functions such as doing questions, recording, discussion, statistics, etc. When the learner uses the system to do the problem training, the learner can automatically count the right or wrong situation of the user's questions, and communicate with others when making the questions. I have different opinions and can discuss the experience of doing the exercises with other learners after completing the questions, and improve the efficiency of learning and the effect of practice. The system is based on Spring, MyBatis, SpringMvc framework combined with MySql data database, and uses Redis for cache management, using Java language for development, combined with the use of software engineering development theory, demand analysis, design, implementation and testing. Firstly, the paper analyzes the requirements of the online answering system in detail, and clarifies the functions and system logic to be realized by the system. Then, the detailed design is carried out to describe the functions of each module and design detailed class diagrams and structure diagrams. Finally, the system test is carried out.

**Keywords: online answering; SSM; Redis; MySql**

第1章 前 言

* 1. 项目的背景和意义

随着5G时代的到来，我们进入了一个快速发展的网络时代。传统的纸质考试等手段已经开始落后了，人们更希望能随时随地学知识，而不需要带着繁多的书籍，试卷；同时也不希望考试完成，在那之后，你仍然需要等待繁琐的过程。可以说网络为信息处理和信息传输提供了更快，更安全的平台，作为练习，巩固知识点并给予网络答题系统设计符合用户的需求。因而建立基于网络的答题系统有很强的现实意义。通过在系统中不断的练习，巩固自己的知识点，为学习知识的学生等人群提供了系统的练习、学习知识的基于网络的途径，同时也能迎接5G时代的到来，让人们在生活中更快、更便捷 的服务。

* 1. 研究开发现状分析

21世纪最重要的即是Internet，可以连接到Internet的设备已扩展到各种设备，例如移动电话和平板电脑。所有这些都认为这种传统设备和资源已经到了变革的时刻。目前，考试的主要方式是通过纸质试卷进行的，这种方法有许多缺点：每个问题都需要手动评分，不能自动评分，工作量非常大；无法从问题库中随机选择测试题来形成测试试卷；考试完成后，考试无法实现立刻看到成绩，时效性差；考生在拿到纸质试卷后才能看到自己的错误；相对于纸质试卷，对于很多题目汇总困难。

基于这种现状，岩生网在线答题系统能够非常好的解决当下的这些问题，节省大量的纸质试卷，同时能够减少很多人力，试卷可以使用系统自动出题、考试结果能够使用系统自动阅卷、题后解析能够直接展示，并且错题、经典的知识点都能够在系统中随时记载，节省了很多时间，同时所有的题目参与者都能够自由的参与讨论，表达对题目的见解，加深对知识的吸收，减轻老师、同学的负担。综上所述，岩生在线答题系统的开发的价值非常高。

* 1. 研究目标及范围

伴随5G时代的到来，将练习获取知识的过程搬上网络，使得网上答题系统能够让用户随时随地进行练习学习，学习新知识，同时能够多方面的组合学习，增强自己的能力；在练习之后能够随即知道自己哪些不足，能够自己查看解析，并与其他联系者一起沟通，能够全方位的了解所学知识点；并且在练习过程中会记录所有的统计记录，方便自己随时随地的了解自己的不足，以及自己的优势，通过练习错题来补充自己的不足；当然练习时临时需要处理其他事情的时候，平台也将提供“断电续作”的功能，方便随时随地的进行学习。

* 1. 论文结构简介

论文主旨在于对系统的可行性，系统的需求，以及如何实现进行叙述。论文第二章主要介绍了为了实现系统使用到的技术，第三章主要对当前系统的需求进行了详细的分析，同时结合实际技术对需求如何实现做了简要分析；四到五章介绍了实现的细节，对数据库的设计，系统架构、结构的设计，以及各个模块之间的依赖关系；第六章介绍了系统的部署环境，以及用户手册；最后部分即使对论文的结论以及对老师的致谢。

第2章 技术与原理

B/S结构是浏览器/服务器，直接通过浏览器即可为用户提供服务，将系统的功能实现集中在服务器上，简化了系统的开发、维护以及使用。系统使用了spring，mybatis以及mvc的设计模式，将模块间的耦合度降到最低—spring中使用IOC容器减少对依赖注入的耦合：mvc模式将controller，view，model三层分离，增加model层也就是数据访问dao以及service层的代码逻辑复用，降低耦合；使用了mysql以及索引的技术，结合redis缓存增加了访问的速度；使用了html5，jquery，bootstrap，ajax实现异步访问，局部刷新，为用户提供服务而不刷新整个页面，同时ajax定时发送请求，是用户能够实时的接收到评论的消息

2.1 系统开发技术

2.1.1 SSM框架技术

系统开发的主体框架技术是SSM，SSM框架是Spring，SpringMVC，MyBatis三个框架的简称；使用Spring对系统中的事务、Bean进行管理，降低耦合，通过SpringMVC对请求进行拦截，访问到特定的controller，通过Mybatsi实现对数据库的封装访问，通过三个框架整合，是系统的耦合度降到最低，同时能够方便快捷的对数据库进行访问，返回数据到页面。

三个框架的简介：SpringMVC：实现三层分离，model、controller、view 也就是数据访问层、页面、以及连接三层实现分离，好处就是能够实现model层代码的复用，减少代码量，降低耦合。MyBatis框架简介：MyBatis封装了JDBC访问数据库的功能，并在此基础上进行优化，减少重复的代码，增加程序的可读性，同时能够自定义SQL，大大增加了与数据库连接的灵活性，并且能够支持存储过程和高级映射。 MyBatis可以使用XML配置文件或注解的方式来配置和映射数据库信息。Mybatis 缓存机制：一级缓存. 针对每一个sqlsession实现缓存；执行两次相同的查询时，即第二次使用缓存中的数据； 缓存使用哈希表实现，默认开启以及缓存，二级缓存：针对每一个namespace，实现二级缓存； 默认不开启，需手动配置，自带得缓存机制缺陷：针对每次 增删改 都会重新刷新整个缓存，而不是某一个单一得信息；

2.1.2 Redis以及NoSQL

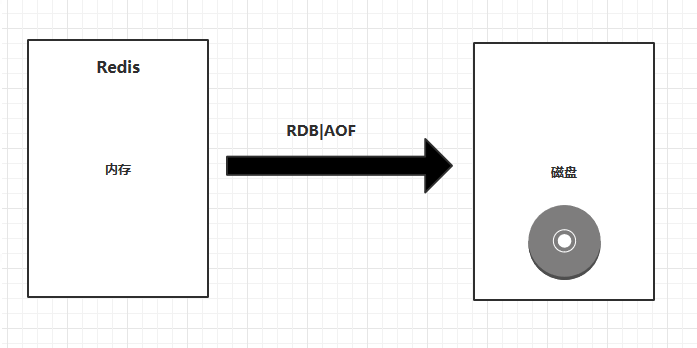
Redis介绍：Redis即使缓存机制，将对数据库查询频繁的数据加入缓存中减少数据库读取的次数，直接从缓存中读取即可大大降低数据读取方面的时间消耗，使系统访问更加迅速；系统中使用到了Redis存储用户的做题记录，实现了“断点续作”，同时使用到Redis存储用户的做题统计信息，做完练习即对统计信息进行动态的更新减少读取数据库的次数。Redis速度快的原因：Redis把所有数据都是存放在内存中的，同时是Redis是用C语言实现的,而C语言即是计算机的基本语言，执行速度相对更快，Redis使用了单线程的模式，即不会产生的竞争问题。Redis 的持久化：Redis提供了两种持久化方法：RDB和AOF，也就是说，您可以使用两种策略来保存硬盘上的内存数据，从而确保数据的持久性，如 图2-1 Redis的持久化。  


图2-1 Redis的持久化

虽然Mybatis自带了缓存机制，但是使用Mybatis的自带缓存机制，有很多缺陷：每次增删改查都需要对所有的进行重新刷新，所以导致服务器压力大；使用Redis缓存机制能够很好的解决当前的问题。

Redis是Nosql非关系型数据库，与传统的mysql等数据库关系型数据库相比有所不同。Nosql介绍：不仅sql是非关系型数据库; （这种类型的数据不需要固定模式，可以扩展而无需不必要的操作）Nosql的优点，易于扩展，高数据量，高性能，多样化和灵活的数据模型（增加字段很简单）；Nosql的三大特性：Consistence 强一致性 ：所有节点在同一时间具有，相同得数据；数据事物提交之后，所有查询都是一个结果， Availability可用性可以快速获取数据，在确定的时间内返回操作结果，不管成功与否， Partition tolerance 分区容错性 系统的任意信息丢失或者错误，不会影响整个系统的正常运行。CAP 只能三选二：最多满足两个，分区容错性必须存在。

2.1.3 ajax

为什么要使用ajax？传统的提交请求需要重新刷新整个页面，加重浏览器负担，如果服务器长时间未响应，浏览器也会无响应，用户体验差，提交表单，发送的数据大，可能造成网络问题，使用ajax可以很好的解决这方面的问题：问题：当数据部分刷新且页面没有响应时，页面仍然显示原始网页可以在后台与服务器执行少量数据交互。异步刷新网页是异步刷新的，无需等待服务器响应。发送请求后，还能对页面进行操作，浏览；系统中绝大部分的请求数据返回都是使用到了ajax，通过ajax的使用，能很好的增强用户体验，使系统更加适配低配置机器访问。

2.1.3 MySQL

系统中使用的是MySQL数据库，MySQL是现有企业中选择比较多的数据库存储，优点是体积小，速度快，成本低，免费同时能够支持较高的数据量查询，提高速度。

2.2 系统开发工具介绍

2.2.1 Intellij IDEA

是开发软件中比较智能的一款IDE，能够通过在编码中的各种智能提示来辅助开发者更加快速的、更加方便的编写出健壮性强的代码，同时也能够支持多种语言开发。

2.2.2 Git

Git最大的开源代码托管工具，Git能够上传任何文件，是一个免费的代码集中管理仓库，同时在没有网络的情况下也能够存储提交代码方便快捷。

2.2.3 Maven

Maven基于POM项目对象模型。自动化构建工具发展史：Mark->Ant->maven->Gradle以“java源文件。jsp，配置文件，html，静态资源“等去“编译“一个可以运行的工具；分三部分，编译，部署，搭建，编译：java源文件，编译成class字节码执行。自动化构建的工具。

第3章 需求建模

3.1 系统可行性分析

系统的开发往往有时间和资源的限制。因此，在项目启动之前，需要进行可行性分析，可以降低项目开发的风险，以免造成人力、物力、财力的浪费。下面从技术可行性、社会可行性、操作可行性三个方面来进行分析。

3.1.1 技术可行性

项目采用Spring+SpringMVC+MyBatis框架完成后台系统，加上MySQL数据库结合索引加快读取速度，同时结合Redis存入用户读取数据量频繁的数据，减轻数据库的压力，并且增加访问的速度；前端页面ajax使用多，异步刷新数据，减轻浏览器以及服务器的压力

3.1.2 社会可行性

1. 法律可行性：系统未触犯任何法律。
2. 人员可行性：系统的使用简单便捷，方便实时练习，查询自己的统计；为学习的学生提供了很好的平台

3.1.3 市场需求可行性

随着网络越来越发达，流量越来越便宜智能设备的普及人民获取信息的方式手段比以前更多，使用网络来获取知识也同时大受欢迎，能够随时随地的访问网站获取知识。并同时总结自己的不足，提供错题本

3.1.4 操作需求可行性

简单易用，智能设备普及以及网络的发达，系统能够达到简单快速上手的要求，方便广大用户进行学习

3.2 系统需求分析

《岩生网在线答题系统》的目的是能够方便用户随时随地的学习，并且省去自己统计，以及记录错题，并能够随时随地的和其他用户讨论题目的机会，让用户无时无刻的都能在学习中找到自己的不足，并且和别人一起共同的成长，让互联网成为用户获取知识的更有效的有段。无需纸质试卷，无需人工阅卷，无需自己统计记录错题，了解自己的不足，无需等到老师的讲解，练习完即可和一起答题的用户讨论题目，随时随地了解自己的不足。基于网络的《岩生网在线答题系统》相比传统的纸质试卷能够更加方便，更加的帮助用户提升自己，以及学习新知识

3.2.1 系统流程图



图 3-1 系统总体流程图

如图 3-1 系统总体流程图 ，用户进入网站，根据自己是否已有账号，来决定自己下一步：注册，或者登录；没有账号则新注册一个，有账号忘记密码则进入忘记密码流程，重置自己的密码，登录之后用户即可正常的使用平台。

进入平台用户即查看自己的资料，以及修改，查看自己的收藏题目，查看自己的错题，查看自己的做题统计，查看与自己相关的消息，做练习，以及断点续作。断电续作会自动记录上次做题的记录，下次继续学习的时候即可有选择的直接进入练习阶段；练习阶段：可以选择三种方式，错题练习，专项练习，随机练习，系统会自动出题，用户做完题目之后，能够评论题目，以及与各个用户讨论题目，并且查看自己的错题，解析。同时觉得题目的知识点是自己需要的时候，可以收藏起来，留着以后在看。



图 3-2 登录注册忘记密码流程图

用户第一次使用系统时，需要进行邮箱以及手机号的验证保证信息唯一且方便以后修改密码，同时能够只能够在登录之后才能使用系统，系统会根据用户来记录做题统计，以及错题统计等信息，所以用户使用系统时必须登录系统。登录流程图如图 3-2 登录忘记密码流程图



图 3-3 用户修改个人资料

如图 3-3 用户修改个人资料；用户在登录之后，系统会提供一个默认头像，需要用户自己修改用户头像，同时能够修改一些自己的信息，住址、电话号码、邮箱，同时会经过验证才能修改。



图3-4 用户练习

如图3-4用户练习：用户登录系统之后能够选择练习的方式，专项练习：顺序练习某一类题目；随机练习，能够选择多个分类，将根据分类的数量随机的输出题目的数量；错题练习：能够随机的练习错题，但是不会更新了正确率；选择好练习的方式，就能选择好题目的类型，java等会自动生成题目，用户在做题时，会记录到缓存中，在用户未做完题目时，能够再次进入系统时能够查询到缓存并提示用户续作，做完练习之后，系统会自动阅卷，返回答案以及解析给用户提醒大致解题逻辑，并且用户做完题目之后能够收藏题目，在之后能够点击收藏的题目继续查看题目知识点；做完题之后还能够对题目进行评论，回复，能够与其他人一起讨论题目的知识点，巩固自己的知识，加深记忆的印象。



图3-5 其他功能简略流程图

系统还提供了很多小功能，如图 3-5 其他功能简略流程图 能够查看自己的错题，能够查看自己收藏的题目，同时也能够查看自己的做题统计，了解自己的不足。

3.3 系统角色定义

3.3.1用户

所有注册，并进入系统的用户即为系统的使用用户。进入系统必须要登录，并且用户注册时，需要进行邮箱验证，并且在之后修改密码等需要使用邮箱进行验证。用户进入系统时，默认显示首页，能够选择的查看自己收藏的题目、查看和自己相关的消息评论、查看自己的做题统计记录、选择做题的类型、个数、题目的类型、并且修改自己的个人信息，浏览自己的信息。

3.3.2 用例描述

如图3-6用户登录注册用列图，用户在第一次使用系统时需要进行注册，注册成功后即可登录进入系统；忘记密码时能够通过系统邮箱验证进行密码修改，详细用例描述如下表3-1 表3-2 表3-3



图3-6用户登录注册

用户注册用例描述如表3-7

表3-7注册用例描述

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 001 | 用例名称 | 注册 |
| 编制人 | 雷科 | 编制日期 | 2019-05-06 |
| 用例描述 | 用户输入信息，进行注册 | | |
| 参与者 | 未注册用户 | | |
| 前置条件 | 首次进入系统，没有账号的用户 | | |
| 后置条件 | 验证信息符合规则，并且验证成功 | | |
| 优先级别 | 高 | | |
| 基本事件流   1. 用户进入注册界面 2. 输入正确的基本信息，以及邮件验证码 3. 效验基本信息正常、邮件验证码正确注册成功   返回登录界面 | | | |
| 可选事件流   1. 输入错误的信息，提示用户修改 | | | |
| 特殊需求 | 无 | | |
| 扩展点 | 无 | | |

用户登录用例描述如表3-8

表3-8登录用例描述

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 002 | 用例名称 | 登录 |
| 编制人 | 雷科 | 编制日期 | 2019-05-06 |
| 用例描述 | 用户需要使用系统的时候，即需要进行登录 | | |
| 参与者 | 用户 | | |
| 前置条件 | 首次进入系统，没有账号的用户 | | |
| 后置条件 | 验证信息符合规则，并且验证成功 | | |
| 优先级别 | 高 | | |
| 基本事件流   1. 输入正确的账号信息点击登录 2. 效验账号密码信息，同时效验是否在黑名单中 3. 登录成功   跳转到系统首页 | | | |
| 可选事件流   1. 输入错误的信息，提示用户修改 2. 用户在黑名单中，联系管理员 | | | |
| 特殊需求 | 无 | | |
| 扩展点 | 无 | | |

用户修改密码用例描述如表3-9

表3-9修改密码用例描述

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 003 | 用例名称 | 修改密码 |
| 编制人 | 雷科 | 编制日期 | 2019-05-06 |
| 用例描述 | 用户忘记密码的时候即可以通过此用例重置密码 | | |
| 参与者 | 用户 | | |
| 前置条件 | 忘记密码 | | |
| 后置条件 | 验证信息符合规则，并且验证成功 | | |
| 优先级别 | 高 | | |
| 基本事件流   1. 输入自己的账号 2. 输入自己的账号绑定的邮箱 3. 邮箱账号绑定无误之后，发送邮箱验证码 4. 输入正确的验证码 5. 输入修改后的密码并确认   返回登录界面 | | | |
| 可选事件流   1. 重置密码时，记起密码能够进入登录界面 | | | |
| 特殊需求 | 无 | | |
| 扩展点 | 无 | | |





图3-10 用户修改信息、练习

如图3-7用户修改信息、练习用例图，用户能够在登录之后，修改自己的个人信息；并且能够选择练习的方式、题目的数量、题目的类型进行练习，练习未完成退出练习下次即可进行“断点续做”，也可以删除上次做题记录，重新开始练习。

修改个人信息用例描述如表3-11

表3-11修改个人信息用例描述

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 004 | 用例名称 | 修改个人信息 |
| 编制人 | 雷科 | 编制日期 | 2019-05-06 |
| 用例描述 | 用户在登录之后能够通过修改个人信息，修改自己的绑定邮箱、手机号、性别、住址、头像等信息 | | |
| 参与者 | 用户 | | |
| 前置条件 | 无 | | |
| 后置条件 | 验证信息符合规则，并且验证成功 | | |
| 优先级别 | 中 | | |
| 基本事件流   1. 选择需要修改的信息 2. 输入正确信息 3. 修改完成，返回修改个人信息界面 | | | |
| 可选事件流   1. 无 | | | |
| 特殊需求 | 无 | | |
| 扩展点 | 无 | | |

练习用例描述如表3-12

表3-12练习用例描述

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 005 | 用例名称 | 练习 |
| 编制人 | 雷科 | 编制日期 | 2019-05-06 |
| 用例描述 | 用户在登录之后，进入首页选择练习的方式、练习的题目数量，系统出题，用户练习 | | |
| 参与者 | 用户 | | |
| 前置条件 | 登陆之后，用户需选择练习的方式以及题目的数量 | | |
| 后置条件 | 无 | | |
| 优先级别 | 高 | | |
| 基本事件流   1. 用户选择做题的方式、题目的数量 2. 系统出题，用户练习 3. 交卷 | | | |
| 可选事件流   1. 做题时，关闭页面下次登录能够在首页看到“断点续做”按钮，点击即可继续上次做题 2. 能够提前交卷 | | | |
| 特殊需求 | 无 | | |
| 扩展点 | 无 | | |

如图3-13 查看练习详情，用户做完题目之后，能够查看自己的做题详情，同时能对喜欢的题目进行收藏，下次便于查看知识点，系统会自动记录错题，供下次练习使用，同时用户能够对题目进行评论，并且能够回复别人的评论，针对题目的知识点展开讨论、学习。



图3-13 查看做题详情

查看做题详情用例描述如表3-14

表3-14查看做题详情

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 006 | 用例名称 | 查看做题详情 |
| 编制人 | 雷科 | 编制日期 | 2019-05-06 |
| 用例描述 | 用户做完练习之后点击交卷，显示题目的内容，选项，答案，对错，解析评论，回复，收藏等信息 | | |
| 参与者 | 用户 | | |
| 前置条件 | 用户做完练习之后 | | |
| 后置条件 | 无 | | |
| 优先级别 | 高 | | |
| 基本事件流   1. 用户做完题目，并交卷 2. 显示题目的对错信息，对则显示绿色，错则显示红色 3. 显示收藏信息，收藏即显示取消收藏，没有收藏即显示收藏按钮 4. 显示评论，回复信息 5. 有评论即显示评论信息，并显示评论下的回复信息数量 6. 有数量即显示数量并且显示点击查看详情，没有数量即显示暂无评论 7. 自己的评论显示删除按钮，不显示回复按钮 | | | |
| 可选事件流   1. 收藏题目，能够在首页查看 2. 评论题目 3. 回复别人的评论，参与讨论 | | | |
| 特殊需求 | 无 | | |
| 扩展点 | 无 | | |

收藏题目用例描述如表3-15

表3-15收藏题目

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 007 | 用例名称 | 收藏题目 |
| 编制人 | 雷科 | 编制日期 | 2019-05-06 |
| 用例描述 | 用户在查看题目详情界面能够点击题目进行收藏 | | |
| 参与者 | 用户 | | |
| 前置条件 | 用户做完练习之后 | | |
| 后置条件 | 无 | | |
| 优先级别 | 中 | | |
| 基本事件流   1. 题目默认为不收藏 2. 没有收藏即形式为未收藏按钮 3. 收藏了即显示取消收藏按钮 | | | |
| 可选事件流   1. 取消收藏 | | | |
| 特殊需求 | 无 | | |
| 扩展点 | 无 | | |

收藏题目用例描述如表3-16

表3-16收藏题目

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 008 | 用例名称 | 收藏题目 |
| 编制人 | 雷科 | 编制日期 | 2019-05-06 |
| 用例描述 | 用户查看自己的收藏的题目时，能够删除自己的收藏信息 | | |
| 参与者 | 用户 | | |
| 前置条件 | 用户做完练习之后或者在查看收藏信息界面 | | |
| 后置条件 | 无 | | |
| 优先级别 | 中 | | |
| 基本事件流   1. 在展示用户收藏的信息界面上 2. 点击取消收藏即可删除收藏信息 | | | |
| 可选事件流  无 | | | |
| 特殊需求 | 无 | | |
| 扩展点 | 无 | | |

评论题目用例描述如表3-17

表3-17评论题目

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 009 | 用例名称 | 评论题目 |
| 编制人 | 雷科 | 编制日期 | 2019-05-06 |
| 用例描述 | 用户查看题目详情的时候，能够对题目新增评论，输入信息后点击评论按钮即可对题目评论 | | |
| 参与者 | 用户 | | |
| 前置条件 | 用户在查看题目详情页面 | | |
| 后置条件 | 无 | | |
| 优先级别 | 高 | | |
| 基本事件流   1. 评论框默认显示，用户在输入框中输入自己的评论信息时，点击评论时，即发送请求，添加评论信息 2. 评论完毕之后显示自己的评论信息，并且自己仅仅能够删除，不能回复 | | | |
| 可选事件流  无 | | | |
| 特殊需求 | 无 | | |
| 扩展点 | 无 | | |

删除评论用例描述如表3-18

表3-18删除评论

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 0010 | 用例名称 | 删除评论 |
| 编制人 | 雷科 | 编制日期 | 2019-05-06 |
| 用例描述 | 自己的评论能够删除，并且连带删除评论之下的所有回复 | | |
| 参与者 | 用户 | | |
| 前置条件 | 用户在查看题目详情页面，并且仅仅能够删除自己的评论 | | |
| 后置条件 | 无 | | |
| 优先级别 | 高 | | |
| 基本事件流   1. 评论完之后可以选择删除评论 | | | |
| 可选事件流  无 | | | |
| 特殊需求 | 无 | | |
| 扩展点 | 无 | | |

回复用户评论用例描述如表3-19

表3-19删除评论

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 0011 | 用例名称 | 回复用户评论 |
| 编制人 | 雷科 | 编制日期 | 2019-05-06 |
| 用例描述 | 用户能够对其他用户的评论进行回复 | | |
| 参与者 | 用户 | | |
| 前置条件 | 用户在查看题目详情页面，仅仅能够回复别人的评论，自己的评论不能回复 | | |
| 后置条件 | 无 | | |
| 优先级别 | 高 | | |
| 基本事件流   1. 用户仅仅能够回复别人的评论 2. 自己的评论不能回复 | | | |
| 可选事件流  无 | | | |
| 特殊需求 | 无 | | |
| 扩展点 | 无 | | |

如图3-7 辅助功能用例描述，系统中还有其他简单的辅助功能，能够优化用户的体验，查看自己收藏的题目，查看消息即是别人对自己的评论或者回复做出的回复，也能够查看自己的做题统计，了解自己的不足。



图3-20辅助功能用例描述

浏览个人做题统计用例描述如表3-21

表3-21浏览个人做题统计

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 0012 | 用例名称 | 浏览个人做题统计 |
| 编制人 | 雷科 | 编制日期 | 2019-05-06 |
| 用例描述 | 用户查看自己在系统中做题的所有统计信息 | | |
| 参与者 | 用户 | | |
| 前置条件 | 用户登录之后 | | |
| 后置条件 | 无 | | |
| 优先级别 | 高 | | |
| 基本事件流   1. 登录之后在首页能够查看自己的统计信息 | | | |
| 可选事件流  无 | | | |
| 特殊需求 | 无 | | |
| 扩展点 | 无 | | |

查看收藏的题目用例描述如表3-22

表3-22查看收藏的题目

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 0013 | 用例名称 | 查看收藏的题目 |
| 编制人 | 雷科 | 编制日期 | 2019-05-06 |
| 用例描述 | 用户查看自己的收藏的题目 | | |
| 参与者 | 用户 | | |
| 前置条件 | 用户登录之后 | | |
| 后置条件 | 无 | | |
| 优先级别 | 高 | | |
| 基本事件流   1. 登录之后在首页能够查看自己收藏的题目信息 | | | |
| 可选事件流   1. 用户能够点击加载更多，显示更多题目收藏信息 | | | |
| 特殊需求 | 无 | | |
| 扩展点 | 无 | | |

查看消息用例描述如表3-23

表3-23查看消息

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 0014 | 用例名称 | 查看消息 |
| 编制人 | 雷科 | 编制日期 | 2019-05-06 |
| 用例描述 | 用户收到消息时，能够在所有页面展示出消息的数量，并且点击消息，能够查看消息的详情信息 | | |
| 参与者 | 用户 | | |
| 前置条件 | 用户登录之后，其他人回复了自己的评论 | | |
| 后置条件 | 无 | | |
| 优先级别 | 高 | | |
| 基本事件流   1. 能够及时的显示未读消息，提醒用户 2. 点击消息详情能够进入题目的详情，显示评论 3. 显示删除按钮 | | | |
| 可选事件流   1. 点击消息即可进入题目的详情界面，显示评论、回复的内容 2. 若无收藏数据，即显示暂无数据 3. 分页展示数据量 4. 点击加载更多即加载其他分页 | | | |
| 特殊需求 | 无 | | |
| 扩展点 | 无 | | |

删除消息用例描述如表3-24

表3-24删除消息

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 0015 | 用例名称 | 删除消息 |
| 编制人 | 雷科 | 编制日期 | 2019-05-06 |
| 用例描述 | 用户浏览自己的消息时，能够点击删除，删除掉自己不想看到的消息 | | |
| 参与者 | 用户 | | |
| 前置条件 | 用户登录之后，其他人回复了自己的评论 | | |
| 后置条件 | 无 | | |
| 优先级别 | 高 | | |
| 基本事件流   1. 显示删除按钮 2. 点击删除即可删除 | | | |
| 可选事件流  无 | | | |
| 特殊需求 | 无 | | |
| 扩展点 | 无 | | |

删除消息用例描述如表3-25

表3-25删除消息

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 0016 | 用例名称 | 删除消息 |
| 编制人 | 雷科 | 编制日期 | 2019-05-06 |
| 用例描述 | 用户浏览自己的消息时，能够点击删除，删除掉自己不想看到的消息 | | |
| 参与者 | 用户 | | |
| 前置条件 | 用户登录之后，其他人回复了自己的评论 | | |
| 后置条件 | 无 | | |
| 优先级别 | 高 | | |
| 基本事件流   1. 显示删除按钮 2. 点击删除即可删除 | | | |
| 可选事件流  无 | | | |
| 特殊需求 | 无 | | |
| 扩展点 | 无 | | |

第4章 架构设计

4.1系统类图

4.1.1 用户模块

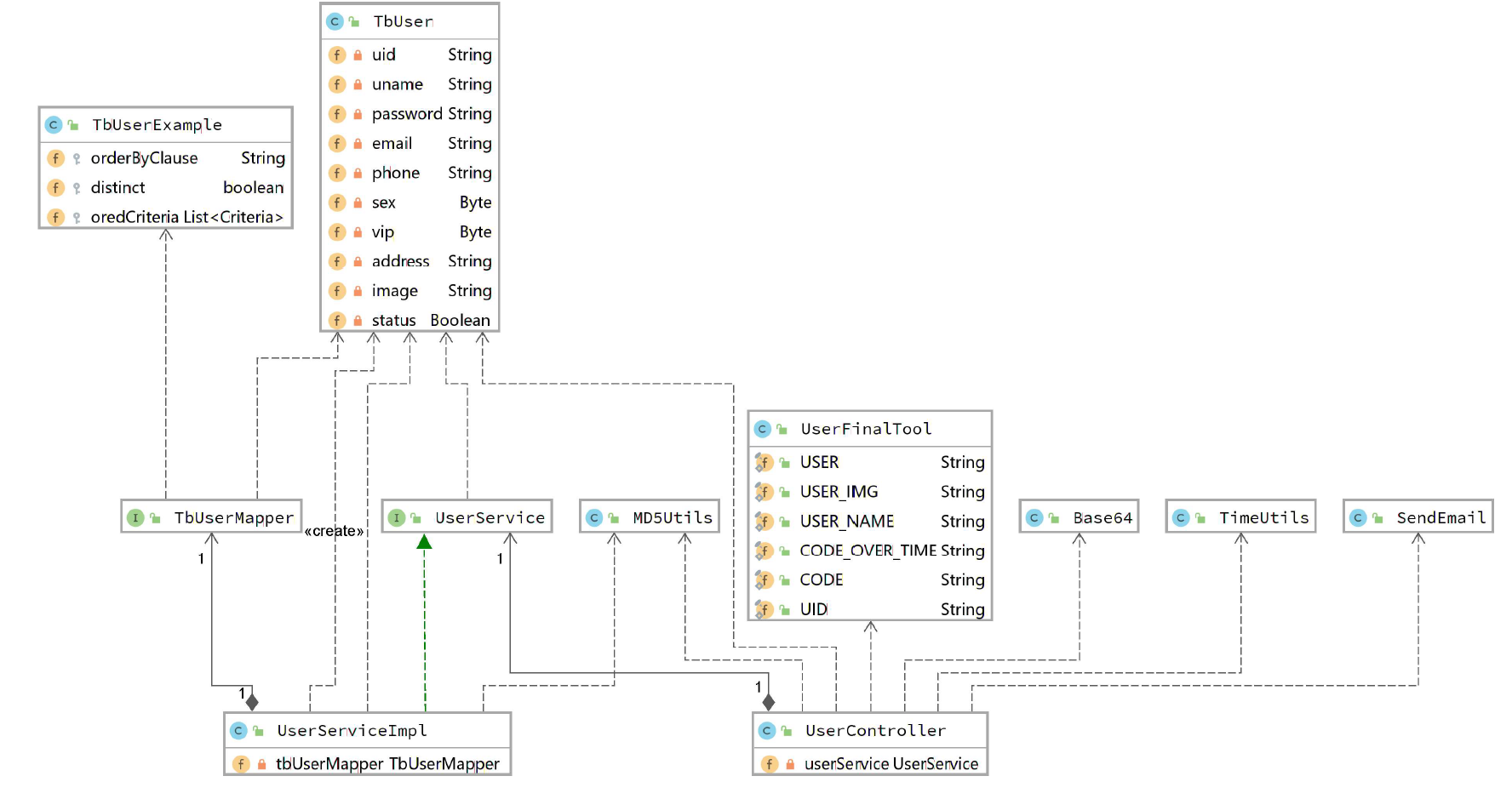


图4-1用户模块类图

如图4-1用户模块类图，用户模块中的TbUser类用于与数据库的字段值相互对应，提供DAO查询时信息的封装；通过UserController结合UserServiceImpl提供了用户登录，注册，个人信息修改查询功能；同时结合各个工具类MD5Utils，Base64，TimeUtils，SendEmail工具类分别提供了密码加密，图片上传，邮件发送验证码即验证码时间验证。

4.1.2 评论回复模块

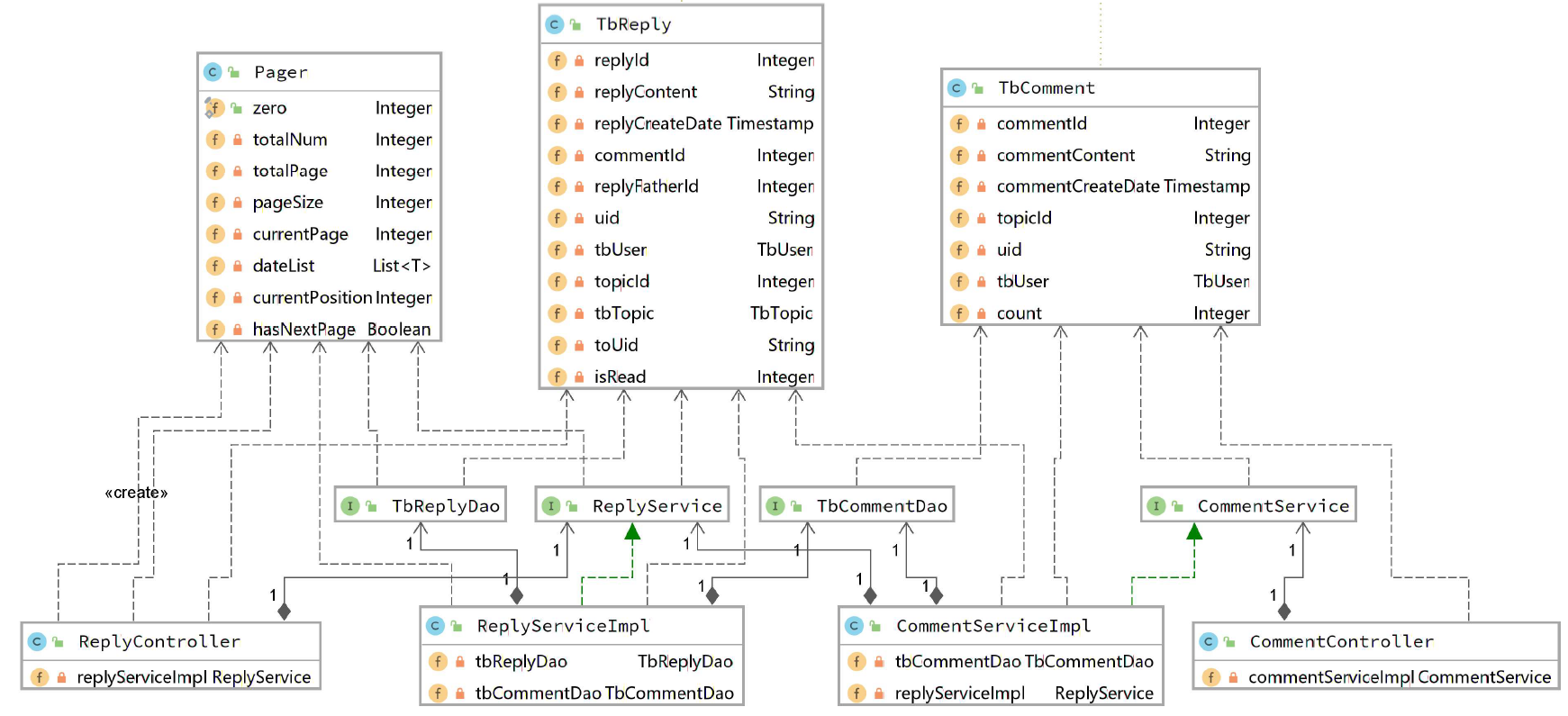


图4-2评论回复模块类图

如图4-2评论回复模块类图评论回复模块中，评论是指用户在题目下的第一级评论，回复是指其他用户对评论的回复；评论需要关联查询用户以及题目的信息，所以评论以及回复的TbCommentTbReply中会存在用户以及题目的信息；通过ComentController和CommentServiceImpl提供评论的功能，对题目进行评论，通过ReplyController和ReplyServiceImpl提供回复功能，分页工具类Pager提供给回复消息提醒时提供分页。

4.1.3 题目收藏模块

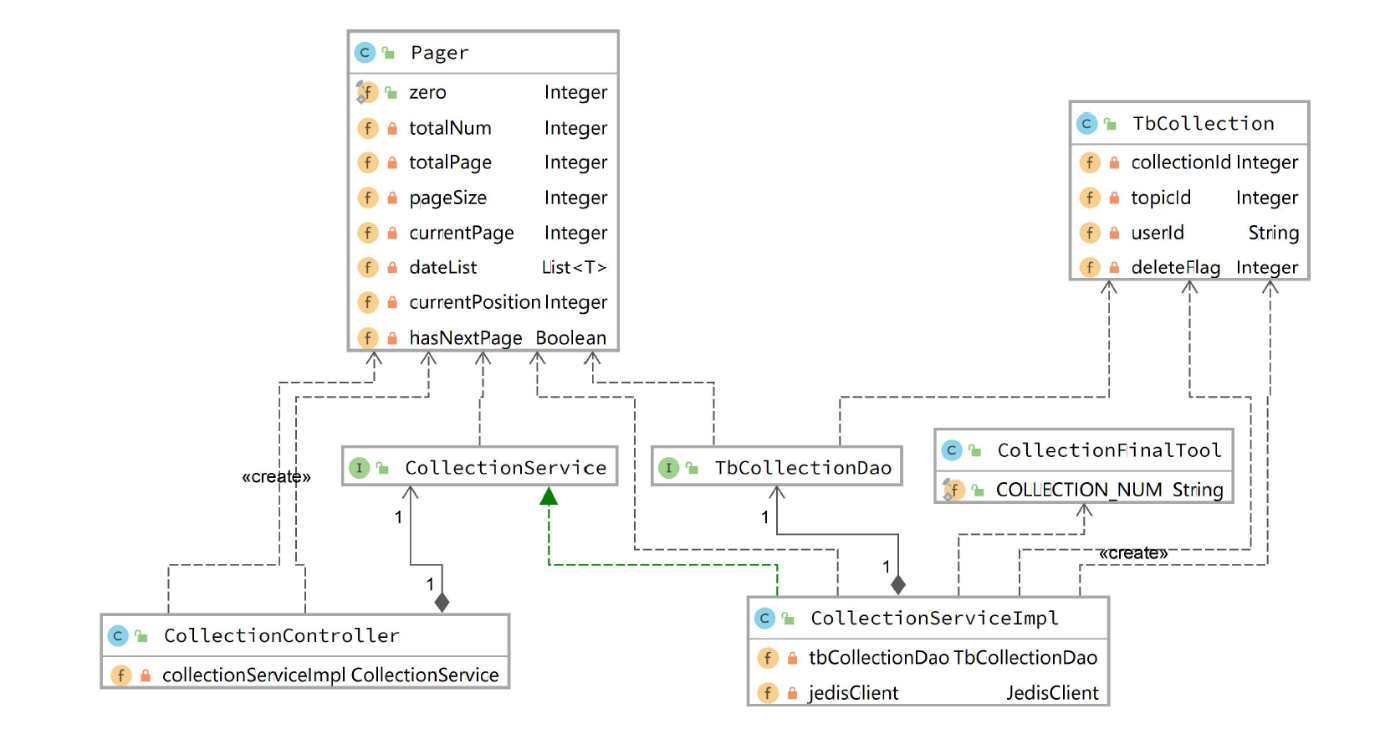


图4-3题目收藏模块类图

题目收藏模块总CollectionController和CollectionServiceImpl提供收藏题目，取消收藏的功能，同时会对收藏的题目查询时进行分页显示，TbCollection与数据库字段相对应，查询数据时进行封装。

4.1.4 题目选项模块

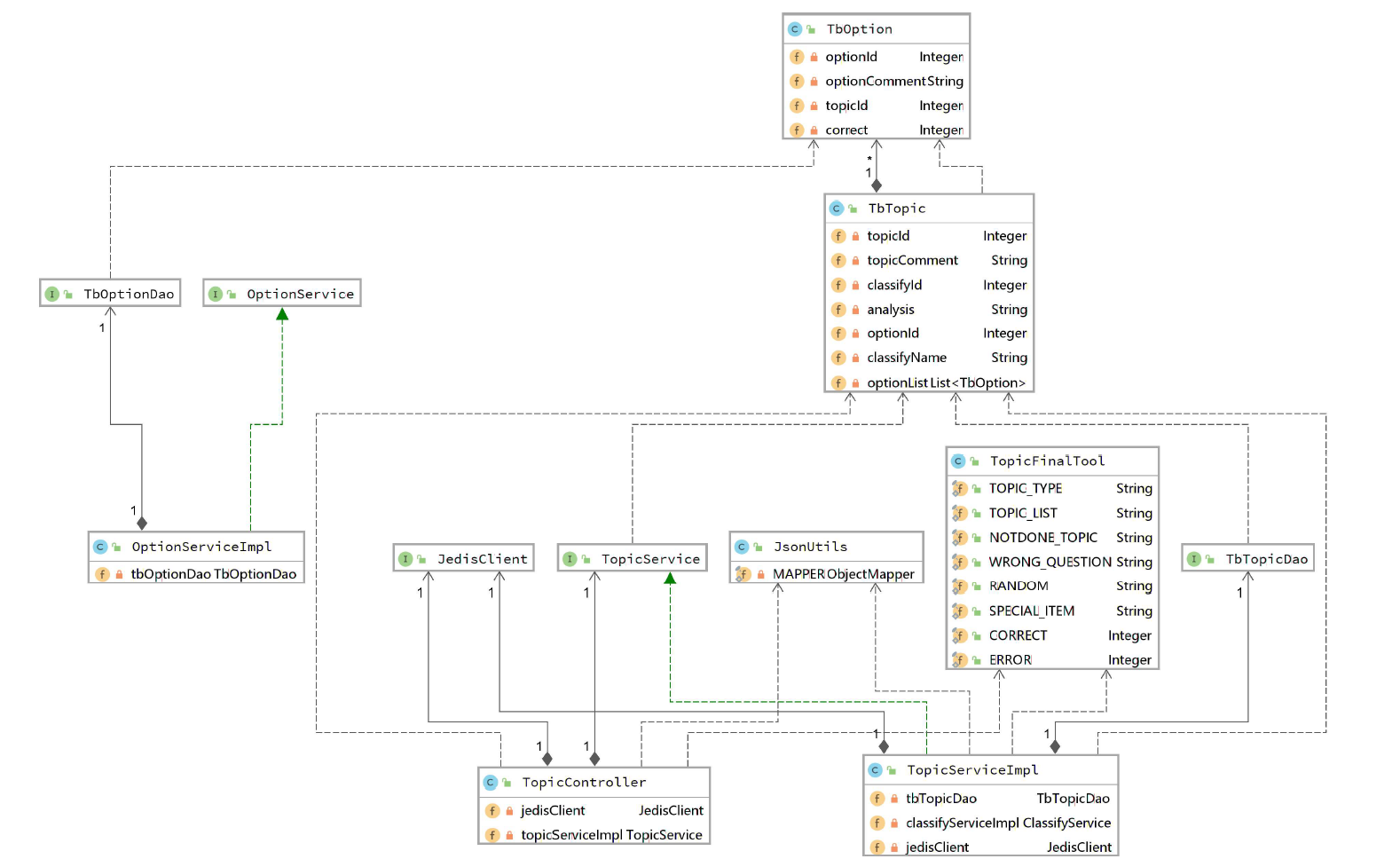


图4-4题目选项类图

题目选项模块一个题目对应多个选项，TbTopic中增加了TbOption选项的信息，再查询是可以将题目以及选项信息封装到一起，返回给前台页面，在题目生成式会使用到Redis缓存，同时集合Json解析工具进行对象转字符串，字符串转对象的方式。

4.1.5 做题模块

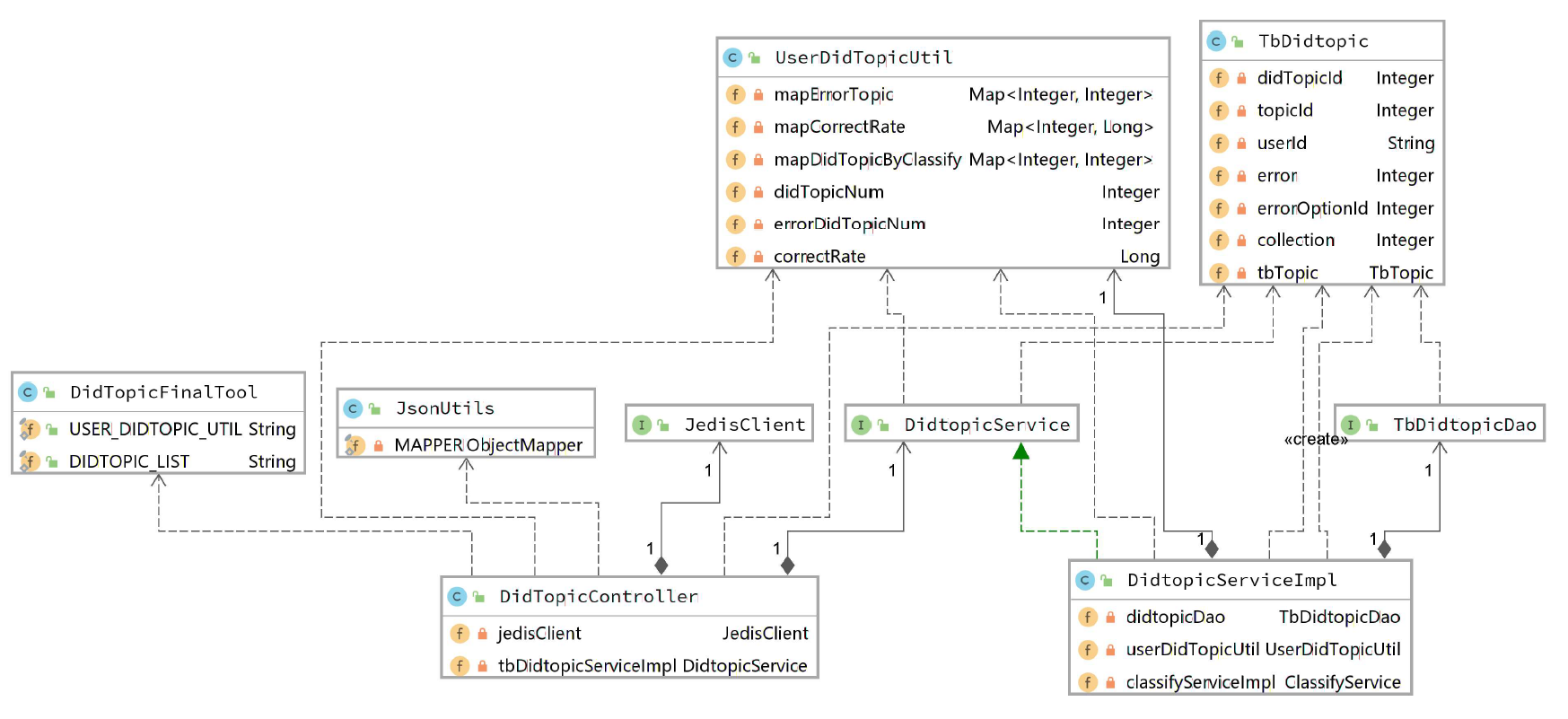


图4-5做题模块类图

做题模块会对用户所有做过的题目进行统计存入到Redis中，同时使用自定义的统计类记录每个用户每个类别的做题数量，及正确率。

4.2 系统技术架构

本系统采用的是B/S架构模式，B/S架构即为我们常说的[浏览器](http://baike.baidu.com/view/7718.htm" \t "_blank)/[服务器](http://baike.baidu.com/view/899.htm)模式，是Browser/Server的简称，是目前最为流行的一种架构模式，它是依托浏览器，是客户端最常用的软件。使用B/S架构模式，统一了访问的客户端，让客户使用更加便捷，系统功能的核心部分都部署在服务器中，简化了开发者的维护与开发。客户只需要在本地计算机上安装浏览器，当系统需要升级时，客户只需要刷新浏览器页面即可获得最新的升级后的效果，服务器安装SQL Service、MySql、oracle等数据库即可以对数据进行存储。方便快捷的操作，使得B/S架构也越来越流行。



图4-6 B/S架构模型

4.3系统功能结构图

如图4-7系统功能结构图，系统主要分为六大模块：登录注册模块，数据查询模块，题目收藏模块，题目评论消息模块，练习模块，个人信息模块。登录注册模块主要提供用户登录以及注册，登录的功能；数据查询模块，按照分类记录用户所有的做题统计信息的模块；题目收藏模块，展示用户收藏的模块信息；练习模块，提供用户进行专项练习，随机练习，顺序练习的模块，评论消息模块，用户的评论以及回复都会及时通知，并显示给用户。

****

图4-7 系统功能结构图

4.4数据库ER图

如图 4-8 数据库ER图，根据现有的功能设计出来的数据库表关联，以用户表为主要的关联表，关联了做题表记录做过哪些题目；关联评论表，记录发表过的评论；关联回复表，记录发表过的回复信息以及接受的回复，同时回复表自我关联，能够将回复一直关联下去；关联了收藏表，记录用户收藏的题目；题目表关联分类表，记录题目所属的分类，同时关联选项表，记录题目的选项信息。

计算机生成了可选文字:
tbcomment
cc•mmentld
3℃03012
「eateOate
c肱032》
tb_city
、℃「《50
tb_reply
3℃03012
「epO'C「eateOate
「epO'Fath•e「黾
3「《30
C「《32》
tb
didtopic
tbuser
d巧TO卩℃
c「《32》
SS0
C「《32》
3「《30
e「「。「Op卜'Id
3011》
3℃03012
3℃03「崆到
tbcollection
3「《30
deleteFlag
tb_province
“℃「《50
option
0000nd
“℃「《50
tbtopic
to卩℃
“℃「《50
、℃「《50
tb_classiW
0“引f
03if'/N艹

图 4-8 数据库中ER图

4.2.1 数据表结构

表4-9 tb\_city城市表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** | **Comment** |
| cid | int(11) | NO | PRI | (NULL) | auto\_increment | 城市id |
| name | varchar(50) | YES |  | (NULL) |  | 城市名称 |
| pid | int(11) | YES | MUL | (NULL) |  | 省份id |

表4-10 tb\_classify题目分类表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** | **Comment** |
| classifyId | int(4) | NO | PRI | (NULL) | auto\_increment | 分类id |
| classifyName | varchar(20) | NO |  | (NULL) |  | 分类名称 |

表4-11 tb\_collection 题目收藏表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** | **Comment** |
| collectionId | int(4) | NO | PRI | (NULL) | auto\_increment | 收藏id |
| topicId | int(4) | NO | MUL | (NULL) |  | 题目id |
| userId | char(32) | NO | MUL | (NULL) |  | 用户id |
| deleteFlag | tinyint(1) | NO |  | 0 |  | 逻辑删除 |

表4-12 tb\_comment题目评论表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** | **Comment** |
| commentId | int(4) | NO | PRI | (NULL) | auto\_increment | 评论id |
| commentContent | varchar(100) | NO |  | (NULL) |  | 评论内容 |
| commentCreateDate | timestamp | NO |  | CURRENT\_TIMESTAMP | on update CURRENT\_TIMESTAMP | 评论创建时间 |
| topicId | int(4) | NO | MUL | (NULL) |  | 题目id |
| uid | char(32) | NO | MUL | (NULL) |  | 用户id |

表4-13 tb\_didTopic用户做题表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** | **Comment** |
| didTopicId | int(4) | NO | PRI | (NULL) | auto\_increment | 做题id |
| topicId | int(4) | NO | MUL | (NULL) |  | 题目id |
| userId | char(32) | NO | MUL | (NULL) |  | 用户id |
| error | tinyint(1) | YES |  | 1 |  | 是否做错 |
| errorOptionId | int(4) | YES |  | -1 |  | 错误选项 |

表4-14 tb\_option选项表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** | **Comment** |
| optionId | int(4) | NO | PRI | (NULL) | auto\_increment | 选项id |
| optionComment | varchar(50) | NO |  | (NULL) |  | 选项内容 |
| topicId | int(4) | NO | MUL | (NULL) |  | 题目id |
| correct | tinyint(1) | NO |  | 0 |  | 是否是正确答案 |

表4-15 tb\_province省份表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** | **Comment** |
| pid | int(11) | NO | PRI | (NULL) | auto\_increment | 省份id |
| name | varchar(50) | YES |  | (NULL) |  | 省名 |

表4-16 tb\_reply评论回复表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** | **Comment** |
| replyId | int(4) | NO | PRI | (NULL) | auto\_increment | 回复id |
| replyContent | varchar(100) | NO |  | (NULL) |  | 回复内容 |
| replyCreateDate | timestamp | NO |  | CURRENT\_TIMESTAMP | on update CURRENT\_TIMESTAMP | 回复创建内容 |
| commentId | int(4) | NO | MUL | (NULL) |  | 评论id |
| replyFatherId | int(4) | YES |  | -1 |  | 回复父id |
| toUid | char(32) | NO | MUL | (NULL) |  | 接受用户id |
| topicId | int(4) | YES | MUL | (NULL) |  | 题目id |
| uid | char(32) | YES | MUL | (NULL) |  | 发送用户id |
| isRead | tinyint(1) | YES |  | 0 |  | 是否已读 0 未读， 1已读 |

表4-17 tb\_topic题目表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** | **Comment** |
| topicId | int(4) | NO | PRI | (NULL) | auto\_increment | 题目id |
| topicComment | varchar(50) | NO |  | (NULL) |  | 题目内容 |
| classifyId | int(4) | NO | MUL | (NULL) |  | 分类id |
| analysis | varchar(50) | YES |  | test |  | 解析 |

表4-18 tb\_user用户表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Extra** | **Comment** |
| uid | char(32) | NO | PRI | (NULL) |  | 用户id |
| uname | char(255) | NO | UNI | (NULL) |  | 用户名 |
| password | char(32) | NO |  | (NULL) |  | 密码 |
| email | char(32) | NO | UNI | (NULL) |  | 邮箱 |
| phone | char(11) | YES |  | (NULL) |  | 电话号码 |
| sex | tinyint(4) | NO |  | 1 |  | 性别 |
| VIP | tinyint(4) | NO |  | 0 |  | 是否是VIP |
| address | varchar(100) | YES |  | -1 |  | 地址 |
| image | varchar(255) | NO |  | /static/img/userHeadImg/user.png |  | 头像地址 |
| status | tinyint(1) | NO |  | 1 |  | 是否在黑名单中 |

第5章 模块设计

5.1 系统设计

系统主旨在于提供用户随时随地的学习机会和其他学霸随时随地的讨论的机会、方便用户查看自己的不足，同时能够方便用户查看错题，而不需要自己手动记录，同时能够省去很多人工的阅卷成本。

系统的功能环环相扣，练习模块，做题详情展示模块，收藏模块，评论回复模块，做题统计模块缺一不可，能够为用户提供一系列的功能，方便用户的学习以及减少时间的成本

5.2 登陆注册模块

5.2.1 模块描述

用户使用系统的入口，所有使用平台的用户必须要注册，才能方便用户的做题统计信息，收藏，评论，错题等功能的使用，界面大致如图5-1登录界面、图5-2注册界面第一步、图5-3注册界面第二步、图5-4注册完成。

5.2.2 界面设计

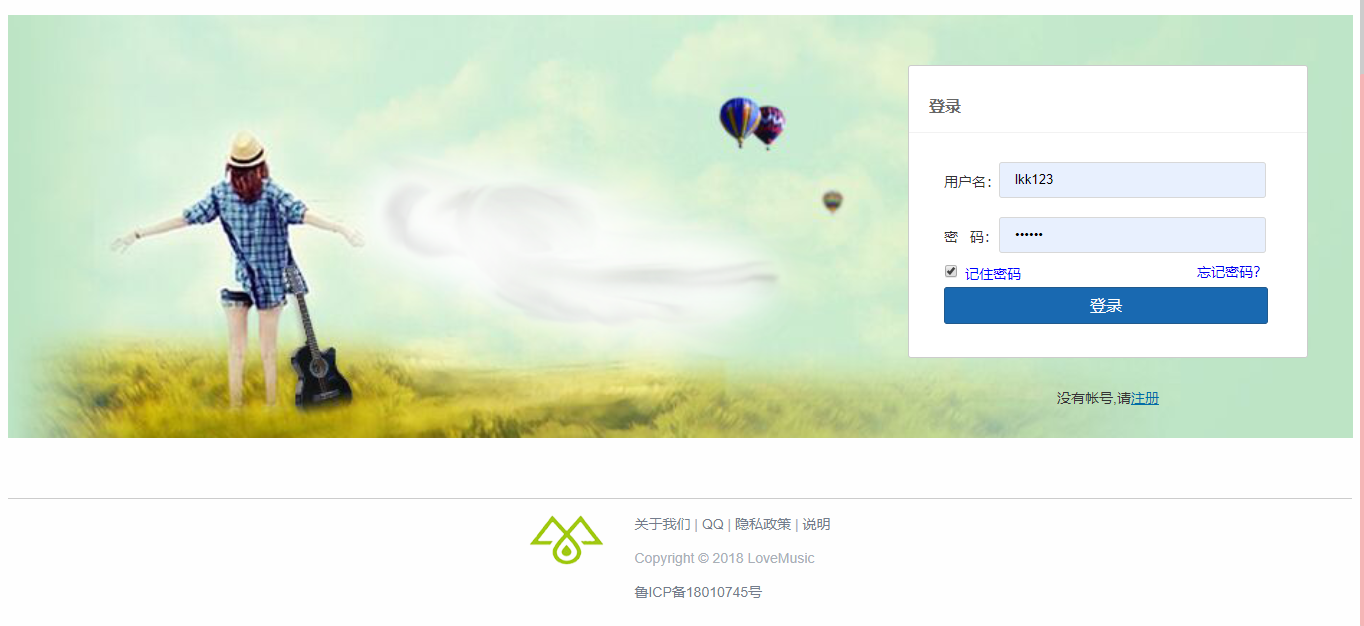


图5-1 登录界面



图5-2 注册界面第一步



图5-3 注册界面第二步



图5-4 注册完成界面

5.2.3 程序逻辑

登录界面，用户输入账号密码，后台会进行md5加密并进行验证，账号不存在，密码错误时提示用户“账号或者密码错误”，同时会进行黑名单判断，在黑名单的用户将无法登录，并且提示“账号异常，请联系管理员”。用户首次使用系统，没有账号的时候，通过注册进行账号的申请。

注册分为三步：第一步用户输入账号名，账号密码，账号绑定的邮箱，以及图片验证码，并且阅读条款；输入的账号名必须使用英文字母开头，并且3到20位大小，并且和已经注册过的用户进行校验，保证用户名不能重复，重复了提示用户“用户名已经使用”；输入的邮箱也将保证和现有注册用户不重复，并且邮箱的格式正确；密码必须是数字字符的组合，不然提示用户密码安装等级太低，正确的输入了图片验证码的验证码之后，勾选条款，点击下一步进入，第二步邮箱验证的界面；第二步邮箱验证，使用用户第一步输入的邮箱，发送验证码到邮箱上，用户输入验证码后进行效验，验证码错误即提示用户验证码错误，此步骤是为了检验邮箱是本人邮箱；邮箱验证完毕之后进入第三部，输入手机号码，验证手机号规则，同时完成注册，界面在5秒之后自动跳转到登陆界面，或者用户手动点击进入登录界面。

5.3.4 后台处理设计

涉及到的类：contoller层：UserController类, PageController页面跳转类；service：UserService接口以及UserSeviceImpl接口的实现类，dao：TbUserMapper接口，以及配置文件TbUserDao.mapper，POJO类：TbUser、TbUserExample类，发送邮件的工具类：SendEmail，密码加密工具MD5Utils，JedisClient Redis工具类，以及静态常量类UserFinalTool。

5.3 个人信息管理模块

5.3.1 模块描述

如图5-1 注册第一步、图5-2 注册第二步、图5-3 注册第三步、图5-4 修改个人信息，用户在登录时，忘记密码能够点击忘记密码来修改自己的密码，修改密码时需要验证邮箱，保证是账号的主人操作修改密码；登录之后，用户能够查看自己的个人资料，绑定的手机号，性别，绑定的邮箱，基本住址信息，头像，用户名等信息，并且能对部分信息进行修改。

5.3.2 界面设计



图5-1 注册第一步



图5-2 注册第二步



图5-3 注册第三步



图5-4 用户修改个人信息

5.3.3 程序逻辑

修改密码分三步：验证邮箱，保证是本人操作；输入修改密码并确认；修改成功自动跳转登录界面；用户在登录时，点击忘记密码，输入用户名和用户名下绑定的邮箱，输入完用户名时，先效验后台数据中是否存在该用户，不存在即提示用户名不存在；输入用户名之后在输入邮箱地址，验证邮箱地址是否正确，不正确提示用户邮箱格式错误，同时根据用户名和邮箱地址一同确认邮箱是否是该用户绑定的，不是即提示用户邮箱错误，邮箱地址和用户名都符合要求时，才可以点击发送验证码到邮箱，确认了邮箱以及操作是本人之后才能进入下一步；下一步输入新密码，密码必须是字符和数字的组合，并且需要输入两次确认密码；修改成功后自动进入登录页面或者用户手动点击进入。

修改个人信息界面，能够修改绑定的邮箱，电话号码，头像等信息；修改头像时需要用户上传图片，并且手动的截切；修改邮箱以及电话号码时需要验证是否是本人操作，修改性别，居住地时没有约束。

邮箱验证：验证码有效时间三分钟，三分钟内不会重复发送，将验证码以及沿着那么的有效时间都记录到session中。

5.3.4 后台处理设计

涉及到的类：contoller层：UserController类, PageController页面跳转类；service：UserService接口以及UserSeviceImpl接口的实现类，dao：TbUserMapper接口，以及配置文件TbUserDao.mapper，POJO类：TbUser、TbUserExample类，发送邮件的工具类：SendEmail，密码加密工具MD5Utils，JedisClient Redis工具类，Base64加密，TimeUtils自定义时间类，JsonUtils 对象转Json字符串工具类，以及静态常量类UserFinalTool。

5.4 数据查询模块

5.4.1 模块描述

如图5-5 首页统计数据展示，用户在系统中的所有做题统计都会记录到系统中，记录系统中所有做题统计信息，以及记录每个分类的统计信息以及正确率。

5.4.2 界面设计



图5-5 首页统计数据展示

5.4.3 程序逻辑

用户每次选择完做题类型以及做题的方式，并且完成练习提交答卷之后，系统会自动判断对错并统计到相应的分类，以及总统计的记录中。

5.4.4 后台处理设计

controller层：DidTopicController，service层：DidtopicService，DidtopicServiceImpl，UserDidTopicUtil；dao层DidtopicDao；TopicFinalTool，TbDidtopic，jedisClient，TbTopic

5.5 题目收藏模块

5.5.1 模块描述

如图5-6 练习详情展示、如图5-7 收藏题目展示，用户在完成练习界面，题目详情显示界面能够收藏题目，方便之后查看；同时首页提供查询收藏页面的入口，收藏页面展示用户所有的收藏题目信息。

5.5.2 界面设计

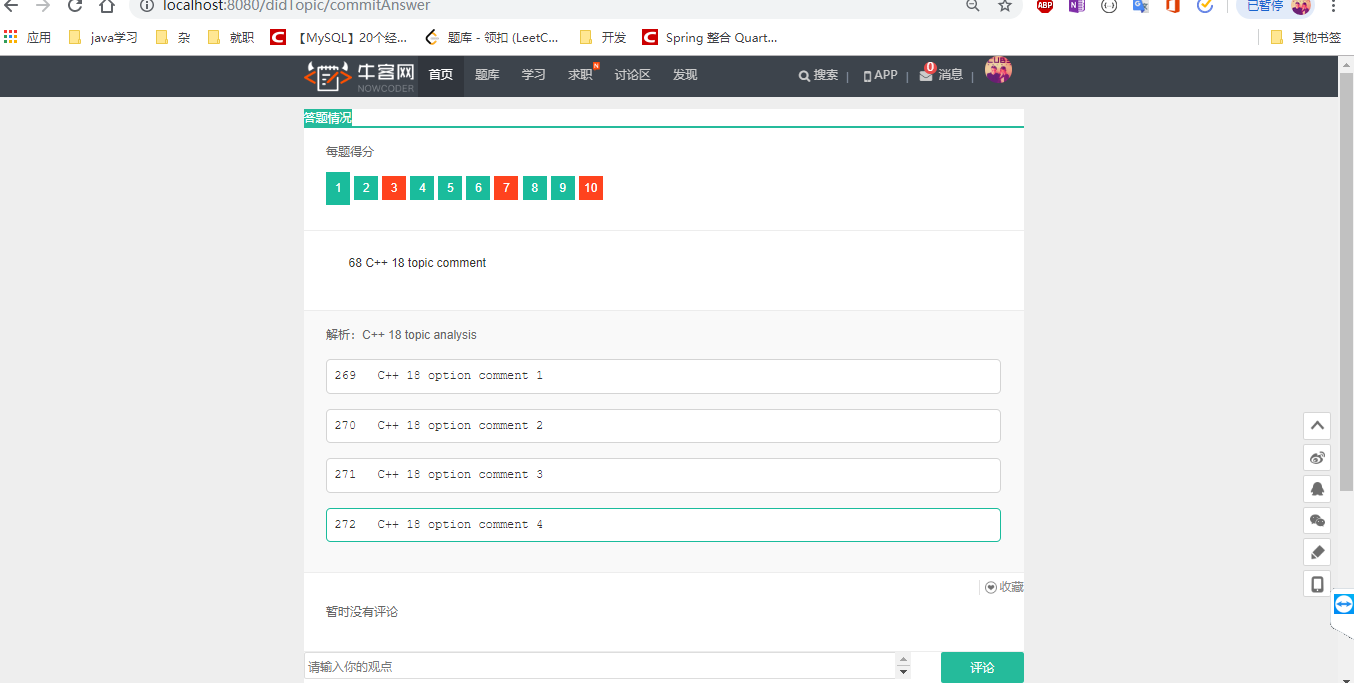
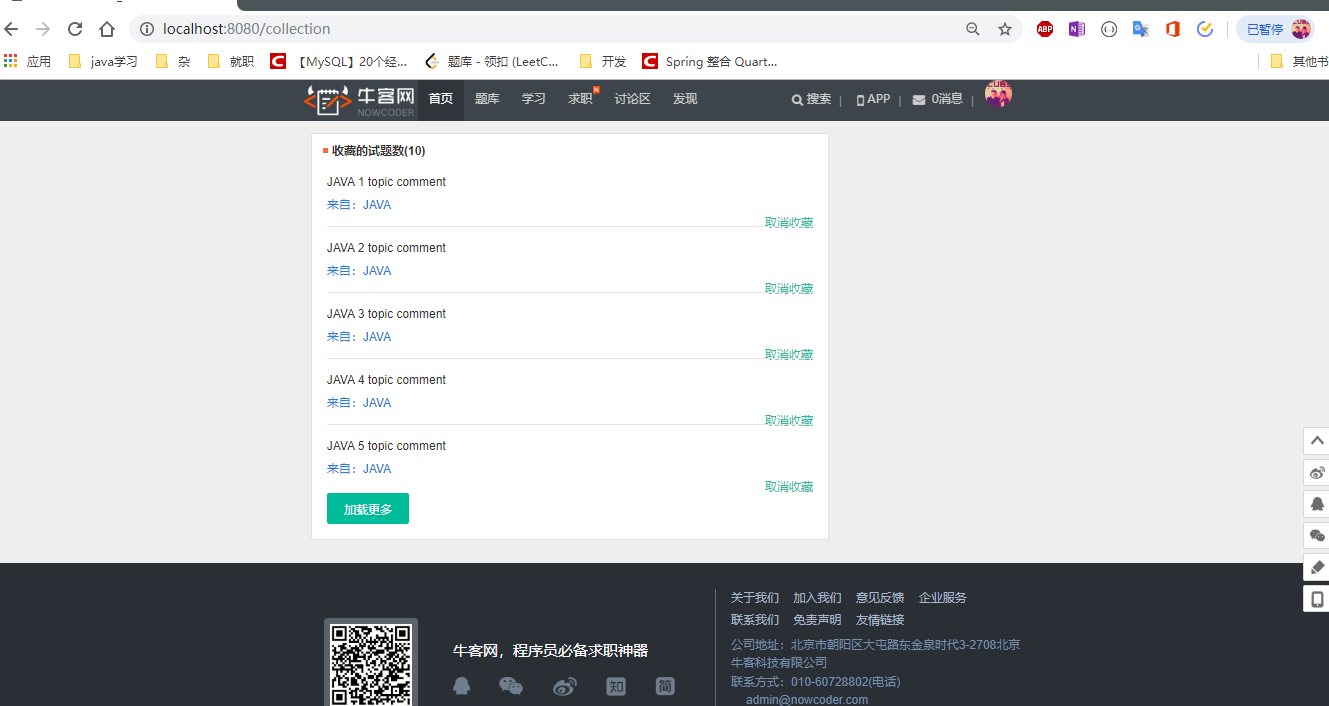


图5-6 练习详情展示

图5-7 收藏题目展示

5.5.3 程序逻辑

在详情页面显示收藏按钮，为收藏的题目显示收藏，收藏的题目即显示取消收藏按钮。收藏之后能够在首页的按钮中点击显示收藏的题目信息，每次显示5条，有更多的数据即显示加载更多，没有更多即显示已经到底了,同时分页显示的页面也能够对题目进行取消收藏。

5.5.4 后台处理设计

TbCollection收藏表，TbCollectionDao，CollectionController，CollectionService, CollectionServiceImpl, CollectionFinalTool

5.6 题目评论消息模块

5.6.1 模块描述

如图5-9 消息展示、图5-8 评论展示，用能够在题目详情页对题目进行评论，也能够删除评论，其他用户能够对评论了进行回复，能够一直回复关联，删除评论回复时，会同步的删除其评论下的所有回复以及评论。在其余的能够显示头部导航栏的页面下，如果有新的评论或者回复与自己有关，会及时的更新数量显示，点击消息提醒时，显示所有消息的简略详细，并且能够选择删除已读。

5.6.2 界面设计

图5-8 评论展示

图5-9 消息展示

5.6.3 程序逻辑

用户在题目下创建的评论记录在评论表里边，对评论的回复即记录在回复表里边，回复的回复就记录在回复表中自关联。只要回复和自己有关，即会在消息列表中每间隔30秒刷新一次消息的数量，点击消息能够进入到消息详情界面，用户可以手动删除消息记录。一直到用户看到了评论为止，或者用户手动删除消息时，消息的数量才会减少，最少为0。

5.6.4 后台处理设计

控制器层：CommentController，ReplyController，Service层 ReplyService，ReplyServiceImpl,静态工具类： ReplyFinalTool,Dao层： TbCommentDao, TbReplyDao,POJO：TbReply,TbComment

5.7 练习模块

5.7.1 模块描述

如图5-10 练习界面、图5-11 首页练习界面，用户选中练习的方式，练习的题目类型，练习的数量，系统开始生成题目，同时用户进行练习，如果练习未完成用户即退出网站，系统会记录到缓存中，提供用户下次进入系统时继续练习的入口。用户做完练习即可交卷，系统会自动记录分数，更新相应的统计信息

5.7.2 界面设计

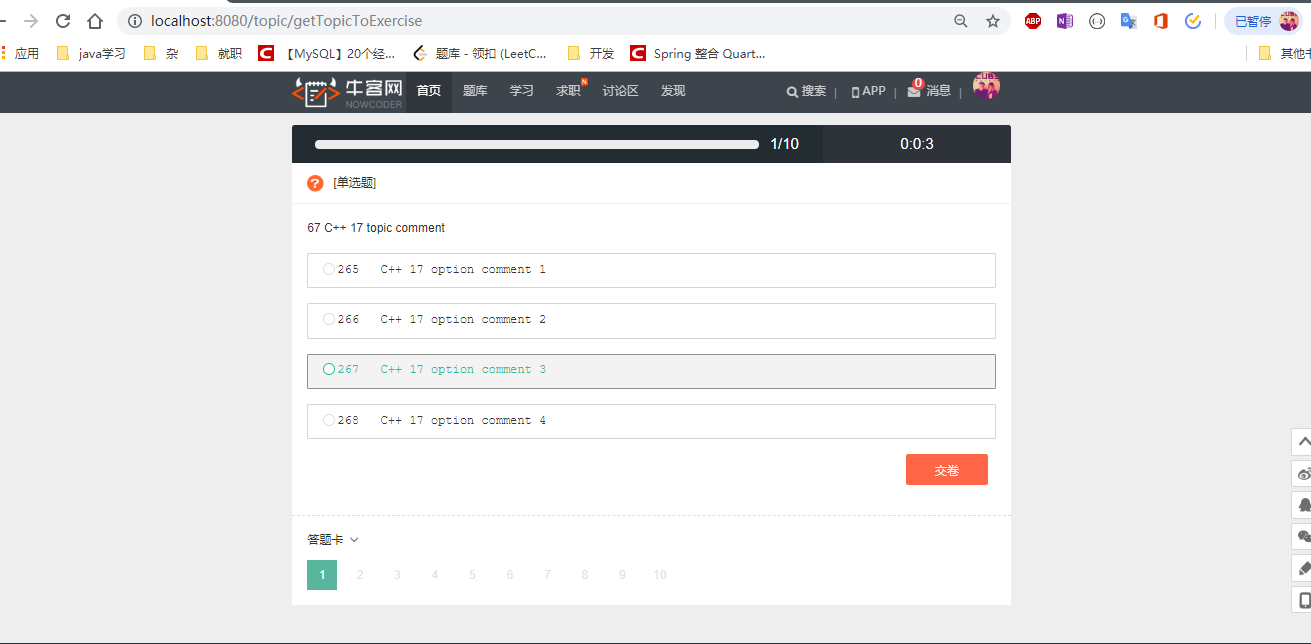
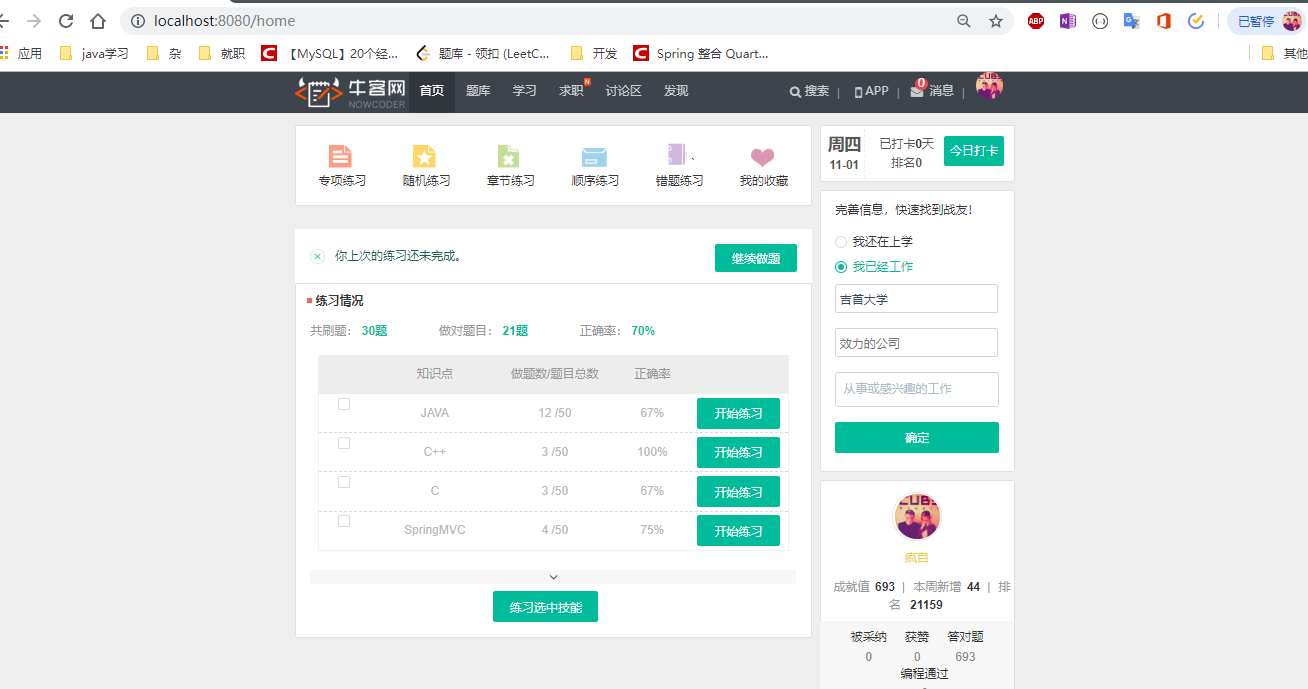


图5-10 练习界面

图5-11 首页练习方式界面

5.7.3 程序逻辑

用户可以选择随机练习，专项练习，错题练习三种方式练习，随机练习可以指定多个分类，同时随机选择题库中的试题，每个分类会尽可能的做到平均；专项练习即为顺序的练习一个题目分类；错题练习即练习所有的错题。练习的题目数量分为三种，五题，十题，二十题。

用户在选择一个题目的答案时即会发送数据回到后台，记录用户的练习记录，下次进入系统时即提醒用户进行继续练习。交卷之后系统会判断对错，并将所有的题目信息返回给页面，页面展示对错显示答案题目的详情页面。

5.7.4 后台处理设计

DidTopicController，TopicController，UserDidTopicUtil，Jedis，TopicService,TopicServiceImpl.DidTopicService,DidTopicServiceImpl,TbTopic,TbDIdTopic，TbDidTopicDao，TbTopicDao。

第6章 部署与应用

6.1 系统运行环境和系统部署

6.1.1系统运行环境

数据库服务端配置：

CPU：Intel® Core i5-4288U CPU @ 2.6 GHz \* 4

内存：8G

硬盘：300G SSD

应用端服务器配置：

CPU：Intel® Core i5-4288U CPU @ 2.6 GHz \* 4

内存：6G

客户端配置：

操作系统Windows 7 以上

建议内存2G以上

6.1.2系统部署过程

安装jdk，在服务器上安装jdk 1.8同时配置好环境变量，在安装tomcat 9.0.1，解压tomcat压缩包，在配置好tomcat的环境变量。最后安装MySQL 5.7.21，解压版安装，配置数据库初始化数据，同时修改默认密码，导入数据库的结构文件；

6.2 系统的主要应用(用户手册)

6.2.1登录登出功能

用户进入首页点击登录按钮进入登录界面，输入用户名，密码点击登录即可进入练习首页。

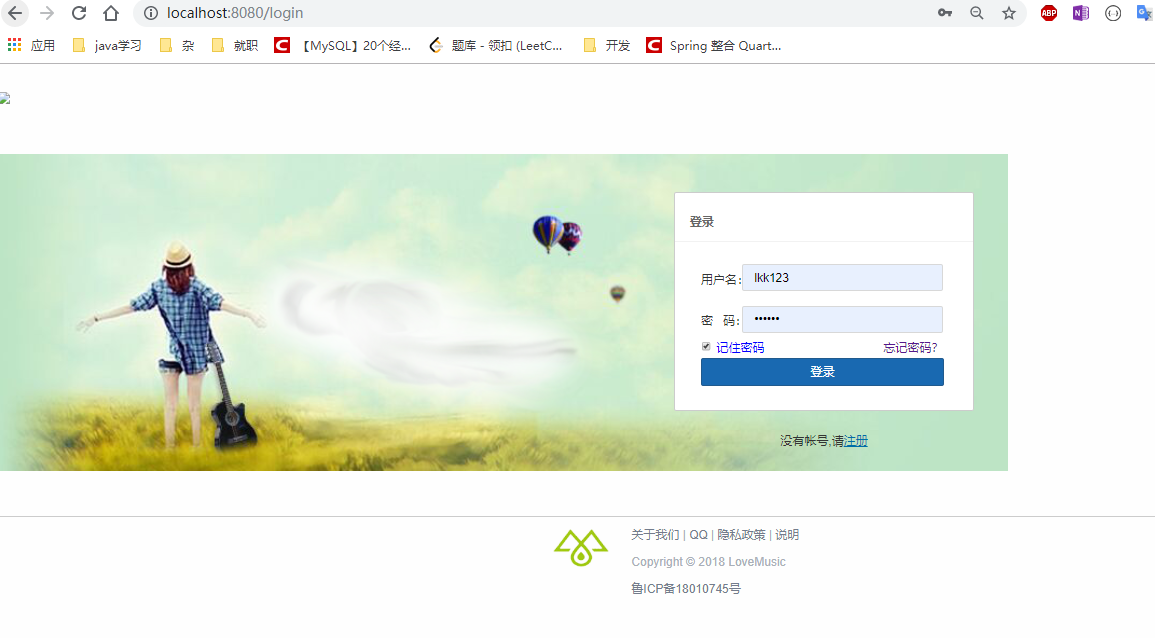


图6-1 用户登录界面

6.2.1练习功能

如图6-1 练习功能简介图，用户在练习首页点击开始练习，即弹出练习的方式，练习的题目数量，练习的分类，选中后即可以开始练习；点击练习选中技能即可以选中分类，进行随机练习。

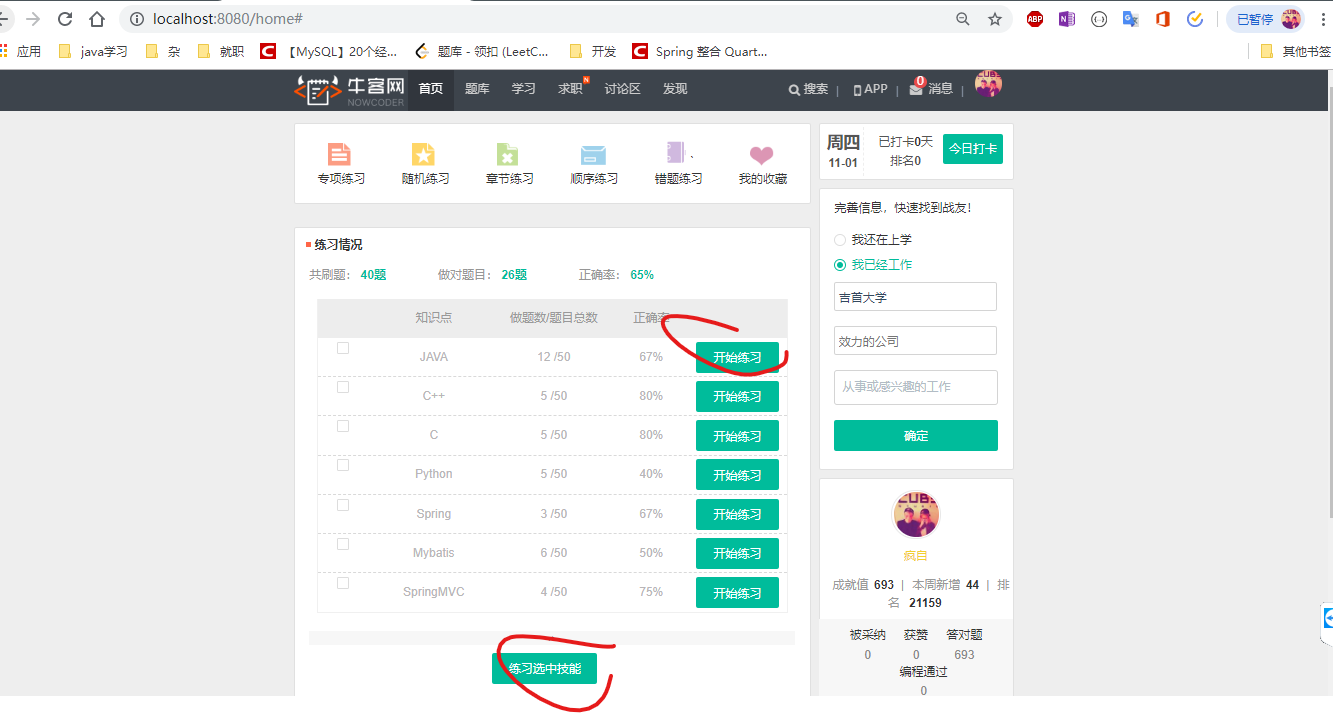


图6-2 练习功能简介图

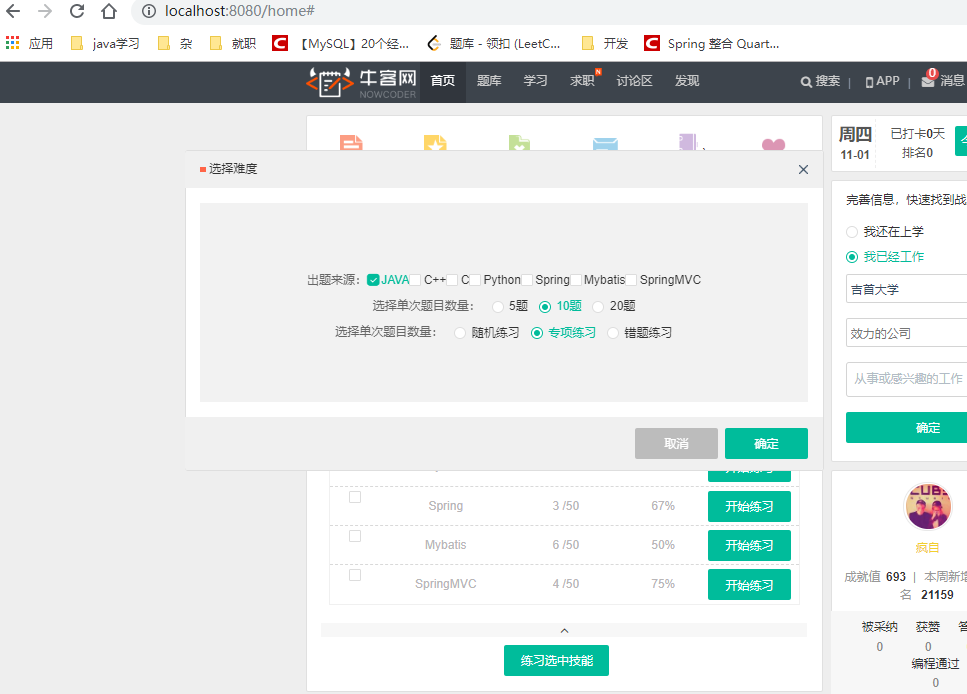


图6-3 练习功能简介图二

如图6-2 练习功能简介图二，用户在练习页面选中选项后，点击题号，即可以选择下一题，练习完成之后，点击交卷能够提交试卷，提交答案。

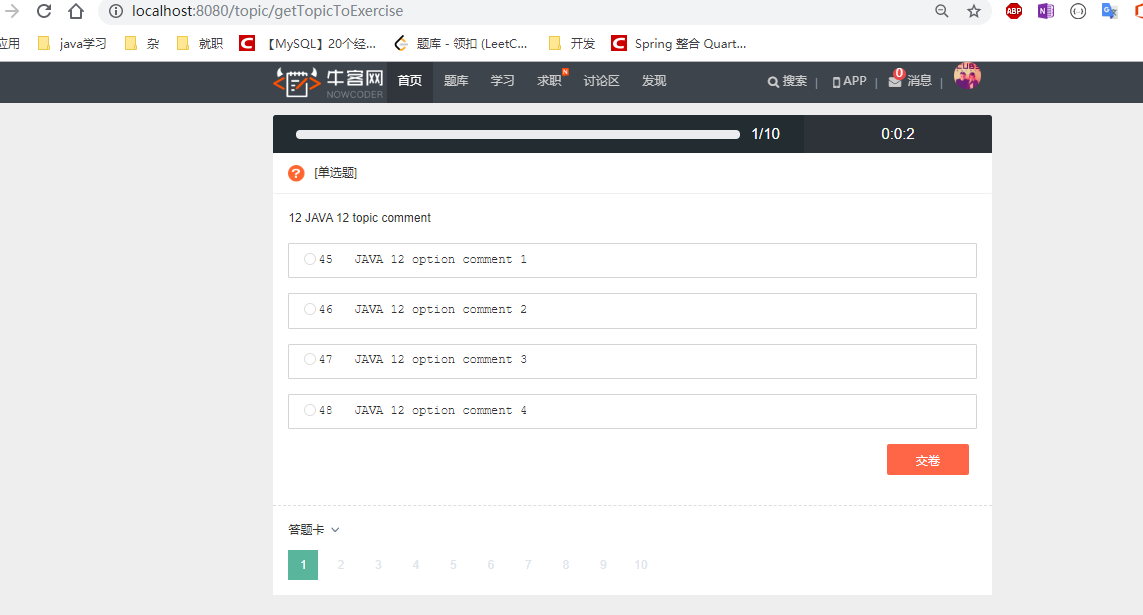


图6-4 练习功能简介图三

如图6-3 练习功能简介图三，提交完试卷以及答案之后，系统会返回题目的对错，以及解析，展示评论的内容，是否收藏等信息

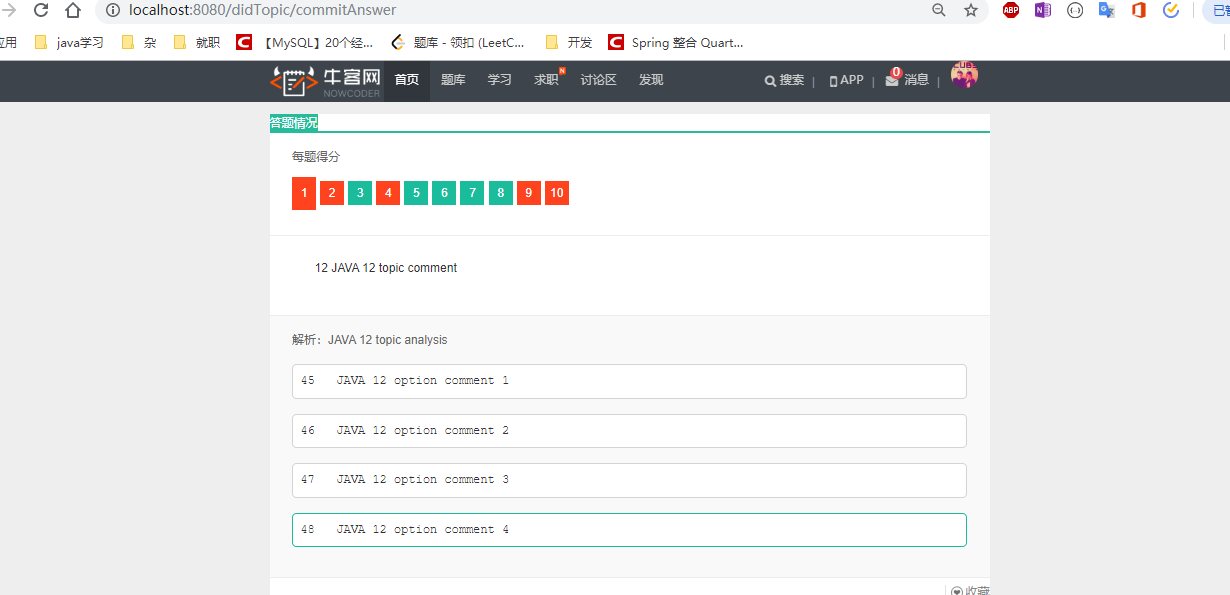


图6-5 练习功能简介图四

6.2.3 收藏功能

用户提交答案之后，能够显示收藏的按钮，点击收藏能够收藏题目，取消收藏能够收藏取消收藏题目。

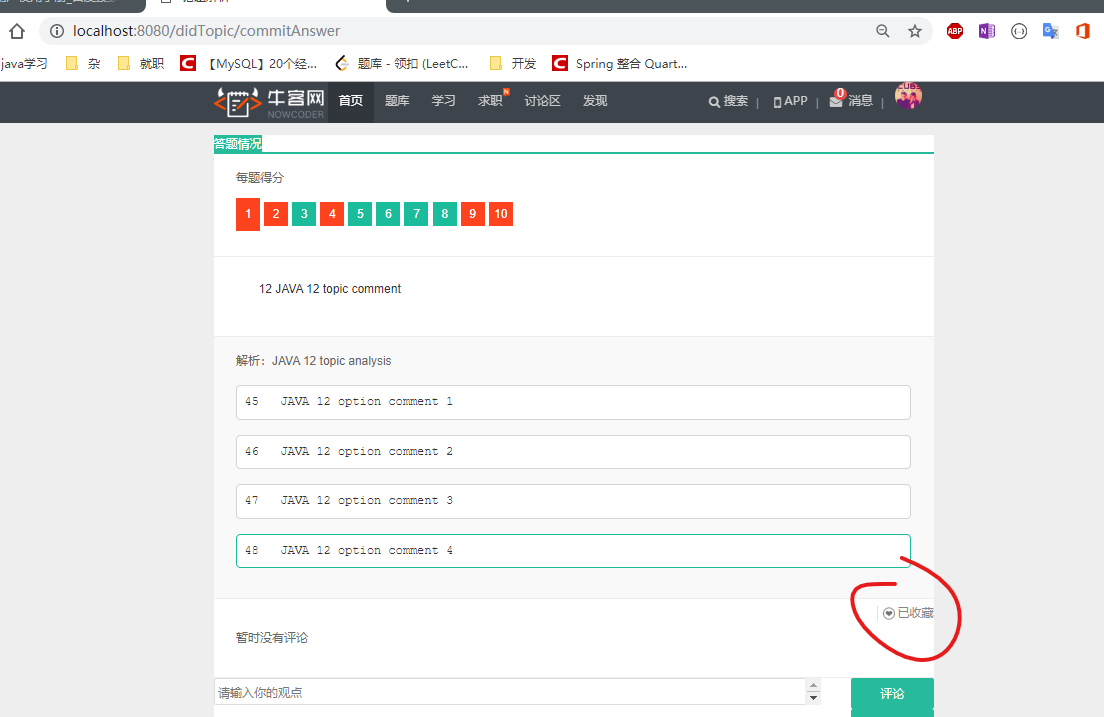


图6-6 收藏功能简介图一

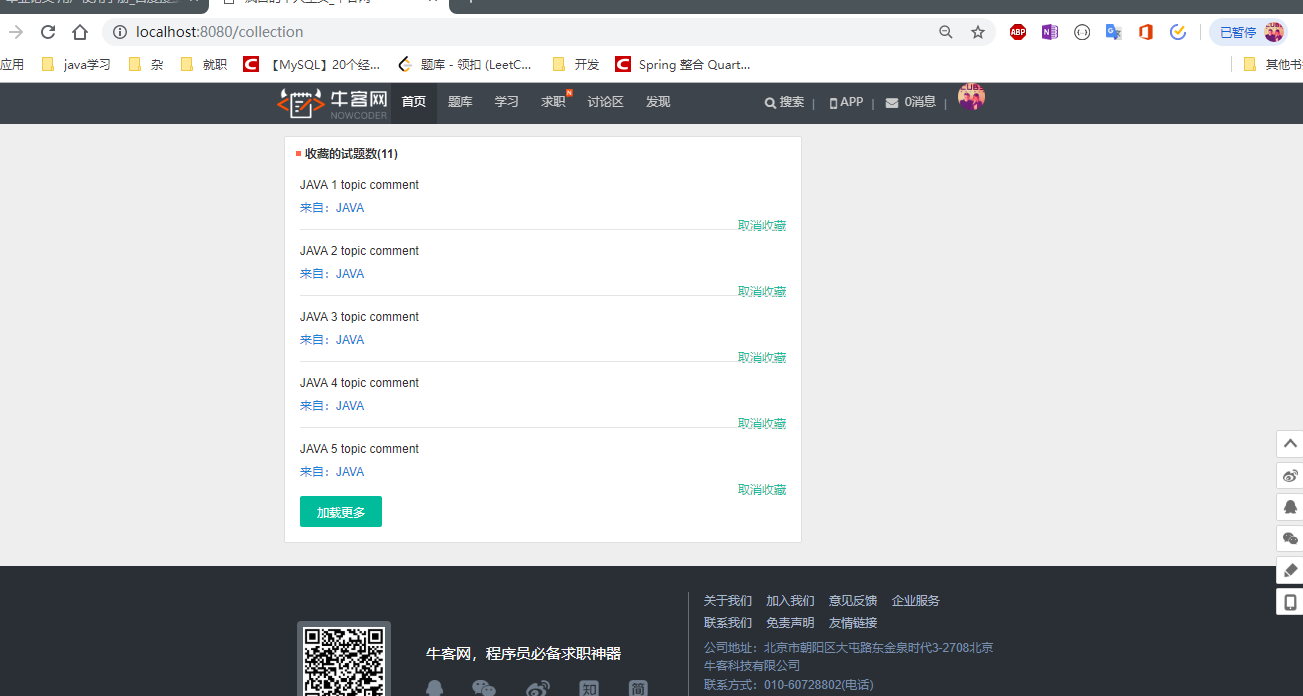
如图6-5 收藏功能简介图一，收藏题目之后，能够在首页点击我的收藏进入收藏的列表页，展示收藏的题目信息，以及分页展示。

图6-7 收藏功能简介图二

如图6-6 收藏功能简介图二，点击取消收藏能够删除收藏信息，点击加载更多能够加载更多的收藏题目信息。点击题目能够进入题目的详细信息界面，显示题目的详细信息。

6.2.4 评论功能

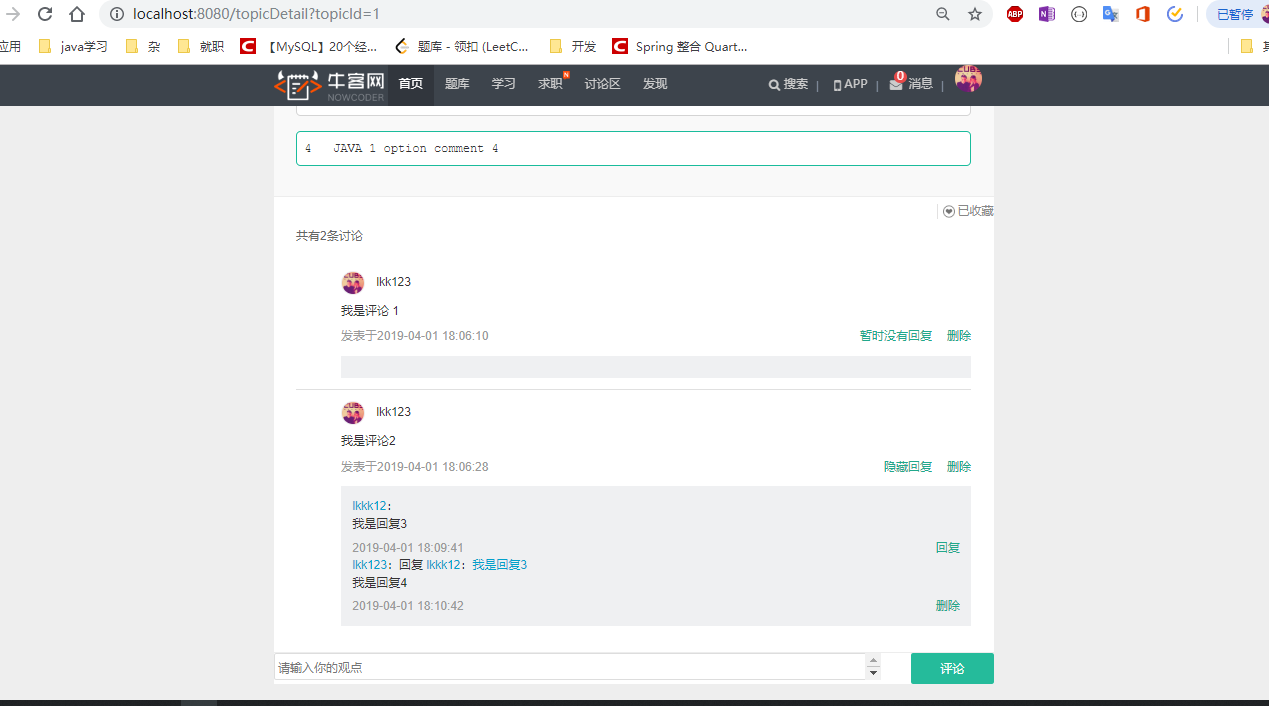


图6-8 收藏功能简介图一

如图6-7 收藏功能简介图一，用户在题目详情界面，能够对题目进行评论，对别人的回复进行回复，对自己的评论进行删除。

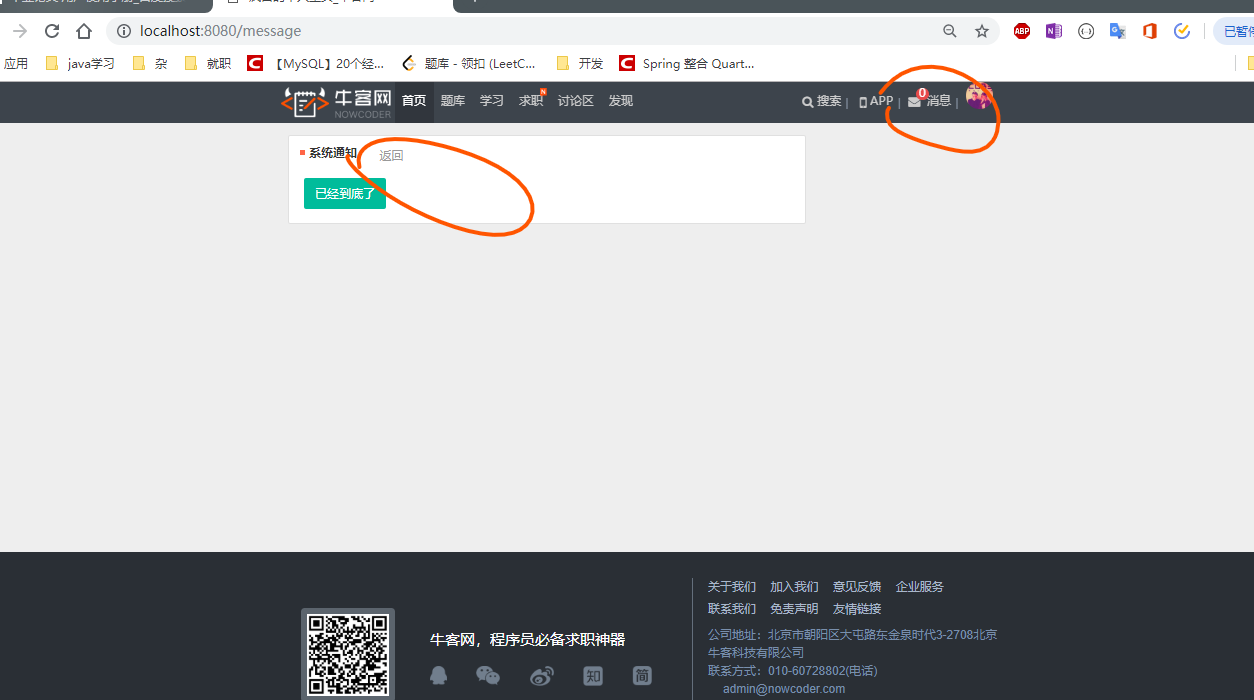


图6-9 收藏功能简介图二

如图6-8 收藏功能简介图二，用户收到了其他用户发送的评论消息与自己有关的，会在消息上显示数量，点击消息界面能够显示消息的详情，在详情界面能够删除消息提示，点击消息能够进入题目的详情界面。

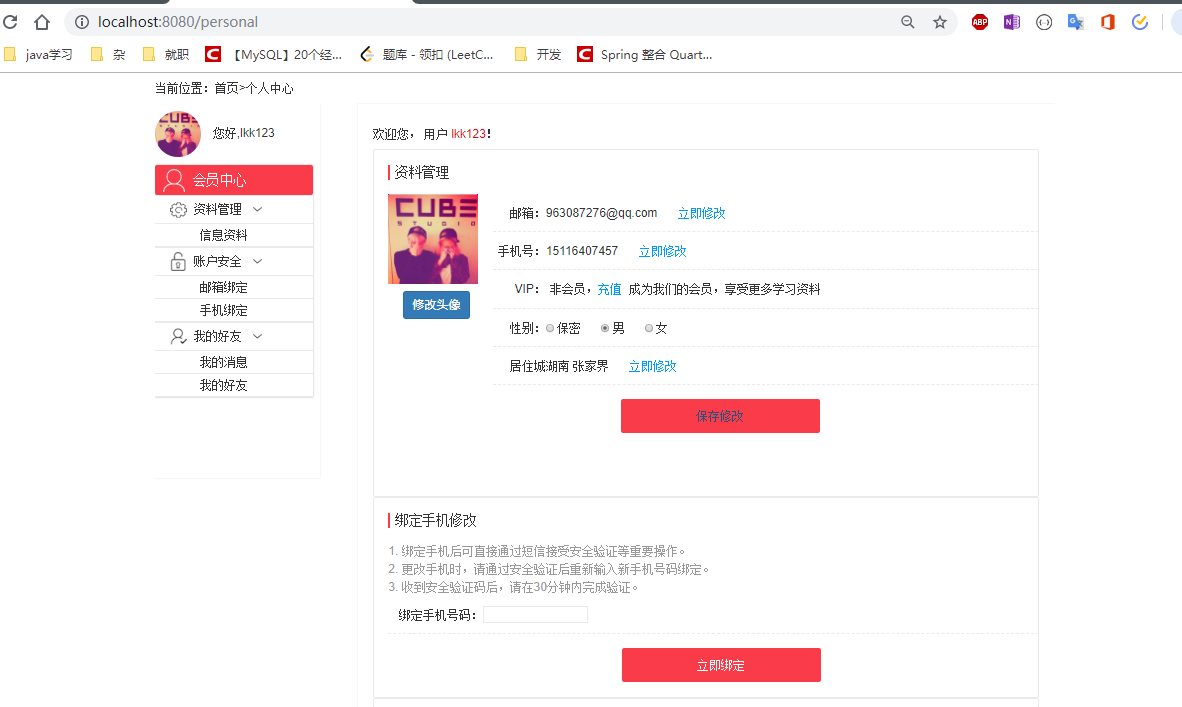
6.2.5 个人信息修改界面

图6-10 收藏功能简介图一

如图6-9 收藏功能简介图一，用户登录之后鼠标移动到自己的头像，会弹出退出登录，个人信息查看修改界面，能够查看自己的信息，并且修改自己的信息。

6.2.6 统计数据显示模块



图6-11 数据显示功能图一

如图6-10 数据显示功能图一，系统会展示用户所有的做题记录，总统计信息，以及每个分类下的统计信息。

第7章 结 论

所研发的系统，能够满足市场上大部分用户随时随地学习的需求，能够让用户在学习过程中享受网络、科技带来的进步，逐步抛弃之前的纸质试题，人工阅卷，由答题系统代替。在实际的使用中，用户将能够随时随地的练习，学习新知识，同时能够节省很多时间记录自己的错题，以及经典的知识点，并且能够在完成练习之后和现实中更多的人进行交流，讨论并且能够查看其他人的讨论结果，加速用户吸收并理解知识。

系统还存在很多需要改进持续维护的地方，用户只能直观的看到自己的练习统计正确率，不能知道该题目的正确率，不能和其他人形成对比，让用户缺乏前进的动力；题目的来源仅仅是系统提供，并没有通过网络吸取更多的优质题目，等。同时系统也做了相对应的优化，能够满足大批量用户在线浏览练习的目的，很多细节的地方做了相应的优化处理。相信在以后的开发和维护上，系统会变的更加健壮，满足用户的需求。

参考文献

1. 网络在线考试的探索与实践[J]. 庞建会,石鑫.  承德医学院学报. 2016(04)
2. 基于MySQL的在线考试系统分析与设计[J]. 邓春丽,陈胜迁.  轻工科技. 2016(08)
3. 基于SSH的在线答题系统的设计与实现[D]. 蔡恒跻.湖北工业大学 2017
4. 张峰. 应用SpringBoot改变web应用开发模式[J]. 科技创新与应用,2017(23):193-194.
5. 刘猛.基于移动Web的学者在线交流平台的研究与应用[D]. 博士，广东:华南师范大学,2016.
6. 王春晓. 基于改进蚁群算法的多目标枢纽机场停机位分配与系统开发[D].大连交通大学,2018.
7. 王安娜. 基于MVC设计模式的源代码检测平台的设计与实现[D].北京邮电大学,2018.
8. ]曹东升. 基于web的数据可视化方法研究与实现[D].西安电子科技大学,2017.
9. 滕宇峰. 基于Java Web的软件生产过程管理系统的设计与实现[D].东北大学,2015.
10. 刘喆. 基于B/S结构的实盘训练营系统的设计与实现[D].华中科技大学,2019.
11. 梁亚楠. 并发环境下数据竞争检测方法研究[D].河北科技大学,2019.

致 谢

本次毕业设计圆满的完成，是对整个自己大学期间所学知识的一种综合检验，在这里衷心感谢学院领导的督促，特别感谢我的指导老师曾明星教授，悉心指导论文修改以批注，仔细到错别字都能指正，非常感谢指导老师曾明星教授。同时也感谢学校无私的栽培，为我们提供了优质的教学环境。