

本 科 毕 业 论 文

论文题目：基于Springboot+vue的学科竞赛管理系统的设计与实现

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  | | |  |
| 姓名： | 钟雷 | 学号： | 20200802050240 | | |
| 导师： | 郭茗涵 | 职称： | 助教 | | |
| 专业： | 软件工程 | 提交日期： | | 2024年 5月27日 | |

**独创性声明**

本人呈交的学位论文，是在导师的指导下，独立进行研究工作所取得的成果，所有数据、图片资料真实可靠。尽我所知，除文中已经注明引用的内容外，本学位论文的研究成果不包含他人享有著作权的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确的方式标明。本学位论文的知识产权归属于培养单位。

本人签名：钟雷 日期：

**基于Spingboot+vue学科竞赛管理系统设计与实现**

**Design and implementation based on Springboot+VUE discipline competition management system**

作 者 姓 名： 钟雷

专 业： 软件工程

学 号： 20200802050240

指 导 教 师： 郭茗涵

完 成 日 期： 2024-05-27

兰 州 城 市 学 院

Lanzhou City University

# 摘 要

近年来，社会对高等教育的期望持续上升，学科竞赛作为高校人才培养体系中的关键要素，已经获得了广泛的关注和认可。学科竞赛不仅是衡量大学生对基础知识掌握程度和实践能力的重要指标，也是推动校园学术风气建设、提高教育质量的关键因素。其核心目的在于通过实践和创新的方式，培育学生的专业素养和综合能力。它不仅是专业知识教育的基地，也是学生个性发展和创新能力培养的重要场所[11]。

当前我国高校各类学科竞赛层出不穷，在各竞赛的比赛和准备过程中，学生综合运用大量所学专业知识并付诸实践，对师生专业水平的提高起到了非常积极的作用，学科竞赛的参与、指导、获奖等工作也被很多高校纳入到教学工作的管理评价体系中[1]。学科竞赛可以看作是学科竞赛的组织人员首先给出竞赛的题目，然后由学生在一定时间内以竞赛的方式来完成竞赛要求的一系列活动[23]。然而，竞赛信息的发布和传播存在障碍，使得参赛学生和指导教师难以获取准确的信息。其次，报名和评审流程繁琐，耗费大量人力和时间。最后，竞赛结果的发布和查询过程也存在不便，给参赛者带来了诸多不便。

随着互联网的普及，人们的生活方式发生了巨大改变[24]。构建一个高效、便捷的竞赛管理系统显得尤为必要，大学生学科竞赛管理系统的建立正是为了解决这些挑战。该系统旨在通过现代化的管理手段，优化竞赛信息的发布、报名、评审及结果查询等环节，从而提升学科竞赛的整体效率和质量。

本系统在设计上采用了前后端分离的开发模式，后端以Spring Boot和Mybatis-Plus技术为核心，保证了系统的稳定性和高效性；前端则运用了Vue和Element Plus等技术，使得系统界面更加美观、用户操作更加流畅。同时，前后端数据大多通过Axios以JSON格式进行数据传输。在数据存储和管理方面，系统采用了MySQL数据库，为数据的存储和查询提供了有力的支持。在安全方面，系统采用Redis+Jwt+SpringSecurity技术实现了有状态的 token 信息和用户认证，提升了系统的安全性。

在功能设计上，系统充分考虑了学生、教师和管理员等不同用户的需求。学生可以通过系统轻松查询各类竞赛信息，进行在线报名和成绩查询等操作；教师可以利用系统进行竞赛申请、参赛学生管理、成绩录入等工作；管理员则负责整体的用户管理、竞赛申请审核、数据统计和分析等任务。这样的设计使得系统更加贴近实际使用场景，满足了不同用户的多样化需求。

大学生学科竞赛管理系统的开发和应用，不仅为学科竞赛的组织和管理提供了强有力的技术支持，也为高校学风建设和人才培养质量提升注入了新的活力。

**关键词：学科竞赛；信息化管理；SpringBoot；数据库；ElementPlus**

**论文类型：软件工程**

# Abstract

In recent years, the society's expectation of higher education continues to rise, and discipline competition, as a key element in the talent training system of colleges and universities, has been widely concerned and recognized. Discipline competition is not only an important index to measure college students' mastery of basic knowledge and practical ability, but also a key factor to promote the construction of campus academic atmosphere and improve the quality of education. Its core purpose is to cultivate students' professional literacy and comprehensive competence through practical and innovative means. It is not only the base of professional knowledge education, but also an important place for students' personality development and innovation ability cultivation .

At present, various discipline competitions in Chinese colleges and universities are emerging in an endless stream. In the process of competition and preparation for each competition, students make comprehensive use of a large amount of professional knowledge and put it into practice, which plays a very positive role in improving the professional level of teachers and students. Participation, guidance and awards in discipline competitions have also been included in the management evaluation system of teaching work in many colleges and universities . The discipline competition can be regarded as the organizer of the discipline competition first gives the title of the competition, and then students complete a series of activities required by the competition in a certain period of time . However, there are barriers to the release and dissemination of competition information, making it difficult for participating students and instructors to obtain accurate information. Secondly, the registration and review process is cumbersome and consumes a lot of manpower and time. Finally, the process of releasing and querying the results of the competition is also inconvenient, which brings a lot of inconvenience to the participants.

With the popularization of the Internet, people's lifestyle has undergone great changes. It is necessary to construct an efficient and convenient competition management system, which is designed to solve these challenges. The system aims to optimize the release, registration, review and result inquiry of competition information through modern management means, so as to improve the overall efficiency and quality of discipline competition.

The system adopts the development mode of separating front and back end, and the back end takes Spring Boot and Mybatis Plus technology as the core, which ensures the stability and high efficiency of the system. The front-end uses technologies such as Vue and Element Plus to make the system interface more beautiful and the user's operation more smooth. At the same time, the front and back end data is mostly transmitted in JSON format through Axios. In terms of data storage and management, the system adopts MySQL database, which provides strong support for data storage and query. In terms of security, the system adopts Redis+Jwt+SpringSecurity technology to realize stateful token information and user authentication, which improves the security of the system.

In the functional design, the system fully considers the needs of different users such as students, teachers and administrators. Students can easily query all kinds of competition information through the system, online registration and result query operations; Teachers can use the system to apply for competitions, manage students, record scores and other work; The administrator is responsible for the overall user management, competition application review, data statistics and analysis. This design makes the system more close to the actual use scenario, to meet the diverse needs of different users.

The development and application of university students' discipline competition management system not only provides strong technical support for the organization and management of discipline competition, but also injects new vitality into the construction of academic style and the improvement of talent training quality.

**Key Words：Discipline competitions; Information management; SpringBoot;Database; ElementPlus**

**Paper type: Software engineering**

# 目 录

[摘 要 I](#_Toc167136894)

[Abstract II](#_Toc167136895)

[目 录 IV](#_Toc167136896)

[1 引言 1](#_Toc167136897)

[1.1 系统开发的背景 1](#_Toc167136898)

[1.2 本课题的研究意义 2](#_Toc167136899)

[1.3 本课题的基本内容 3](#_Toc167136900)

[1.4 本课题的重点和难点 3](#_Toc167136901)

[2 系统分析 5](#_Toc167136902)

[2.1 系统概述 5](#_Toc167136903)

[2.1.1 系统的总体功能需求和性能要求 5](#_Toc167136904)

[2.1.2 系统开发的框架 5](#_Toc167136905)

[2.2 可行性分析 6](#_Toc167136907)

[2.2.1 经济可行性 6](#_Toc167136908)

[2.2.2 技术可行性 6](#_Toc167136909)

[2.2.3 操作可行性 6](#_Toc167136910)

[2.2.4 法律可行性 7](#_Toc167136911)

[2.3系统功能分析 7](#_Toc167136912)

[2.3.1 功能性需求分析 7](#_Toc167136913)

[2.3.2 非功能性需求分析 10](#_Toc167136914)

[3 需求分析 11](#_Toc167136915)

[3.1业务流程 11](#_Toc167136916)

[3.2系统范围 12](#_Toc167136917)

[3.3 数据流图 13](#_Toc167136918)

[3.4 数据字典 15](#_Toc167136919)

[4 总体设计 19](#_Toc167136920)

[4.1 系统总体结构设计 19](#_Toc167136921)

[4.2 系统的接口设计 19](#_Toc167136922)

[4.2.1 用户接口 19](#_Toc167136923)

[4.2.2 外部接口 20](#_Toc167136924)

[4.2.3 内部接口 20](#_Toc167136925)

[4.3 软件体系结构 21](#_Toc167136926)

[5详细设计 22](#_Toc167136927)

[5.1 功能模块设计 22](#_Toc167136928)

[5.2 输入/输出格式设计 23](#_Toc167136929)

[5.3 人机对话设计 23](#_Toc167136930)

[5.4 界面设计 24](#_Toc167136931)

[5.4.1 用户界面设计 24](#_Toc167136932)

[5.4.2 菜单设计 32](#_Toc167136933)

[5.4.3 对话框设计 32](#_Toc167136934)

[5.4.4 窗口选项卡界面设计 32](#_Toc167136935)

[5.5数据库设计 33](#_Toc167136936)

[5.5.1 概念结构设计 33](#_Toc167136937)

[5.5.2逻辑结构设计 37](#_Toc167136941)

[5.5.3 物理结构设计 38](#_Toc167136943)

[6系统实现 42](#_Toc167136944)

[6.1实现工具 42](#_Toc167136945)

[6.2技术框架主要功能模块流程和编码 42](#_Toc167136946)

[7系统的测试与维护 46](#_Toc167136947)

[7.1 测试用例与测试结果 46](#_Toc167136948)

[7.2 系统维护 47](#_Toc167136949)

[结论 49](#_Toc167136950)

[参考文献 50](#_Toc167136951)

[致谢 52](#_Toc167136952)

# 1 引言

## 系统开发的背景

大学生学科竞赛是提升大学生综合能力和专业素质的重要手段和途径，也是对传统课堂教学的重要补充。学科竞赛在教育教学改革和创新人才培养中发挥重要作用，越来越受到广大高校师生的关注与重视。大学生学科竞赛活动不仅有利于提升大学生的专业素养, 也有利于提升大学生的创新、实践能力、竞争意识与团队精神[15]。

《国家中长期教育改革和发展规划纲要(2010－2020年)》明确提出高等教育要“培养一批拔尖创新人才”、“促进科研与教学互动、与创新人才培养相结合”[2]。信息技术的发展为教育管理提供了新的解决方案。特别是近年来，云计算、大数据、人工智能等技术的应用，为教育系统的改革开辟了新的道路[12]。随着教育改革的不断深化和学科竞赛的广泛普及，学科竞赛已成为高等教育中培养学生创新能力和实践技能的关键环节。学科竞赛作为创新人才培养的重要载体，是激发学生创新精神的有效途径，也是培养创新型人才的有效手段[3][4]。同样，在当今高等教育体系中，学科竞赛也已成为培养学生综合素质和创新能力的重要平台。它不仅为学生提供了展示自我、锻炼能力的机会，还促进了学术交流和科研创新。

从市场需求的角度审视，随着社会对人才综合素质和创新能力需求的日益增长，学科竞赛活动正逐渐成为教育领域关注的焦点。众多高校和教育机构开始积极举办各类学科竞赛，旨在通过这些活动激发学生的创新思维和实践技能，从而培养出适应未来社会发展的高素质人才。

从学术研究的角度出发，学科竞赛管理系统的研究不仅具有显著的学术价值，更蕴含着深远的实践意义。深入探讨和实现竞赛管理系统的设计，可以揭示信息技术在高等教育领域的应用潜力和演进方向，为相关领域的研究提供宝贵的经验和指导。此外，学科竞赛管理系统的研究还能为高等教育的教学革新和人才培养策略注入新的活力，开拓新的视野。

当今，为了更好地推动学科竞赛的发展，高校通常从多方面着手。首先，加大学科竞赛支持力度，在教学和实践环节培养学生创新创业意识，提升团队合作以及解决实际问题的能力。同时加强指导队伍的建设，让具有丰富经验的校内教师和校外专家参与指导。另一方面，持续优化、完善学科竞赛的组织管理制度，为学科竞赛的开展保驾护航[5][6]。学科竞赛是重要的教育手段，能够促进高校学科教育的改革与发展，为学生提供展现自我的平台。有效的学科竞赛管理系统能够提高竞赛的组织效率，使学生更好地参与其中[7]。然而，随着各类学科竞赛的层出不穷和参与学生数量的急剧增加，竞赛管理面临着诸多挑战。传统的学科竞赛管理方式往往存在操作繁琐、效率低下等问题，无法满足当前竞赛管理的需求。因此，开发一个高效、便捷的学科竞赛管理系统具有重要的现实意义。

最后，学科竞赛管理系统的研究背景深植于学生和管理者的切实需求之中。对于学生群体，他们迫切需要一个能够提供便捷访问、快速报名、准确查询等功能的统一平台，以便及时获取竞赛信息、参与竞赛活动、跟踪个人成绩。而对于竞赛的组织者和管理者，他们则需要一个能够提升工作效率、实现智能化管理的系统，以便更好地进行竞赛的组织、评审、成绩发布等关键环节。

## 1.2 本课题的研究意义

学生学科竞赛是培养大学生综合素质和创新精神的有效手段和重要载体，是高校人才培养的重要标志之一[21]。学科竞赛是培养学生创新精神和实践能力的重要途径，通过参与竞赛，学生可以锻炼自己的学术能力和实践能力，培养自己的团队协作精神和创新精神。而学科竞赛管理系统的研究和开发也需要大量的创新人才和实践经验。因此，系统的研究和开发不仅有助于推动学科竞赛的发展，还有助于培养创新人才和实践能力，为国家的科技创新和人才培养做出贡献。

随着学科竞赛报名、备赛过程、赛后总结等环节产生的数据日益增长，同时参赛师生数量的逐年增加以及跨专业、跨学院、甚至跨校报名等多种新模式学科竞赛的出现，给传统的学科竞赛管理工作带来了巨大的挑战。学科竞赛管理系统的开发旨在解决传统管理方式中的问题。该系统将利用现代信息技术手段，实现竞赛信息的快速发布、选手报名的便捷处理、管理员对竞赛信息的高效审核等关键功能，为学科竞赛活动提供有力的支撑。通过该系统，不仅可以大大提高竞赛的管理效率，减轻管理人员的负担，还能够为参赛者提供比较优质的服务体验，从而推动学科竞赛的健康发展，从而推动学科竞赛的健康发展。

再者，学科竞赛管理系统对于促进学术交流与合作具有重要意义。学科竞赛不仅可以提升学生的综合素质和就业竞争力，提高工程人才培养的质量，还可以为高校赢得良好声誉，提升高校的知名度和影响力[16]。同时，学科竞赛是学术交流与合作的重要平台。通过竞赛，不同学校、不同专业的学生可以互相学习、互相借鉴。学科竞赛管理系统可以提供一个在线平台，方便参赛者上传作品、分享经验、交流心得。这不仅有助于参赛者之间的学术交流和合作，还有助于推动学科知识的传播和普及。

综上所述，研究和开发学科竞赛管理系统具有重要的现实意义和深远的发展价值。学科竞赛是培养学生创新精神和实践能力的重要途径。通过参与竞赛，学生可以锻炼自己的学术能力和实践能力，培养自己的团队协作精神和创新精神。它不仅能够提升竞赛管理的效率和质量，促进学术交流与合作，还能推动教育信息化的发展，培养创新人才和实践能力。因此，我们应该加强对学科竞赛管理系统的研究和开发，为高等教育领域的发展做出更大的贡献。

## 1.3 本课题的基本内容

需求分析是学科竞赛管理系统设计的关键步骤之一，需要充分了解用户需求，明确系统的目标和范围[8]。在需求分析阶段，通过自身全方位多角度的实际调查。本研究首先熟悉并了解了学科竞赛管理系统的核心功能和用户需求，然后明确了该系统应支持的功能，包括学生信息管理、教师信息管理、竞赛审核等。除此之外，最后还需要考虑到系统的安全性、可扩展性和用户的友好性等非功能性需求，确保系统能在不同的环境中稳定运行。

在系统设计阶段，本论文采用了模块化的设计方法，将该系统分为学生、教师和管理员模块，每个模块都将包含其功能的实现。同时，系统的整体架构基于模型-视图-控制器（MVC）模式，以确保代码的可维护性和可扩展性[9]。此外，系统设计还包括了数据库的设计，使用关系型数据库管理系统MySQL和非关系型数据库Redis来存储和管理数据，以及使用ElementPlus组件简化前端开发过程和Spring Boot[20]框架简化后端的开发和部署过程。

系统实现阶段，详细介绍了使用Java语言和Spring Boot框架进行开发的具体过程，还包括后端逻辑的编写、数据库的连接和操作以及前端界面的实现。此外，为了确保用户和系统的安全性，系统实现过程中也加入了多种安全措施，如用户密码加密和验证用户维持状态登录的凭证。

## 1.4 本课题的重点和难点

重点：

1. 需求分析：为了更好的设计系统，访问了大量的学科竞赛网站，从用户的角度出发，去实际体验各个学科竞赛网站的各项功能和操作流程，以了解实际情况的需求并解决需求。

2. 软件建模：将数据库表中的实体属性封装成 ER 图例，方便数据表的建立和系统的开发。

3. 维持状态登录：用户进行登录后，后端会生成一个具有有效时间的token以后续访问需要认证的资源时使用。当用户在token有效时间内退出登录时，我们自然的希望token失效，但由于token一经颁发就不可修改，所以需要借助某种工具来使用某种方法对token进行状态管理，这时我就需要Redis数据库了。Redis基于内存存储，因此读取和写入操作都非常快。当用户尝试访问需要身份验证的资源时，验证token的过程可以迅速完成。如果用户登出或token被撤销，可以在Redis中迅速将其添加到黑名单中，从而确保该token不再有效，这样就可以通过Redis提供高效的token验证和存储服务了。

4. UI设计：系统界面的UI设计页是系统设计的重中之重，良好清晰的界面设计不仅可以提高用户的体验，还可以吸引更多的用户使用该网站。

难点：

1. 系统的UI设计：在设计系统的UI时，尽管前端设计表面看似简单直观，但在实际操作中，还是会面临诸多问题。系统前端采用了响应式设计，所以需要思考如何优化系统与用户之间的人机交互，从而为用户提供良好的体验。

2. 富文本数据传输：对于小型化的网站，对于图片和文件等数据的传输与存储的方式有两种，一是利用云端服务如阿里云的对象存储服务（OSS）进行上传，然后通过外部接口获取存储的数据。另一种是存储在本地的服务器端的静态资源目录下。选择后者的方式进行富文本数据传输的，这种方法传输的难点是，前端通过怎样的数据格式传输到后端，然后后端又通过怎样的数据格式来接受。

3. 数据库的设计：数据库系统设计既是系统设计的重点又是难点。每当在设计数据库的表结构时，为了优化数据存储效率和减少冗余，需要考虑是否能将一张表合理的拆分为多张表，并通过数据库的外键将它们紧密关联起来。这样不仅有助于提升数据的规范性和完整性，还能增强系统的灵活性和可维护性。

4. 用户认证与访问拦截：本系统使用Redis+Jwt+SpringSecurity技术实现了有状态的 token 信息和用户认证。在服务器端使用Spring Security来配置访问控制策略，定义哪些URL或资源需要认证才能访问(如获取用户信息)。当用户登录时，首先检查账号和密码的正确性，如果正确，服务器则生成一个包含用户信息的token，并将其返回给客户端存储；否则，返回用户名或密码错误。当用户尝试访问受保护的资源时，前端会将token加入请求头中以让Spring Security来检查认证（例如，检查其是否过期、是否已被撤销等）。如果用户已认证且有权访问该资源，则允许访问；否则，拒绝访问并返回适当的错误消息。当用户退出登录时，把用户的token用Redis加入黑名单中，以免退出登录的token被用来访问用户的私密信息。

# 2 系统分析

## 2.1 系统概述

### 2.1.1 系统的总体功能需求和性能要求

功能需求：本系统是一个适用于高校的学科竞赛管理系统。对于管理端，管理端需提供一套完整的审核流程，用于审查和批准教师提交的竞赛信息，确保信息的准确性和合规性。同时系统应允许管理员对教师和学生用户的账户进行管理以实现信息更新，管理端也应能够更新和维护首页的轮播图，用于展示重要通知或特色竞赛，吸引用户关注。对于教师端，教师端应允许教师提交新的竞赛信息，包括竞赛详情、规则、时间安排等。教师也应能够方便地输入、查看和更新学生的成绩信息，以及管理自己发布的竞赛，包括编辑或查看竞赛参与情况。对于学生端，系统应提供简单明了的报名流程，使学生能够轻松报名参加感兴趣的竞赛，管理自身信息、追踪自己参与的竞赛状态以及为学生提供查看个人获奖记录的功能。

性能要求：本系统核心功能是竞赛报名和竞赛信息查询等，需要支持大量用户的同时访问，所以对后端服务的稳定性具有较高要求。系统后端采用先进的架构设计，确保在用户访问量剧增时，仍能保持高性能和快速响应。异常处理，系统内置了健全的异常处理机制。当操作触发异常时，系统能够优雅地处理这些异常，避免因为单一操作的错误导致整个系统运行中断。前端页面设计注重简洁性和直观性，以提升用户的操作体验。通过合理的布局、清晰的指引和直观的反馈，即使是初次使用的用户也能快速上手。安全性是任何系统设计和实现的核心关切，它直接关系到用户信息的保密性、完整性和可用性，以及整个系统的可靠性。在构建学科竞赛管理系统时，通过采用Spring Security框架和JWT技术对本系统在安全性和用户认证方面提供了全面的保护，确保了系统的安全稳定运行，保障了用户信息的安全，从而为学科竞赛的顺利进行提供了坚实的技术支撑。

### 2.1.2 系统开发的框架

本系统以Spring Boot作为后端支撑，利用其强大的依赖注入和RESTful服务能力，构建健壮的后端服务。系统的整体架构基于模型-视图-控制器（MVC）模式，以确保代码的可维护性和可扩展性[13] 。在 Spring Boot 应用程序中，通过添加 spring-boot-starter-web 依赖来自动配置 Spring MVC，进而快速构建基于 Java 的 Web 应用程序。SpingMVC三层架构图如图2.1所示。

前端则采用Vue.js框架结合Element Plus UI库，打造响应式且用户友好的界面。Vue 3在继承Vue 2的优秀特性的基础上，进一步通过Composition API的引入，为前端开发领域带来了创新的动力。它不仅在性能上取得了显著进步，还通过模块化的管理方式，允许开发者根据需要选择功能模块，减少应用体积，加快加载速度。Vue 3也加强了对TypeScript的支持，提升了代码的可维护性和可读性[14]。

通过Spring⁃Boot+Vue前后端分离的架构，可以完美地使前端和后端代码分离开来，对开发人员的门槛降低了很多，前端是前端，后端是后端招聘模式，服务端和客户端进行了分离，很好地解决了代码的耦合度，方便以后对系统进行维护和优化[18]。

### 

图2.1 SpingMVC三层架构图

## 2.2 可行性分析

### 2.2.1 经济可行性

本系统是个人开发的小型管理系统，使用的技术框架如 SpringBoot、VUE 等，都是开源技术，数据库用的是 MYSQL 社区版本，开发工具如 IDEA等都是试用版本，经济投入少，因此，在经济方面是可行的。

### 2.2.2 技术可行性

本系统采用前后端分离的开发模式。后端采用 SpringBoot，可以快速创建一个独立的、生产级别的基于 Spring 框架的 Web 应用，前端采用 VUE3+ElementPlus，可以快速构建页面。无论前端还是后端，都可以对系统进行快速开发。因此，在技术上是可行的。

### 2.2.3 操作可行性

从操作可行性角度分析，本系统界面采用科学合理的布局，并且基于已有的学科竞赛网站进行过操作上的优化，确保用户能够快速熟悉和有效使用系统，使操作的可行性更高。

### 2.2.4 法律可行性

系统开发过程中所采用的技术框架均为开源技术，这些技术在遵守相应的开源许可协议的前提下，可以自由使用和分发。并且严格遵循所有相关技术的开源许可条款，确保在开发和使用过程中不侵犯任何专业版权，从而避免了潜在的法律风险。

## 2.3系统功能分析

### 2.3.1 功能性需求分析

对于本系统的需求分析首先是从用户的需求分析进行调研，根据用户实际需求定制所需的功能，然后对系统功能需求分析展开讨论，实现真正意义上的服务用户[27]。该系统的设计充分考虑了不同用户群体的需求，总体上划分为三个核心角色：管理员、教师和学生。每个角色都拥有特定的功能和权限，以确保系统的高效运行和数据的安全性。系统的总体功能架构如图2.2所示。

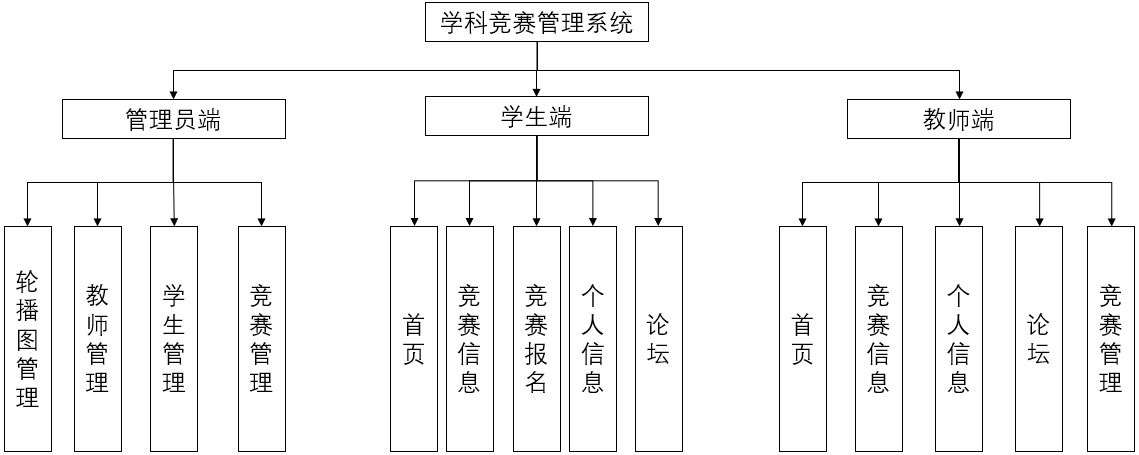


图2.2系统总体功能图

#### 2.3.1.1管理员需求分析

根据前面对学生学科竞赛管理系统的需求分析[22]，为了确保学科竞赛管理流程的高效、透明，并能实时更新最新的竞赛相关信息，本系统为管理员提供了全面的管理功能。管理员端包括轮播图管理、用户管理、竞赛管理和成绩管理等。当管理员登录后可管理员可以从已发布的竞赛信息的图片中添加到轮播图中展示，以便用户能及时看到更好的竞赛内容；用户管理，管理员对教师和学生两个端的信息进行管理；竞赛管理，当竞赛项目审核过后，竞赛项目的信息将会发布到本系统的竞赛查询功能中，以便其他用户能够及时查询竞赛信息，同时管理员端也可以对竞赛进行管理；成绩管理，当竞赛结束后，教师端会进行成绩打分，而管理员端就可以对这些成绩进行管理。其中，管理员的用例分析描述如图2.3所示。

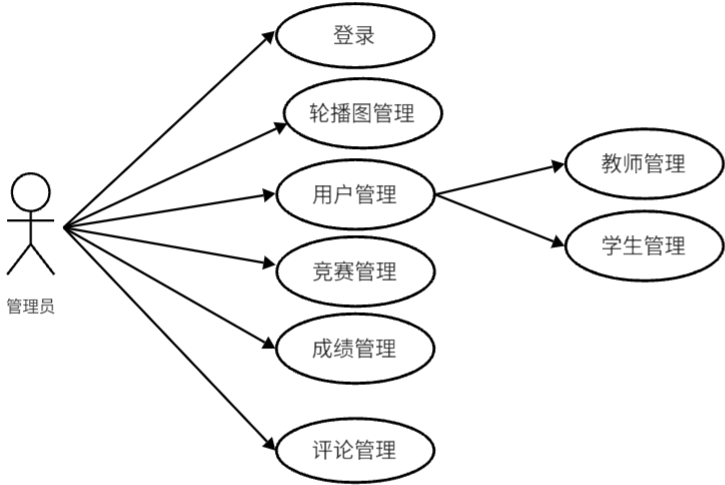


图2.3管理员用例图

#### 2.3.1.2 教师需求分析

本系统提供了教师端。这包括登录功能，让教师能够安全访问自己的账户；个人中心，确保教师能够及时更新个人资料；竞赛申报，使教师能够随时申报自己的竞赛项目，但是这需要管理员端对提交的竞赛内容进行审核；竞赛管理，当竞赛项目审核过后，教师端就能够看到自己的竞赛项目并对后续的竞赛进行管理了。其中，教师的用例分析描述如图2.4所示。

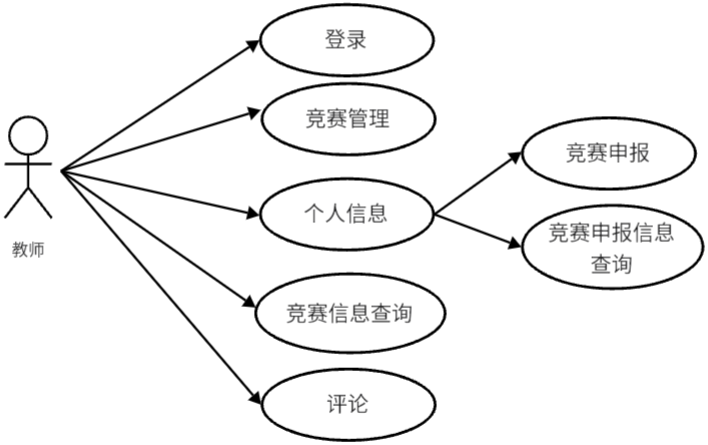


图2.4教师用例图

#### 2.3.1.3 学生需求分析

为了让学生能够更快捷地获取学科竞赛信息资源，提高学生竞赛积极性，系统提供了高效和美观的查询功能，这使用户端不用登录就可以查看各种学科竞赛信息。竞赛报名功能，当用户点击报名时，首先会检查该用户是否登录，如果登录就可以进行报名的后续操作，否则将会跳转到登录页面。此外，学生端还提供了个人中心，允许学生进行个人信息修改、已报名或参加的竞赛信息的查询和获奖情况查询。其中，学生的用例分析描述如图2.5所示。

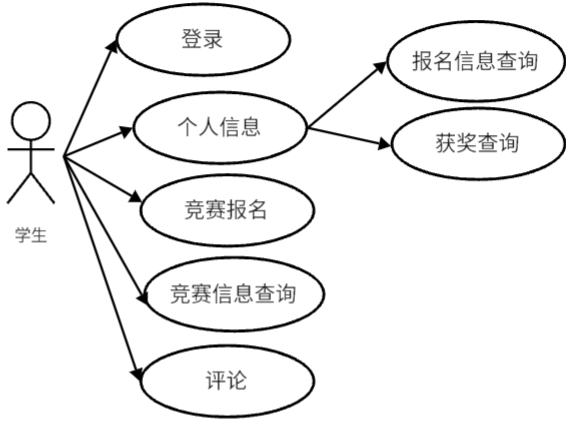


图2.5学生用例图

### 2.3.2 非功能性需求分析

#### 2.3.2.1 系统实用性

本系统通过提供高效的组织与管理、便捷的报名与查询以及实时互动与交流等功能，极大地提高了学科竞赛的实用性和效率。

#### 2.3.2.2 系统安全性

本系统采用了SpringSecurity框架，它提供了身份认证、授权、加密和防止会话固定等安全功能。还采用了Jwt技术，当用户访问需要认证的信息时，会进行token的验证。

#### 2.3.2.3 系统稳定性

本系统后端对各个接口都进行了异常处理，当功能出现问题时会在前端进行提示，以更好的跟进处理。前端是基于VUE的组件式开发，当一部分组件出现问题时，不会导致整个页面崩溃。

# 3 需求分析

## 3.1业务流程

登录模块是用户进入系统的入口，并且该系统一共设计了管理员、教师和学生三种用户角色，用户登录时输入账号和密码，账号和密码均正确时，根据其用户角色进行系统功能的动态加载，以保证用户对该系统功能的使用。用户名或密码不正确时，提示其错误信息，并重新登录。其业务流程图如3.1所示。

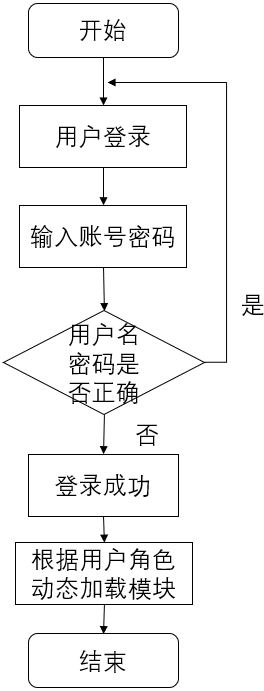


图3.1 登录流程图

学生竞赛报名模块是学生用户进行竞赛报名的入口。学生根据竞赛信息页查询竞赛，然后点击报名按钮进入竞赛报名页，首先填写报名信息，然后点击下一步进入确认报名信息页，最后学生确认报名信息点击确认报名，等待系统查询认证，若该用户或姓名已经报名过此次竞赛提示该用户或姓名已经报名，否则提示报名成功。其业务流程图如3.2所示。

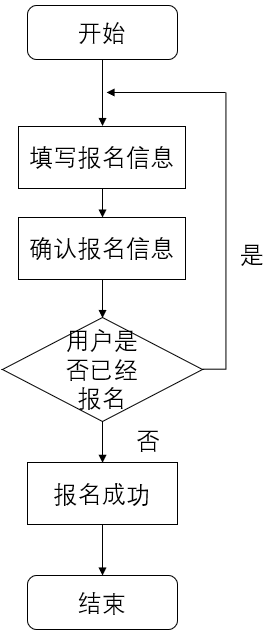


图3.2 报名流程图

## 3.2系统范围

在本系统中，合理地定义不同用户角色的操作范围对于保障系统的稳定性和数据的安全性至关重要，下面我将分别介绍三个端的使用范围。

学生使用范围：学生可以浏览所有已发布且对其开放的竞赛信息；学生能够报名参加感兴趣的竞赛，并提交个人报名信息；学生可以查看和管理自己已报名的竞赛状态，包括报名竞赛的结果和竞赛结果；学生还能够在竞赛讨论区发表意见，参与学术交流，但必须遵守论坛规则。

教师使用范围：教师不仅可以浏览开发的竞赛信息，还能够创建新的竞赛并提交给管理员审核。竞赛结束后教师可以在自己发布的竞赛中录入和更新参赛学生的成绩，发布竞赛结果。

管理员适用范围：管理员可以维护已创建的用户信息（包括教师和学生）。对教师提交的竞赛信息进行审核并提出意见，还可以对竞赛信息和轮播图进行管理。

## 3.3 数据流图

数据流图（Data Flow Diagram，简称DFD）是一种描述系统数据流动和处理的图形化表示方法[10]。接下来我将通过三层数据流图来展示系统内部的主要数据流动和处理过程。

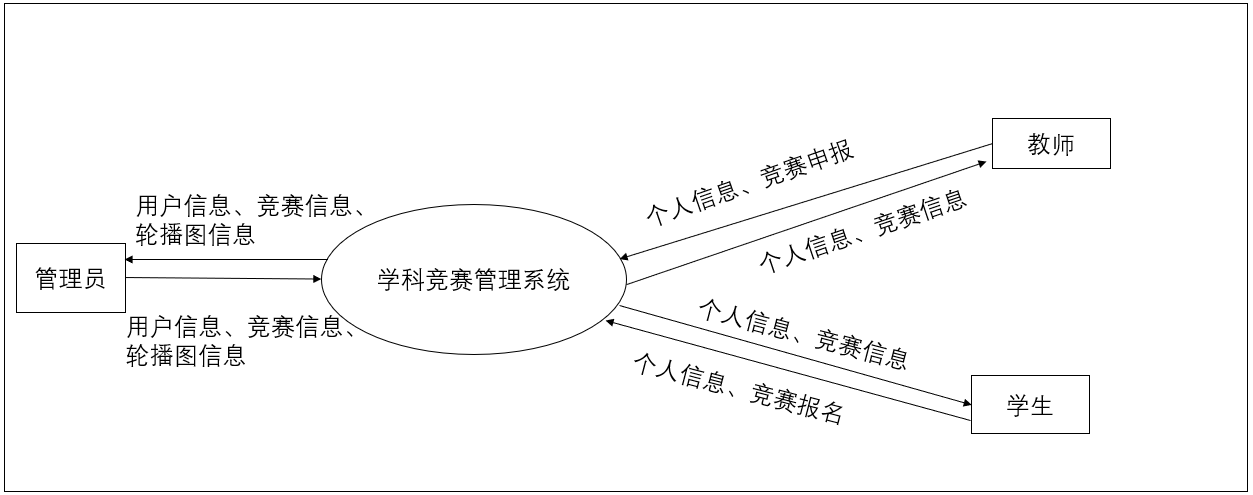
1. 顶层数据流图如图3.3所示。在顶层数据流图中，通常会有一个中心过程，代表系统的最核心功能，而外部实体通过数据流与这个中心过程交互。从下图可以看出管理员使用该系统时输入和输出的数据包括用户信息、竞赛信息和轮播图信息。当学生和教师用户使用该系统时，输入和输出的数据则是个人信息和竞赛信息。

图3.3 顶层数据流图

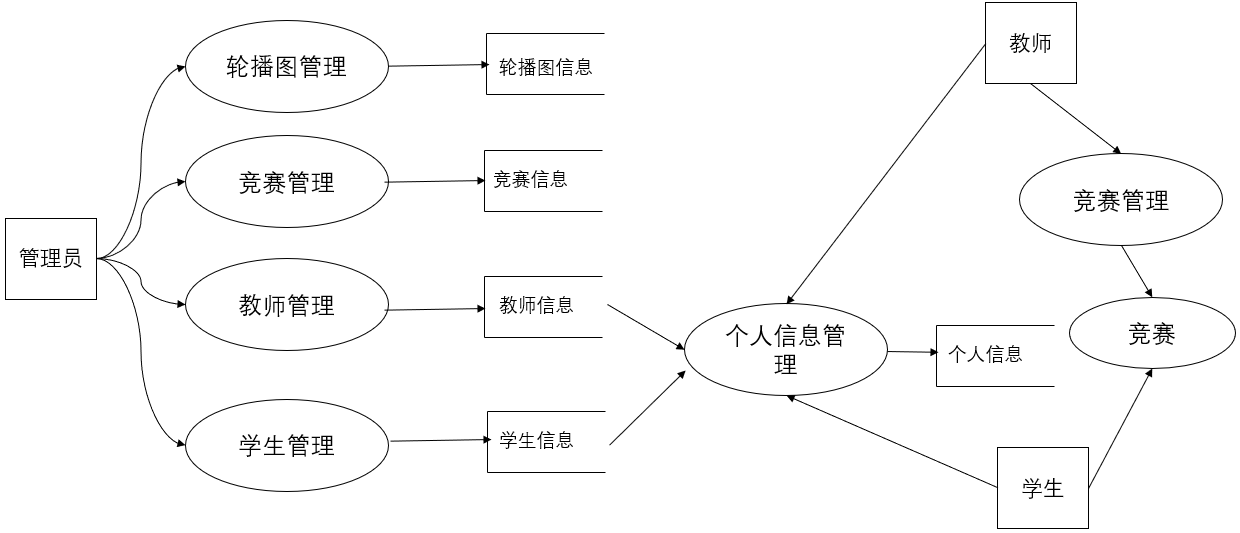
2. 第一层数据流图。即图3.4所示，在第一层数据流图中，我们将详细展示学科竞赛管理系统的核心业务流程。教师首先申报竞赛，然后管理员端就可以看到待审核的竞赛，等待管理员审核竞赛信息后，教师可以在个人信息的竞赛审核中可以查看竞赛审核结果，若审核通过后将会把竞赛发布到竞赛信息中，学生可以在竞赛页中查询竞赛并且报名。竞赛结束后教师对学生进行评分，学生可以在个人信息中查看竞赛成绩。

图3.4 第一层数据流图

3. 第二层数据流图。如图3.5所示，我们将深入细化管理员的管理模块的流程。该模块涵盖了用户的查询、添加、修改、删除以及竞赛信息的审核等操作。

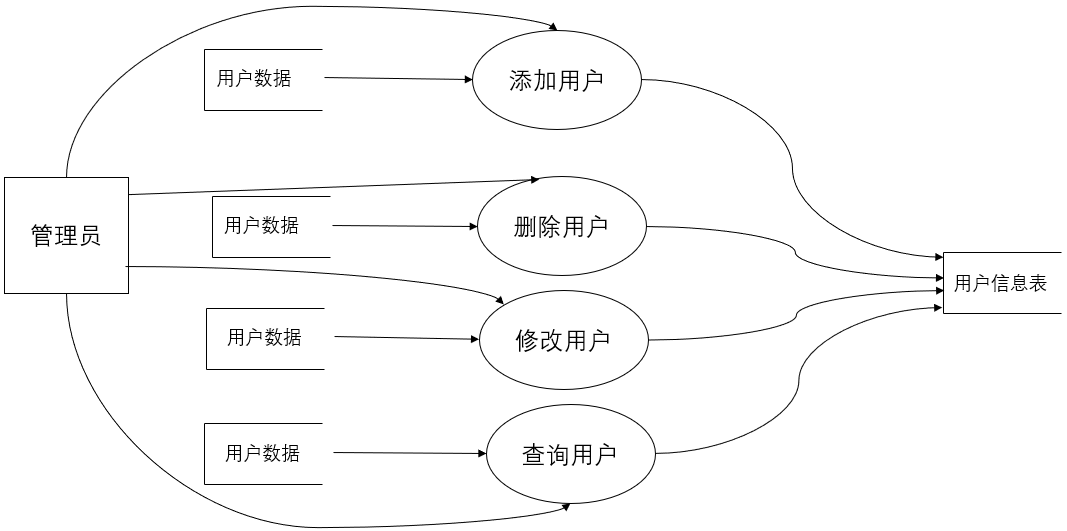


图3.5第二层数据流图

## 3.4 数据字典

（1）用户表:该表存放用户信息和用户登录时需要信息的的表，其主键为u\_id。password为用户密码字段，其通过用BCryptPasswordEncoder将密码转换为哈希值。

表3.6 用户表 db\_account

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 | 主键 | 默认值 |
| u\_id | int |  | 用户id | 主键 |  |
| tel | varchar | 20 | 电话 |  |  |
| email | varchar | 20 | 邮箱 |  |  |
| password | varchar | 255 | 密码 |  |  |
| role | varchar | 10 | 角色 |  |  |
| register\_time | datetime |  | 用户注册时间 |  |  |
| username | varchar | 10 | 用户名 |  |  |
| url | varchar | 20 | 图片路径 |  | 5fa46133f |

（2）竞赛申请表:该表存放教师申请竞赛信息的表，主键为application\_id。

表3.7 竞赛申请表 db\_applications

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 | 主键 | 默认值 |
| application\_id | int |  | 竞赛申请id | 主键 |  |
| teacher\_id | int |  | 申请教师 |  |  |
| competition\_id | int |  | 竞赛id |  |  |
| application\_time | date |  | 竞赛申请时间 |  |  |

（3）竞赛审核表:该表存放竞赛审核信息的表，主键为approval\_id。

表3.8 竞赛审核表 db\_approvals

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 | 主键 | 默认值 |
| approval\_id | int |  | 主键，竞赛审核id | 主键 |  |
| competition\_id | int |  | 竞赛id |  |  |
| approvver\_id | int |  | 审核人id |  |  |
| result | tinyint |  | 审核结果 |  |  |
| approval\_time | datetime |  | 审核时间 |  |  |
| suggest | varchar | 255 | 审核建议 |  |  |

（4）获奖表:该表存放各个竞赛的获奖信息，主键为award\_id。

表3.9获奖表 db\_awards

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 | 主键 | 默认值 |
| award\_id | int |  | 获奖记录id | 主键 |  |
| competition\_id | int |  | 竞赛id |  |  |
| user\_id | int |  | 获奖学生id |  |  |

（5）竞赛大类表:该表存放竞赛大类别信息，如大的分类(工科，文科等)，主键为id。

表3.10 竞赛大类表 db\_bigcategory

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 | 主键 | 默认值 |
| id | int |  | 竞赛大类别id | 主键 |  |
| family | varchar | 255 | 竞赛大类别 |  |  |

（6）竞赛小类表:该表存放竞赛小类别信息（程序设计、数学建模等），主键为category\_id。

表3.11 竞赛小类表 db\_category

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 | 主键 | 默认值 |
| category\_id | int |  | 竞赛小类别id | 主键 |  |
| subject | varchar | 255 | 竞赛小类别 |  |  |
| bigtype | int |  | 竞赛大类别id |  |  |

（7）评论表:该表存放用户的评论信息，主键为comment\_id。

表3.12 评论表 db\_comments

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 | 主键 | 默认值 |
| comment\_id | int |  | 评论id | 主键 |  |
| post\_id | int |  | 帖子id |  |  |
| user\_id | int |  | 评论用户id |  |  |
| parent\_comment\_id | int |  | 该评论的父评论id |  |  |
| content | varchar | 255 | 评论内容 |  |  |
| comment\_date | datetime |  | 评论时间 |  |  |

（8）帖子表:该表存放用户发的帖子信息，主键为post\_id。

表3.13 帖子表 db\_post

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 | 主键 | 默认值 |
| post\_id | int |  | 帖子id | 主键 |  |
| user\_id | int |  | 发帖人id |  |  |
| title | varchar | 255 | 帖子标题 |  |  |
| content | varchar | 255 | 帖子内容 |  |  |
| post\_date | datetime |  | 发帖时间 |  |  |

（9）竞赛信息表:该表存放竞赛信息的表，主键为competition\_id。

表3.14 竞赛信息表 db\_competition

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 | 主键 | 默认值 |
| competition\_id | int |  | 竞赛id | 主键 |  |
| competition\_name | varchar | 255 | 竞赛名称 |  |  |
| category\_id | int |  | 竞赛小类别 |  |  |
| organizer\_id | int |  | 组织者id |  |  |
| publish\_time | date |  | 发布竞赛间 |  |  |
| competition\_time | date |  | 竞赛时间 |  |  |
| registration\_time\_start | date |  | 竞赛报名始时间 |  |  |
| registration\_time\_end | date |  | 竞赛报名截止时间 |  |  |
| location | varchar | 255 | 竞赛地点 |  |  |
| status | tinyint |  | 竞赛审核状态 |  | 0 |
| is\_hot | tinyint |  | 竞赛是否热门 |  | 0 |
| award\_time | date |  | 竞赛获奖发布时间 |  |  |
| url | varchar | 255 | 竞赛图片地址 |  |  |
| description | text |  | 竞赛信息描述 |  |  |

（10）组织者表:该表存放主办方信息的表，主键为organizer\_id。

表3.15 帖子表 db\_organizer

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 | 主键 | 默认值 |
| organizer\_id | int |  | 组织者id | 主键 |  |
| sponsor | varchar | 255 | 主办方名称 |  |  |

（11）竞赛报名表:该表存放报名信息的表，主键为registration\_id。

表3.16 帖子表 db\_registrations

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 | 主键 | 默认值 |
| registration\_id | int |  | 报名id | 主键 |  |
| student\_id | int |  | 学生id |  |  |
| competition\_id | int |  | 竞赛id |  |  |
| registration\_time | datetime |  | 报名时间 |  |  |

# 4 总体设计

## 4.1 系统总体结构设计

本系统的开发采用的是前后端分离的技术。后端框架SpringBoot+SpringMVC+

MybatisPlus，前端框架是VUE+ElementPlus。系统总体架构设计如图4.1所示。



图4.1系统总体架构设计

## 4.2 系统的接口设计

### 4.2.1 用户接口

用户接口作为软件与用户之间的桥梁，通过提供多样化的交互方式，使得软件更加易于使用、功能更加丰富。一般而言，用户接口可以分为三种主要类型：程序接口、命令行接口和图形用户界面。本系统通过图形化界面给用户提供了用户登录接口，接口详情如下表4.2

表 4.2 登录接口详情

|  |  |
| --- | --- |
| 模块 | 接口描述 |
| 请求地址 | http://localhost:8888/zl/login |
| 请求方式 | POST |
| 请求参数类型 | application/x-www-form-urlencoded |
| 请求参数示例 | username=your\_username&password=your\_password |
| 返回参数类型 | Json格式的对象 |
| 返回参数示例 | {code:200,data:null,message:请求成功} |

### 4.2.2 外部接口

外部接口构成了软件和系统的核心要素，它们为第三方开发者和服务提供商提供了接入和整合的便捷通道。这些接口使得系统能够轻松扩展其功能，实现数据的流畅交换，同时推动整个系统生态系统的扩展与兴旺。在设计这些接口时，严格遵循业界标准和最佳实践至关重要，这不仅确保了接口的互操作性和易用性，也有助于吸引更广泛的开发者和服务提供商加入。

正如微信和支付宝等大型平台通过开放接口，为开发者提供了多样的功能，涵盖支付、社交和营销等多个领域。这些工具极大地促进了业务创新，使开发者能够利用这些平台的强大生态系统，创造出具有特色的应用和服务，捕捉商业机遇。

-

### 4.2.3 内部接口

内部接口是实现客户端与后端服务器之间数据传输的关键部分，当用户点击某个页面时，就会触发某个js方法以此来向后端某个内部接口发送http请求，当后端相应的内部接口接受到请求后就会调用数据库的数据并响应给前端。

1.获取用户信息接口如表4.3所示

表 4.3 用户信息接口详情

|  |  |
| --- | --- |
| 模块 | 接口描述 |
| 请求地址 | http://localhost:8888/personal/getUserInfo |
| 请求方式 | GET |
| 请求参数类型 | String |
| 请求参数示例 | String类型的token字符串 |
| 返回参数类型 | Json格式的对象 |
| 返回参数示例 | {code:200,data:自定义类型的对象,message:请求成功} |

2.修改用户信息接口如表4.4所示

表 4.4 修改用户信息接口详情

|  |  |
| --- | --- |
| 模块 | 接口描述 |
| 请求地址 | http://localhost:8888/personal/updateInfo |
| 请求方式 | POST |
| 请求参数类型 | Json格式的对象 |
| 请求参数示例 | {username:username，email:email,tel:tel,uid:uid} |
| 返回参数类型 | Json格式的对象 |
| 返回参数示例 | {code:200,data:自定义类型的对象,message:请求成功} |

3.删除轮播图的接口如表4.5所示

表 4.5删除轮播图接口详情

|  |  |
| --- | --- |
| 模块 | 接口描述 |
| 请求地址 | http://localhost:8888/lbt/delete |
| 请求方式 | GET |
| 请求参数类型 | String类型的字符串 |
| 请求参数示例 | name=“name” |
| 返回参数类型 | Json格式的对象 |
| 返回参数示例 | {code:200,data:null,message:请求成功} |

4.竞赛申请的接口如表4.6所示

表 4.6 竞赛申请接口详情

|  |  |
| --- | --- |
| 模块 | 接口描述 |
| 请求地址 | http://localhost:8888/competition/insertComp |
| 请求方式 | POST |
| 请求参数类型 | Json格式的对象 |
| 请求参数示例 | 自定义的竞赛类型的对象 |
| 返回参数类型 | Json格式的对象 |
| 返回参数示例 | {code:200,data:null,message:请求成功} |

## 4.3 软件体系结构

本系统采用B/S架构，即浏览器/服务器架构，它是一种基于Web技术的网络应用模型，其核心处理过程主要在服务器端进行。这种架构为网络应用提供了极高的灵活性和可扩展性，并且有价格低廉的优势，减少了客户端软件分发、安装和维护的成本和复杂性。用户只需通过浏览器与应用程序交互，浏览器负责展示界面和接收用户指令，这些优点使其成为我软件架构的选择。

# 5详细设计

## 5.1 功能模块设计

1.存储过程

通过存储过程的方式实现帖子的新增比动态生成的sql要快，因此定义了一个新增帖子的存储过程如图5.1。

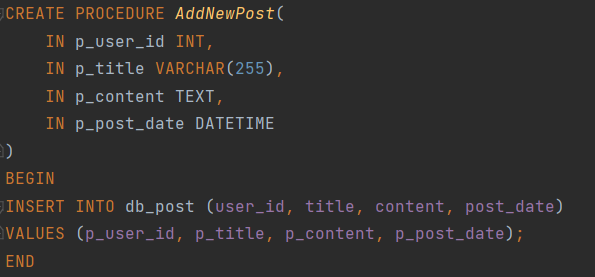


图 5.1 新增帖子存储过程

2.存储函数

统计竞赛报名人数在本系统中是非常重要的，因此可以通过定义一个带竞赛id参数的存储函数来实现这个功能。如图 5.2 所示。

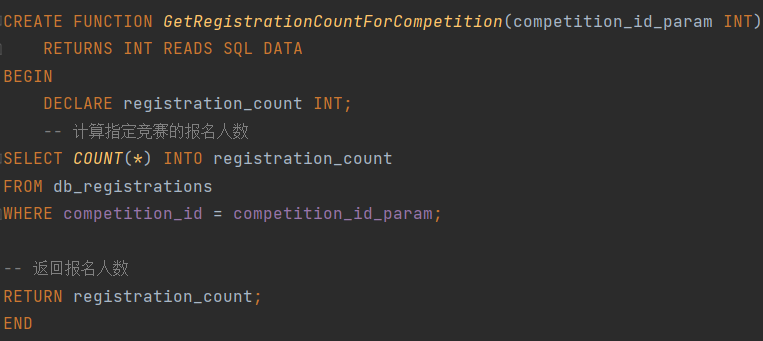


图 5.2 统计竞赛报名人数存储函数

3. 触发器

在本系统中，用户端可以查看到竞赛表中状态字段（status）为1的竞赛信息。当教师申请竞赛后会在竞赛表中立即创建一个状态字段为0的竞赛信息，同时会在审核表和竞赛申请表中创建对应的信息。在竞赛通过管理员审核后，审核表的结果字段（result）会置为1，刚审核通过的竞赛信息在竞赛表中的状态也需要置为1以便用户端及时查看竞赛信息。因此需要设置一个触发器，管理员审核通过后把审核表的结果字段置为1的同时把竞赛表对应竞赛的status字段置为1。触发器实现如图 5.3所示。



图 5.3 竞赛审核触发器

## 5.2 输入/输出格式设计

输入设计：输入设计的核心目标是确保系统能够接收到准确无误的数据[29]。在本系统的设计中，我们致力于确保输入机制不仅满足用户需求，还能在用户输入错误数据时提供清晰的提示，使用户能够迅速地进行更正。为了维护数据的完整性和安全性，系统对用户的输入行为设置了限制，一旦用户的输入错误次数超过设定的阈值，系统将自动暂时锁定用户的输入权限，直至一段预设时间过后才重新开放。此外，系统前端实施了表单验证机制，这不仅优化了用户体验，还有效减轻了后端服务器的验证负担。

输出设计：在本管理系统中，我们深知以一种清晰和直观的方式展现数据对于用户的重要性。因此，系统界面采用了简洁直观的设计理念，并通过集成的图表分析工具，对关键数据执行可视化处理，让用户能够直观地把握数据变化的趋势。系统的操作流程也经过精心设计，确保简单易懂。同时，在设计新系统的输入方式时，应充分利用现有的设备和资源，尽量避免大量数据通过键盘输入[30]。

## 5.3 人机对话设计

人机对话设计是一个全面的过程，涵盖用户界面构建、对话逻辑规划以及后端技术支持等关键要素。系统响应速度是决定用户体验的关键，因此快速且精确地响应用户请求至关重要。为了提高用户体验，本系统采用了Vue 3框架，其响应速度比Vue 2有显著提升。同时，系统还集成了Redis缓存服务，进一步提升了响应速度。系统采用了多种Element Plus组件以增强人机交互体验：

1.轮播图（Carousel）：用于展示图片的轮播，管理员可以通过新增和删除图片管理轮播图。

2. 时间选择器（TimePicker）：这是一个为用户提供直观界面以选择确切时间的组件。它支持小时、分钟和秒的精确选择，非常适合需要用户输入具体时间的场景。通过Element Plus的时间选择器，用户可以轻松地设置闹钟、预约时间或任何需要时间信息的功能。

3. 消息提示（ElMessage）与确认框（MessageBox）：ElMessage 是一个用于展示简短消息提示的函数，它是一个全局可用的方法，非常适合于快速显示操作反馈或状态更新。ElMessage 会在页面的顶部中间位置显示一个消息框，并且通常在几秒后自动消失。MessageBox则在用户执行关键操作前提供确认提示，增加操作安全性。

## 5.4 界面设计

### 5.4.1 用户界面设计

1.注册和登录是每个系统最基础的功能。注册功能给用户提供了一个使用该系统的对外接口，当用户注册后才能实现更多的权限操作。登录功能是该系统的主要入口，主要实现安全验证和身份确认。虽然用户不用登录页就可以查看竞赛的各种信息，但是只有在登录后才能进行报名，发言以及修改个人信息等功能。该功能要求用户在登录时输入用户名和密码，并向后端发送验证请求。后端系统采用Spring Security框架来安全地处理这些请求。为了增强安全性，所有密码在存储之前都会进行哈希处理，这一措施有效防止了在直接访问数据库时明文密码的泄露风险。登录成功后，系统将把用户信息存入redis中，以持久化和高效化的处理用户信息，系统还会生成一个用于后续操作时能够持续验证身份的token。该功能的实现如图5.0和图5.1所示。



图5.0 注册



图5.1 登录

2.首页这个模块将可以不用进行登录就可以查看竞赛的各种信息，这会给游客用户一个良好的体验。当用户查询或者看到自己感兴趣的竞赛信息时，可以点击图片或者竞赛名称进入竞赛详情页，竞赛详情页面将会提供更详细的竞赛信息和报名入口，但是用户登录后才能进行报名操作。简洁和一目了然的操作界面给用户提供了良好的视觉和操作反馈。该功能的实现如图5.2所示。



图5.2 首页

3.竞赛信息这个模块将会给用户更细致和详细的竞赛信息的查询，同样这个页面不需要登录就可以查看竞赛的各种详细信息。用户可以根据自己的专业或者感兴趣的竞赛项目进行分类查询，还给用户提供了更友好的报名时间和报名倒计时提示，使用户能更快的反应竞赛倒计时，然后用户立即登录后就可以进行报名操作。该功能的实现如图5.3所示。

图5.3 竞赛信息查询

4.该功能模块主要给用户提供个性化管理，用户可以通过该界面对自己的用户名、电话、邮箱、头像和密码进行修改。用户还可以在该界面对自己的报名信息和获奖情况进行查询。该系列功能实现如图5.4、5.5、5.6、5.7、5.8。

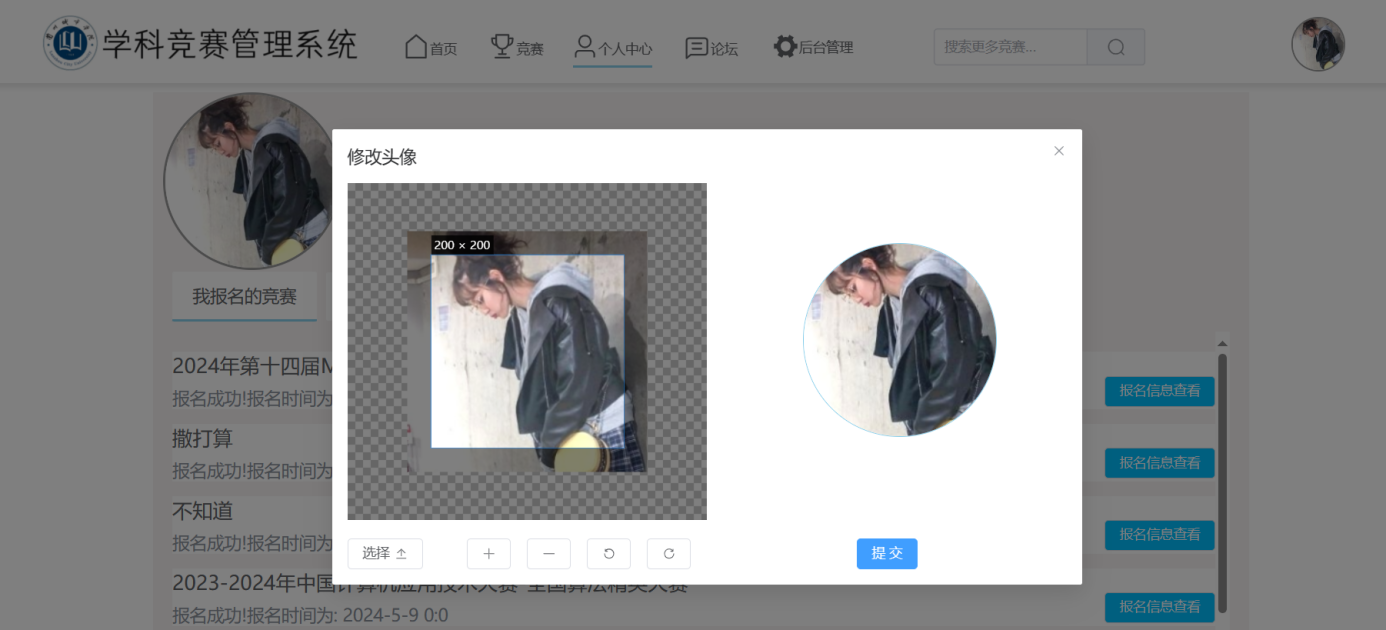


图5.4 个人信息管理



图5.5 个人竞赛信息查询



图5.6 个人密码修改

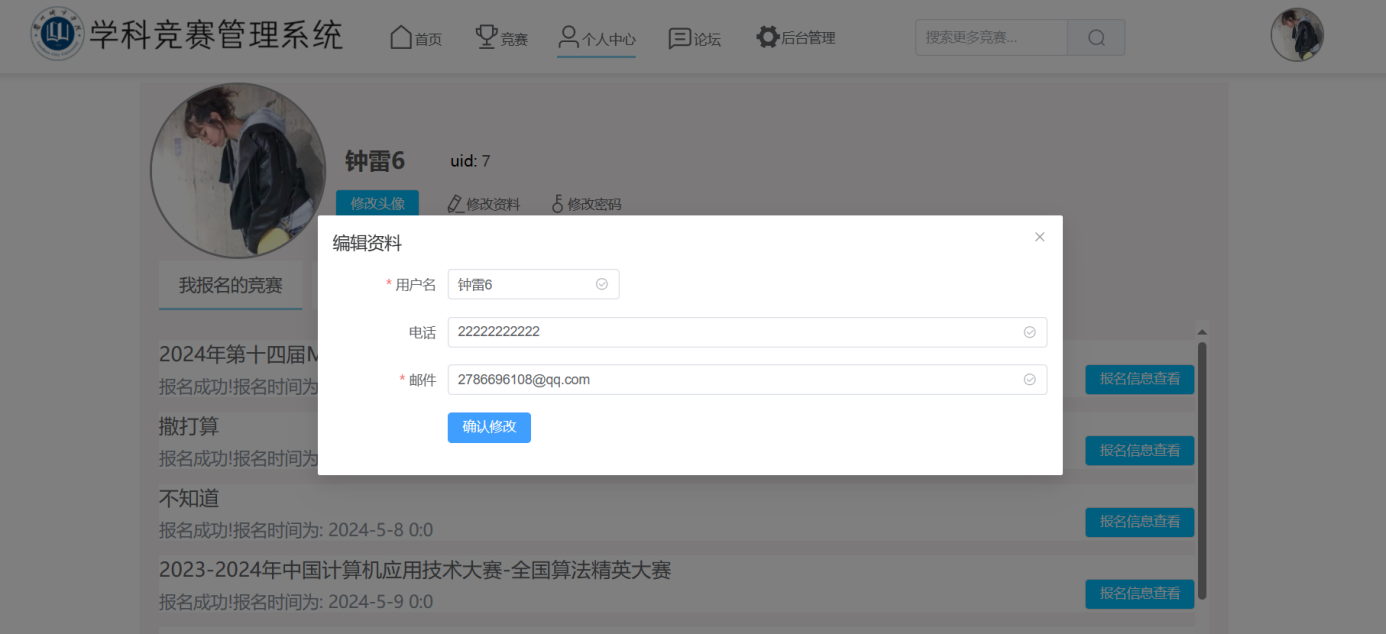
图5.7 个人信息修改



图5.8 用户报名竞赛信息页

5.该功能模块给用户提供了评论功能，用户可以通过该界面发帖，还可以查询其他人的发帖，还能回复他人消息。该功能不仅促进了用户之间的交流，还可以增加用户对竞赛的讨论，提升竞赛热度和更好的竞赛氛围。该功能实现如图5.9。



图5.9 论坛页

6.该模块给用户提供了竞赛报名功能，用户可以通过该界面进行竞赛报名。用户首先填写个人信息，然后点击下一步进行信息确认，最后实现报名成功。该功能实现如图5.10、5.11、5.12。



图5.10 填写个人信息页



图5.11 报名信息确认页



图5.12 报名成功页

7.该模块是给管理员提供的管理页。该模块用户员可以进行教师、学生和竞赛等信息实现管理。学生信息管理页可以对学生信息进行快速查询和修改，如图5.13、5.14、5.15。

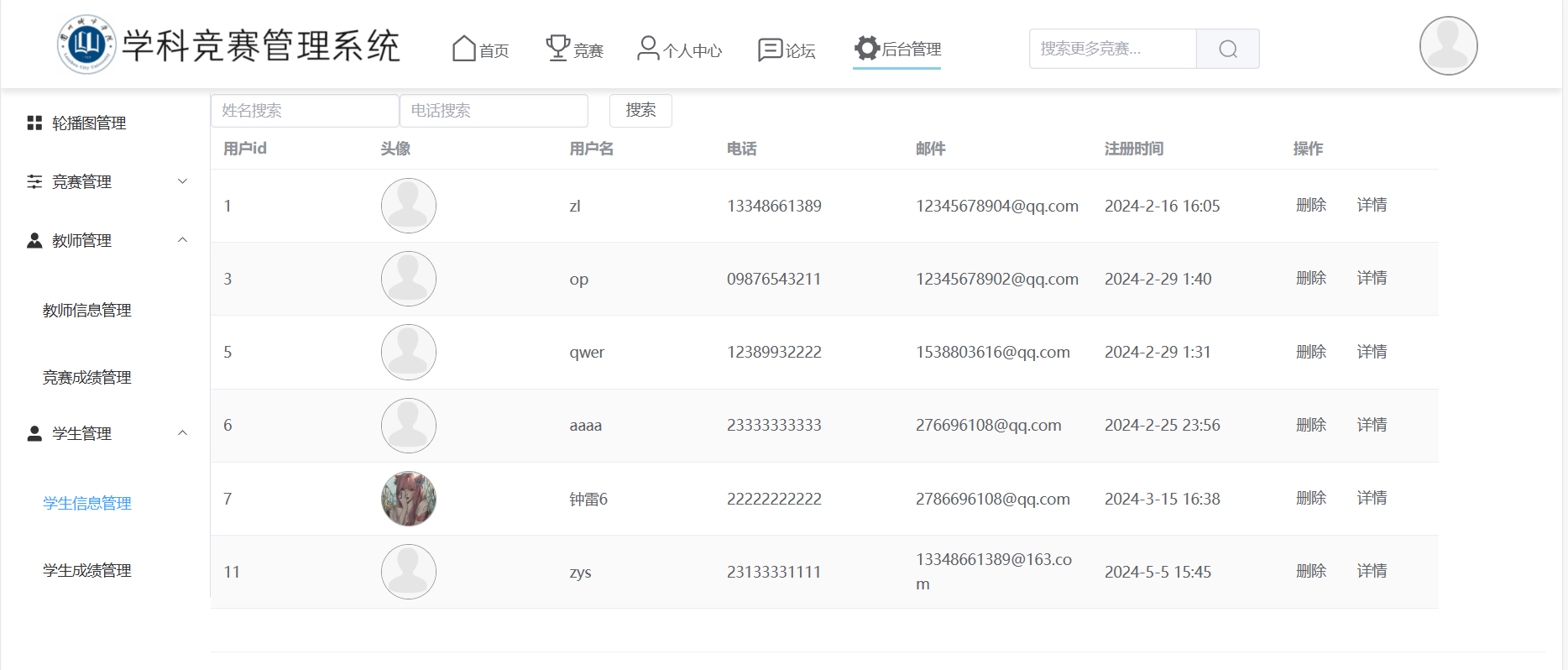


图5.13 学生信息管理页



图5.14 修改学生信息页

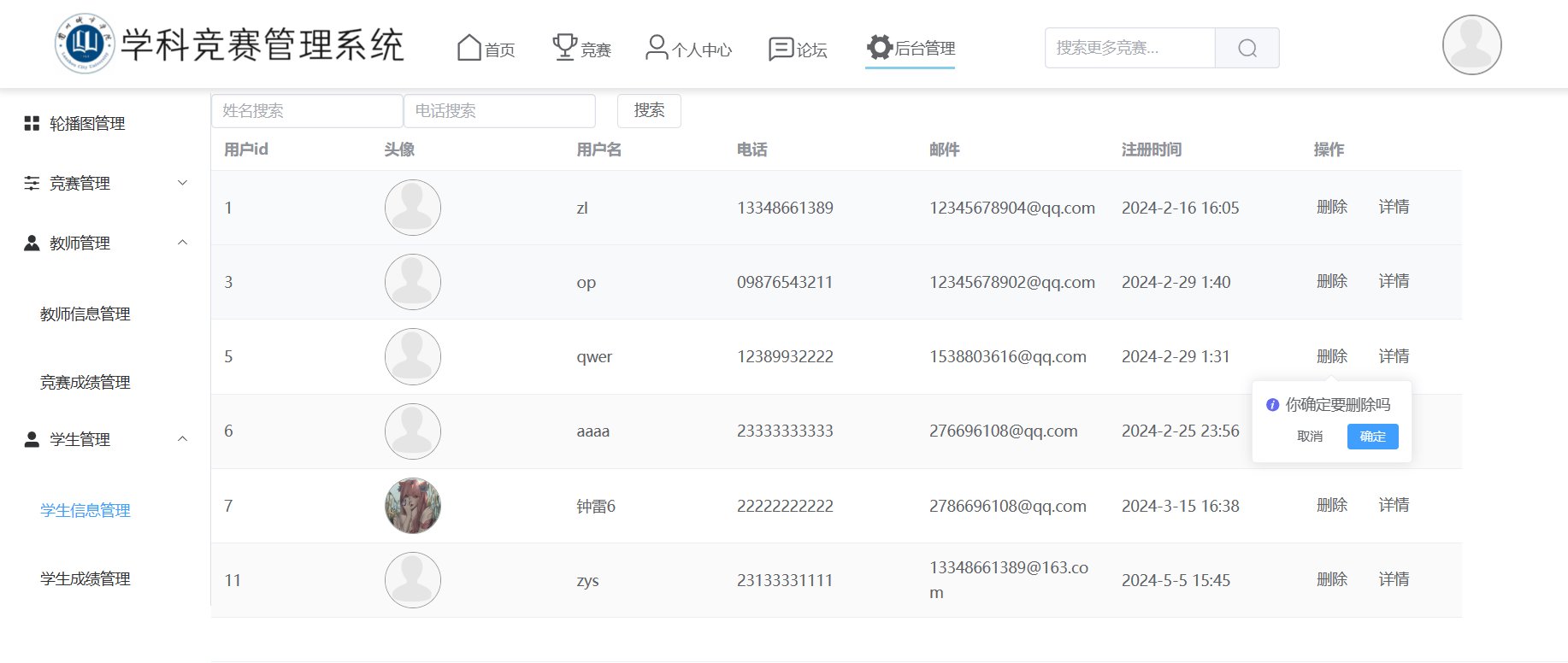


图5.15 删除学生信息页

### 5.4.2 菜单设计

本系统的菜单设计简单明了，采用了elementplus的menu菜单组件的设计。它的优点是具有清晰的选项分组功能，其选项分组中也提供了更多子选项给用户，不仅可以节省空间还能提高易观察性。菜单设计如图5.16所示。



图5.16 菜单设计

### 5.4.3 对话框设计

对话框界面设计是构建用户与系统之间交互对话窗口的精心过程，它是图形用户界面（GUI）的关键组成部分。这种设计的核心目的在于清晰展示交互信息，并为用户提供直观的操作选择，用户可以通过对话框来执行关键任务或做出重要决策。如图5.17对话框所示。



图5.17 对话框设计

### 5.4.4 窗口选项卡界面设计

在设计窗口选项卡界面时，目标是打造一个既直观又易于操作的用户界面，同时具备丰富的功能性。这样的界面设计使用户能够无缝地在多个工作区或不同任务之间进行切换，提高工作效率。如图5.18选项卡所示。



图5.18 窗口选项设计

# 5.5数据库设计

### 5.5.1 概念结构设计

数据库设计是系统建设的核心部分，它直接影响到数据的整合、性能和可扩展性。一个良好的数据库设计不仅能够确保数据的一致性和减少冗余，还能提高查询效率并且支持复杂的数据操作。本章将通过概念结构设计、逻辑结构设计和物理结构设计来对竞赛管理系统数据库设计过程、数据表结构设计、主外键联系和数据表一一进行解释。

(1) 用户实体的属性有用户id、电话、邮箱、密码、角色、用户创建时间、用户名、头像路径，其中id为用户实体的主键，用户E-R 图如图 5.19 所示。



图5.19 用户E-R图

(2) 评论实体的属性有评论id、帖子id、评论用户id、回复的用户id、父亲评论id、内容、发布时间，其中评论id为评论实体的主键，评论图如图 5.20 所示。

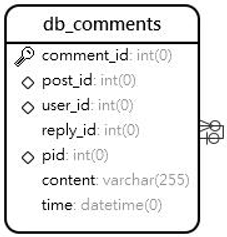


图5.20 评论E-R图

(3) 竞赛实体的属性有竞赛id、竞赛名称、竞赛类别、组织者、竞赛发布时间、比赛时间、发布状态、报名开始时间、报名结束时间、竞赛地址、竞赛描述、竞赛图片路径、是否热门、颁奖时间，其中竞赛id为竞赛实体的主键，竞赛E-R 图如图 5.21所示。

## 

图5.21 竞赛E-R图

(4) 成绩实体的属性有成绩id、分数、竞赛id、用户id，其中成绩id为成绩实体的主键，成绩E-R 图如图 5.22 所示。

## 

图5.22 成绩E-R图

(5) 学生成绩表的c\_id和u\_id分别与竞赛表的competition\_id和用户表的u\_id关联，学生成绩局部 E-R 图如图 5.23 所示。

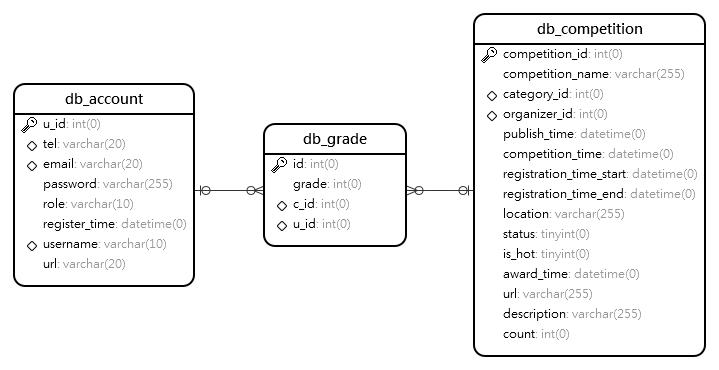


图5.23 学生成绩局部E-R图

(6) 竞赛报名表的u\_id和competition\_id分别与用户表的u\_id和竞赛表的competition\_id，竞赛报名局部E-R 图如图 5.24 所示。

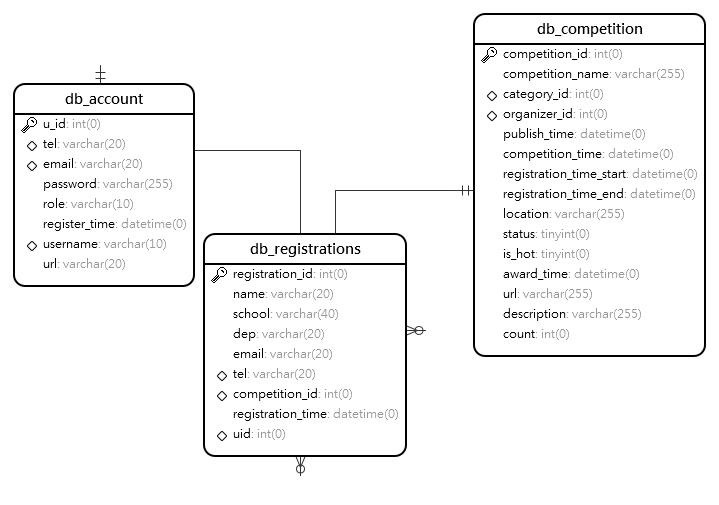


图5.24 竞赛报名局部E-R图

(7) 评论表的user\_id关联用户表的u\_id，pid关联自身表的comment\_id,评论局部E-R 图如图 5.25 所示。

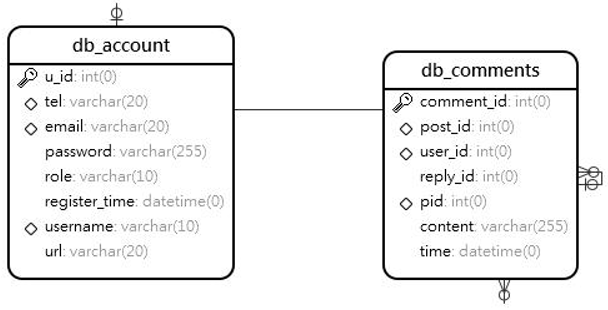


图5.25 评论局部E-R图

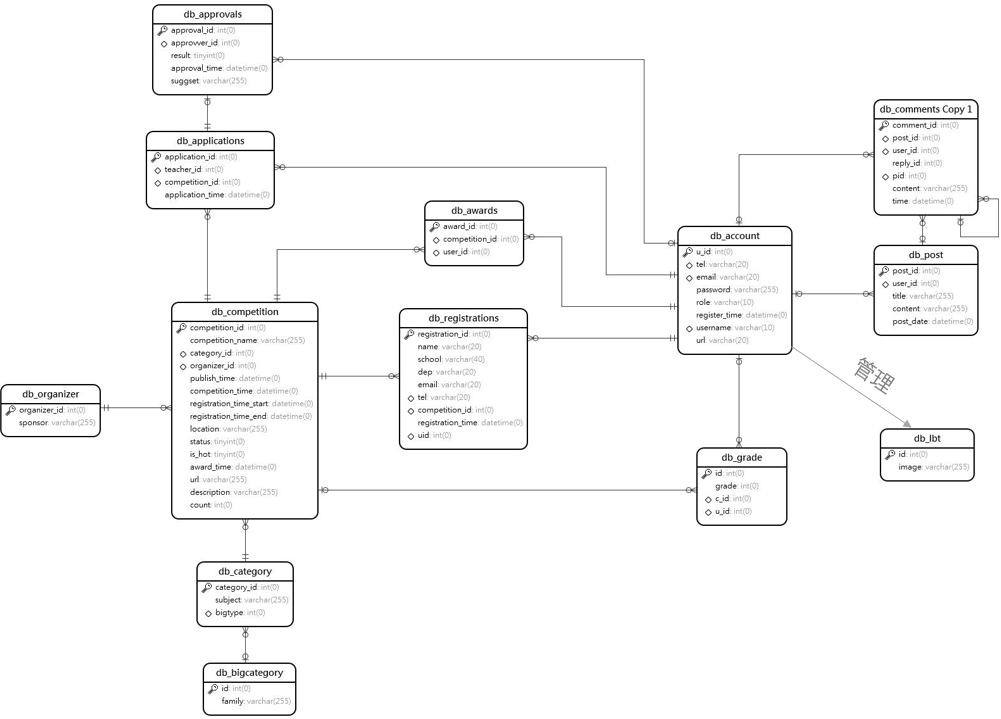


图5.26 全局E-R图

全局E-R模型（实体-关系模型）的设计是数据库概念设计的一个关键环节。在开展全局E-R模型设计的过程中，关键在于将各个局部E-R模型图进行细致的整合与优化，以形成连贯、一致的全局概念结构。这一整合过程确保了全局设计精准地映射出用户需求和业务流程的每一个细节，包括实体的界定、属性的分配、关系的构建以及约束的设定。全局E-R图如5.26所示。

### 5.5.2逻辑结构设计

### 上节我们提到了某些业务的实体，并进行了展示。然而这些实体之间可能具有某种联系，这时候我们就通过逻辑结构设计简化、优化甚至合并这些实体，并将实体转化为关系模式。比如管理员、教师、学生，我们就可以进行优化和合并，把他们优化为一个关系模式-用户。将上述的实体转化为关系模式如下：

1. 用户(用户id、电话、邮箱、密码、角色、注册时间、用户名、头像路径)。
2. 评论(评论id、帖子id、用户id、回复到的人的用户id、父亲评论id、评论内容、评论时间)。
3. 竞赛(竞赛id、竞赛名称、竞赛分类、组织者、发布竞赛时间、竞赛时间、竞赛报名开始时间、竞赛报名结束时间、竞赛地址、发布状态、是否热门、颁奖时间、竞赛图片路径、竞赛描述)。
4. 成绩(成绩id，成绩分数、竞赛id、用户id)。

### 5.5.3 物理结构设计

数据库模块通常用来存储用户信息、体测数据、体测报告等数据。通过数据库的管理系统对数据进行管理和存取，确保了系统数据的准确性和完整性[28]。数据库表存储了竞赛管理系统软件中的数据库数据，这是对学科竞赛数据进行数据库数据进行了可视化分析，以下就是对管理系统软件中存取数据库表的设计与制作[19]。

（1）用户表:该表存放用户信息和用户登录时需要信息的的表，其主键为u\_id。password为用户密码字段，其通过用BCryptPasswordEncoder将密码转换为哈希值，并将哈希值存储在数据库中。

表5.27 用户表 db\_account

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 |
| u\_id | int |  | 主键，用户id |
| tel | varchar | 20 | 电话 |
| email | varchar | 20 | 邮箱 |
| password | varchar | 255 | 密码 |
| role | varchar | 10 | 角色 |
| register\_time | datetime |  | 用户注册时间 |
| username | varchar | 10 | 用户名 |
| url | varchar | 20 | 图片路径默认值为3d7d45cc-1 |

（2）竞赛申请表:该表存放教师申请竞赛信息的表，主键为application\_id。由于申请表的teacher\_id和用户表的u\_id之间存在依赖关系，所以申请表teacher\_id为该表外键并依赖于用户表的u\_id。该表的competition\_id又与竞赛表的competition\_id之间存在依赖关系，所以competition\_id为该表的外键并依赖于竞赛表的competition\_id字段。

表5.28 竞赛申请表 db\_applications

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 |
| application\_id | int |  | 主键，竞赛申请id |
| teacher\_id | int |  | 申请教师 |
| competition\_id | int |  | 竞赛id |
| application\_time | date |  | 竞赛申请时间 |

（3）竞赛审核表:该表存放竞赛审核信息的表，主键为approval\_id。该表的competiti\_id字段与竞赛表的competiti\_id字段存在依赖关系，所以competiti\_id为该表的外键并依赖于竞赛表的competiti\_id。

表5.29 竞赛审核表 db\_approvals

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 |
| approval\_id | int |  | 主键，竞赛审核id |
| competition\_id | int |  | 竞赛id |
| approvver\_id | int |  | 审核人id |
| result | tinyint |  | 审核结果。1通过，0拒绝 |
| approval\_time | date |  | 审核时间 |
| suggest | varchar | 255 | 审核建议 |

（4）获奖表:该表存放各个竞赛的获奖信息，主键为award\_id。competition\_id和user\_id是该表的外键并分别依赖于竞赛表的competition\_id和用户表的u\_id。

表5.30 获奖表 db\_awards

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 |
| award\_id | int |  | 主键，获奖记录id |
| competition\_id | int |  | 竞赛id |
| user\_id | int |  | 获奖学生id |

（5）竞赛大类表:该表存放竞赛大类别信息，如大的分类(工科，文科等)，主键为id。

表5.31 竞赛大类表 db\_bigcategory

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 |
| id | int |  | 主键，竞赛大类别id |
| family | varchar | 255 | 竞赛大类别 |

（6）竞赛小类表:该表存放竞赛小类别信息（程序设计、数学建模等），主键为category\_id。该表的外键为bigtype并依赖于 db\_bigcategory表的id字段。

表5.32 竞赛小类表 db\_category

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 |
| category\_id | int |  | 主键，竞赛小类id |
| subject | varchar | 255 | 竞赛小类别 |
| bigtype | int |  | 竞赛大类别id |

（7）评论表:该表存放用户的评论信息，主键为comment\_id。该表的外键为post\_id、user\_id和parent\_comment\_id，分别依赖于帖子表的post\_id、用户表的u\_id和本表的comment\_id字段。

表5.33 评论表 db\_comments

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 |
| comment\_id | int |  | 主键，评论id |
| post\_id | int |  | 帖子id |
| user\_id | int |  | 评论用户id |
| parent\_comment\_id | int |  | 该评论的父评论id |
| content | varchar | 255 | 评论内容 |
| comment\_date | date |  | 评论时间 |

（8）帖子表:该表存放用户发的帖子信息，主键为post\_id。该表的user\_id为该表的外键并依赖于用户表的u\_id字段。

表5.34 帖子表 db\_post

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 |
| post\_id | int |  | 主键，评论id |
| user\_id | int |  | 发帖人id |
| title | varchar | 255 | 帖子标题 |
| content | int |  | 帖子内容 |
| post\_date | date |  | 发帖时间 |

（9）竞赛信息表:该表存放竞赛信息的表，主键为competition\_id。该表的外键为category\_id，其依赖于db\_ category的category\_id字段。

表5.35 竞赛信息表 db\_competition

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 |
| competition\_id | int |  | 主键，竞赛id |
| competition\_name | varchar | 255 | 竞赛名称 |
| category\_id | int |  | 竞赛小类别 |
| organizer\_id | int |  | 组织者id |
| publish\_time | date |  | 发布竞赛时间 |
| competition\_time | date |  | 竞赛时间 |
| registration\_time\_start | date |  | 竞赛报名开始时间 |
| registration\_time\_end | date |  | 竞赛报名截止时间 |
| location | varchar | 255 | 竞赛地点 |
| status | tinyint |  | 竞赛审核状态，0审核中,1通过审核 |
| is\_hot | tinyint |  | 竞赛是否热门，0不热门，1热门 |
| award\_time | date |  | 竞赛获奖发布时间 |
| url | varchar | 255 | 竞赛图片地址 |
| description | text |  | 竞赛信息描述 |

（10）组织者表:该表存放主办方信息的表，主键为organizer\_id。

表5.36 帖子表 db\_organizer

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 |
| organizer\_id | int |  | 主键，组织者id |
| sponsor | varchar | 255 | 主办方名称 |

（11）竞赛报名表:该表存放报名信息的表，主键为registration\_id。该表的student\_id 、competition\_id是外键，分别依赖于用户表的u\_id和竞赛表的competition\_id。

表5.37 帖子表 db\_registrations

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 字段长度 | 字段说明 |
| registration\_id | int |  | 主键，报名id |
| student\_id | int |  | 学生id |
| competition\_id | int |  | 比赛id |
| registration\_time | date |  | 报名时间 |

# 6系统实现

## 6.1实现工具

1.集成开发环境（IDE）：

在Java编程领域，IntelliJ IDEA以其卓越的性能和丰富的功能备受赞誉。这款IDE为Java开发者提供了强大的智能代码助手、精准的代码自动提示、灵活的重构工具、全面的JavaEE支持、便捷的版本控制集成以及深入的代码分析等功能。

2.数据库管理工具：

MySQL Workbench是一款功能全面的数据库管理工具，它集成了数据库设计、查询、管理以及优化等多种功能。通过这款工具，开发者可以轻松地对数据库进行各种操作和管理，从而提高开发效率和数据库管理的便捷性。所以，在设计中选择使用MySQL作为数据库的基本材料[25]。

3.SpringBoot相关工具：

Spring Initializr是一个方便快捷的在线工具，它可以帮助开发者快速生成基于Spring Boot的项目基础结构。通过使用Spring Initializr，开发者可以节省大量配置和初始化项目的时间，更加专注于业务逻辑的开发。

4. 测试工具：

JUnit：JUnit是一款广泛使用的Java单元测试框架，它提供了编写和执行单元测试的能力，帮助开发者确保代码的正确性和可靠性。

## 6.2技术框架主要功能模块流程和编码

本系统的功能实现首先聚焦于最基本的用户登录，其涉及了前端的人机交互、后端服务验证以及数据库数据查询等多个环节。用户通过访问WEB服务器提供的服务界面，输入账号和密码发起登录请求。这一过程借助Axios发送HTTP请求至后端SpringBoot服务器，首先SpringSecurity先会进行拦截，然后对前端发来的账号和密码到数据库中查询和对比，以核实数据的真实性。若验证未通过，系统将反馈用户名或密码错误提示用户，以便用户重新操作直至登录成功。一旦数据在数据库中存在且验证通过，系统将会给用户颁布一个token，以用来后续用户访问私密信息时进行鉴权认证。登录成功后前端页面将自动跳转至首页。在整个登录流程中，控制层（Controller）扮演着关键角色。一旦接收到前端发送的正确登录信息，Controller会调用服务层（Service）中预定义的登录处理方法。服务层会进一步调用数据访问层（Mapper）中定义的方法，Mapper层会与数据库进行交互，执行具体的查询操作。完成数据库查询后，服务层会将查询结果返回给控制层，控制层再将结果反馈至前端界面，供用户查看。图6.1是业务逻辑图。

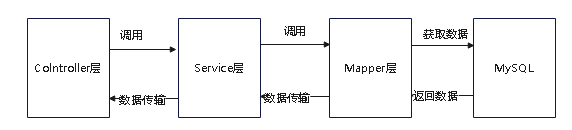


图6.1业务逻辑图

1.用户登录模块设计。由于使用的是SpringSecurity框架，controller层将不用进行方法处理，而是在service层中对方法loadUserByUsername进行重写就可以对用户登录进行判断了。图6.2是service层重写的方法。

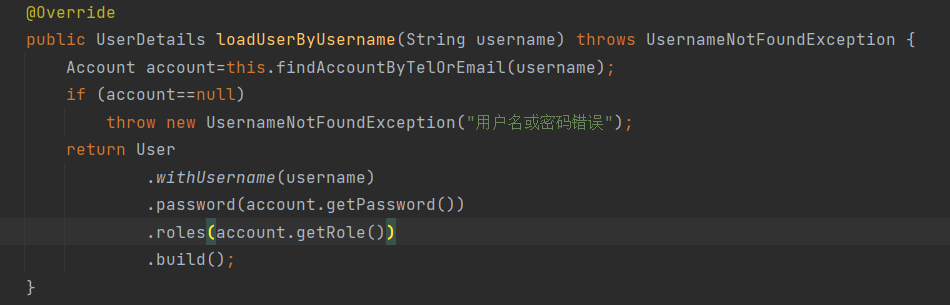


图6.2用户登录service层图

2.用户注册模块设计。首先用户对注册信息进行填写，然后发送请求到后端，后端首先会判断用户名是否已注册，如果是就会在前端返回提示，否则继续下一步。下一步，由于注册是会对注册的用户邮箱进行验证码发送，所以会判断该账号是否已发送验证码，如果没有就提示用户请发送验证码，否则继续下一步对用户填写的验证码和发送的验证码就行比对，通过就提示注册成功并跳转到登录页面，否则提示验证码错误，请用户重新输入验证码。实现代码，如图6.3、图6.4、图6.5所示。

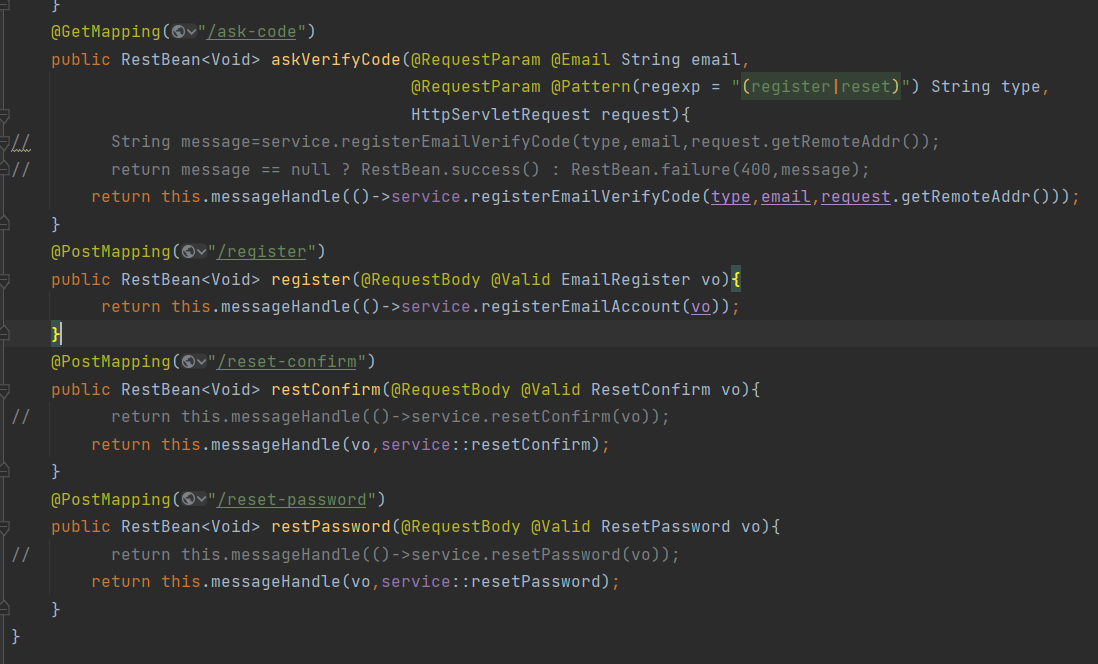


图6.3用户注册controller层图



图6.4用户注册service层图

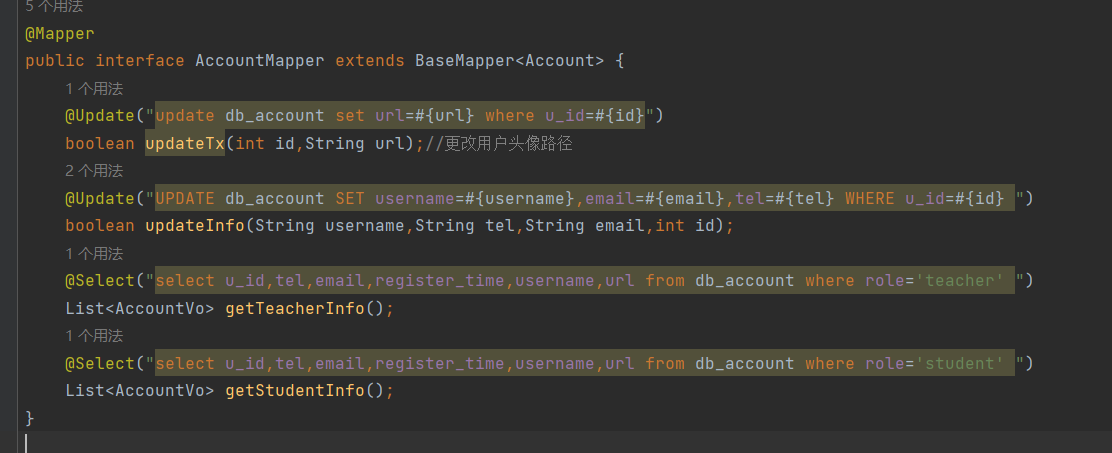


图6.5用户注册mapper层图

# 7系统的测试与维护

## 7.1 测试用例与测试结果

本章将通过测试选择、测试环境、测试过程来完成对学科竞赛管理系统的测试，最终得出一份测试结论，根据测试结论对系统界面、系统功能进行改善，以保证系统功能完善的同时提高系统处理数据的效率。

1、功能测试：本测试旨在核实系统是否精确无误地实现了所有预设功能，包括用户注册、登录流程、竞赛信息查询、竞赛申请流程等。经过严格的测试流程，系统展现了高度的准确性和可靠性，所有测试用例均得到了符合预期的结果。

（1）用户登录模块测试

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试项 | 验证流程 | 预期结果 | 实际结果 | 结论 |
| 用户登录 | 输入账号:test  密码:123456 | 登录成功，跳转到首页 | 登录成功，跳转到首页 | 测试通过 |
| 用户登录 | 输入账号:test  密码:12345 | 登录失败，提示用户名或密码错误 | 登录失败，提示用户名或密码错误 | 测试通过 |

（2）用户注册模块测试

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试项 | 验证流程 | 预期结果 | 实际结果 | 结论 |
| 用户注册 | 第一步，填写注册角色、用户名、密码、电子邮件，然后点击获取验证码  第二步，查看对应邮箱获取的验证码进行填写，然后点击立即注册 | 注册成功，跳转到登录页面 | 注册成功，跳转到登录页面 | 测试通过 |
| 用户注册 | 填写注册角色、用户名、密码和注册的电子邮件，然后点击获取验证码 | 提示此邮箱已被注册 | 提示此邮箱已被注册 | 测试通过 |

（3）竞赛信息查询模块测试

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试项 | 验证流程 | 预期结果 | 实际结果 | 结论 |
| 竞赛信息查询 | 点击竞赛按钮 | 显示全部竞赛信息 | 显示全部竞赛信息 | 测试通过 |
| 竞赛信息查询 | 点击工科按钮 | 显示全部工科竞赛信息 | 显示全部工科竞赛信息 | 测试通过 |
| 竞赛信息查询 | 点击力学按钮 | 显示全部力学竞赛信息 | 显示全部力学竞赛信息 | 测试通过 |

（4）报名模块测试

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 测试项 | 验证流程 | 预期结果 | 实际结果 |  | 结论 |
| 竞赛报名 | 先登录，再报名竞赛 | 直接跳转到报名页面 | 直接跳转到报名页面 |  | 测试通过 |
| 竞赛报名 | 未登录，再报名竞赛 | 跳转到登录页面，提示未登录，请登录后操作 | 跳转到登录页面，提示未登录，请登录后操作 |  | 测试通过 |

（5）个人信息修改模块测试

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试项 | 验证流程 | 预期结果 | 实际结果 | 结论 |
| 个人信息修改 | 第一步，点击修改信息按钮，弹出个人信息对话框  第二步，修改个人信息 | 提示修改成功 | 提示修改成功 | 测试通过 |
| 个人信息修改 | 第一步，点击修改信息按钮，弹出个人信息对话框  第二步，修改重复的用户名 | 提示该用户名已被注册 | 提示该用户名已被注册 | 测试通过 |

2、性能测试：为了模拟真实环境下的高并发场景，我们采用了性能测试工具（如JMeter），对系统的响应时间、处理能力及并发用户数等关键性能参数进行了全面测试。测试结果显示，系统在不同负载条件下均能达到预设的性能标准，不仅能够在用户可接受的时间内快速响应，而且能够高效处理大量并发请求，确保了系统在高负载状态下的稳定运行。

（1）高负载下的性能测试

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试项 | 验证流程 | 预期结果 | 实际结果 | 结论 |
| 首页最新竞赛信息查询 | 1秒钟逐渐启动10个线程，每个线程发送100次请求，共1000次请求 | 请求成功 | 请求成功 | 测试通过 |
| 竞赛页所有竞赛信息查询 | 1秒钟逐渐启动10个线程，每个线程发送100次请求，共1000次请求 | 请求成功 | 请求成功 | 测试通过 |

## 7.2 系统维护

系统维护是确保学科竞赛管理系统稳定运行、功能完善和数据安全的关键环节。它涵盖了故障排查、数据维护以及性能调优与提升等多个方面，旨在通过及时发现并解决潜在问题，保持系统的最佳状态，为竞赛组织者、参赛者和评审人员提供高效、可靠的服务支持。

故障排查：当系统遭遇故障或异常时，首要任务是迅速定位问题根源，并采取有效措施进行修复，以确保系统能够尽快恢复正常的运作状态，最大限度地减少对用户的影响[26]。

数据维护：数据维护一直以来都是系统的重要任务，因此定期执行系统数据的备份工作，以应对可能的数据丢失或损坏风险。在面临任何故障时，这些策略将确保我们能够迅速地将数据库恢复至一个稳定且一致的状态，从而最小化对数据库的潜在损害。

性能调优与提升：基于系统的测试结果以及用户的实际反馈，我们会对系统进行细致的性能分析，并根据分析结果进行相应的优化调整，以提升系统的响应速度和稳定性，为用户带来更加流畅、高效的使用体验。

# 结论

系统基于Java、SpringBoot和MySQL技术开发，有效地满足了学生、教师和管理员不同用户角色的需求，学科竞赛管理系统已经小有成效，但仍存在一些可以改进的空间。

未来工作可以在几个方向上进一步发展和完善。首先，随着用户基数的增长和数据量的扩大，系统的稳定性和科扩展性将成为关键；进一步优化系统的智能化程度，进一步减少教师和管理员的工作量；增加数据分析功能，为竞赛组织者提供更加深入的数据支持。此外，随着技术的不断发展和竞赛的日益复杂化，系统也需要不断更新和完善，以适应新的需求和挑战。

# 参考文献

1. 彭杨丽 , 徐家喜 , 陈培培 . 基于 Java EE 学科竞赛管理系统 的设计与实现 [J]. 电脑知识与技术 ,2018,14(19):93-95.
2. 国家中长期教育改革和发展规划纲要工作小组办公室.国家中长期教育改革和发展规划纲要(2010－2020年).http://www.moe.gov.cn/srcsite/A01/s7048/201007/t20100729\_171904.html,2010,07,29.
3. 王亚男.以学科竞赛助力创新型人才培养.世界教育信息,2020,33(05):62-65,71.
4. 陈伟,方霞,吴俊.基于数字化校园的学生学科竞赛管理系统.实验室研究与探索,2015,34(10):201-205.
5. 罗玲玲.高校学科竞赛信息管理系统的研究与分析[硕士学位论文].云南大学,昆明,2015.
6. 张富晓,韩传军,伍建川.学科竞赛背景下校院两级创新创业教育模式探索.教育教学论坛,2020(49):125-126.
7. 胡沁涵，王亚男，杨季文，等 . 高校学科竞赛组织管理平台的研究与实现［J］．福建电脑，2021，37( 09) : 75 － 78.
8. 方伟鉴，曾俊成，唐银翔，等 . 基于 B/S 的高校大学生学科竞赛大数据管理系统设计与实现［J］． 信息通信，2020( 06) : 164 － 165.
9. 胡金宇.基于Spring Boot和Vue框架的企业绩效考核系统设计与实现[D].湖北师范大学,2023.
10. 陈国才,基于ASP.NET的教学综合管理系统的设计与实现 ,大连理工大学硕士论文,2012.
11. 黄家常,王准.绩效考核管理系统设计与实现[J].福建电脑,2024,40(04):88-92.
12. 徐莲花,胡杨,赵鑫,等.基于BPMN的军工集团企业经营业绩考核管理系统的设计与实现[J].航天工业管理,2023,(04):41-44.
13. 胡金宇.基于Spring Boot和Vue框架的企业绩效考核系统设计与实现[D].湖北师范大学,2023.
14. 郭建卫.基于SpringBoot+Mybatis+Vue的学生德育量化考核系统设计与实现[J].现代信息科技,2023,7(01):18-22.
15. 王晓勇,俞松坤.以学科竞赛引领创新人才培养[J].中国大学教学,2007(12):59-60.
16. 董方旭,况晓慢.高校学科竞赛运行体系的构建[J].中国成人教育,2010(14):32-33.
17. 陆国栋,陈临强,何钦铭,等.高校学科竞赛评估:思路、方法和探索[J].中国高教研究,2018(2):63-68,74.
18. 万常选,廖国琼,吴京慧,等.“数据库系统原理”课程教材建设与教学改革——培养“会”用数据库的计算机应用人才[J].计算机教育,2009(24):153-157.
19. 朱莹芳.基于PHP和MySQL的网上生鲜商城的开发[J].江苏工程职业技术学院学报,2020,20(4):11-14.
20. KAN J．Design and implementation of teaching quality evaluation systembased on SpringBoot［C］．成都: 第七届计算与信息科学国际学术会议，2019．
21. 毛洪贲，王石发，姜 琪，等． 基于． NET 的学科竞赛管理系统设计与实现［J］． 现代教育技术，2013( 12) : 107-110.
22. 张清祥．搭建学科竞赛平台提高学科竞赛水平［J］．南阳师范学院学报(社会科学版) ，2011(9) : 101-111．
23. 李淑芝,兰红,杨书新,等.以学科竞赛为依托强化信息类学生创新能力的探索与实践[J].江西理工大学学报,2013,34(2):70-72.
24. 唐国瑞,秦洪敏.“互联网+”向体育教育的延伸——智能穿戴设备在学校体育中的应用展望[J].体育科技,2016,37(03).
25. 李夏梦.基于作业成本法的A酒店成本管理研究[J].三峡大学,2019.
26. 李冰.中国大学生体质健康监测管理体制与运行机制研究[J].贺州学院学报,2018,34(04).
27. 崔为花,崔为刚.SQL Server2000数据完整性攻略[J].计算机与信息技术,2007,31.
28. 纪方军,张辉,李开方,袁振山.自移式胶带切割修整机的研制与应用[J].科学试验,2009,3.
29. 盛金锋.基于WEB的高校新生报到管理平台设计[J].湖北工业大学,2018.
30. 基于CAN通信技术的火电厂电气监控系统设计与研究[J].2022.

# 致谢

在本次程序设计和论文撰写的过程中，我面临了众多挑战、障碍与困惑。幸运的是，我的指导老师始终以耐心和细致的态度解答我的疑问，为我指引了明确的方向。这让我在完成设计和论文的征途上能够坚定地迈进，逐一克服重重难关。因此，我要衷心感谢我的指导老师郭茗涵老师，在论文撰写过程中给予我的精心指导和慷慨帮助！

同时，我也要向我的同学们和朋友们表达我的感激之情。在论文的编写过程中，我们相互学习、相互激励、相互扶持。正是得益于你们的耐心指导和无私援助，我才能解决众多细节问题，确保我的毕业设计工作得以顺利进行。感谢你们给予我的关心和支持。

我还要向我的家人表达深深的谢意。他们永远是我最坚强的后盾，无论我遇到何种挑战，他们总是毫无保留地给予我支持和鼓励。

最后，我要感谢我的学校，感谢它为我提供了优越的学习环境和丰富的学术资源。在这里，我不仅掌握了专业知识，还学会了如何与人沟通、如何应对挑战。这些宝贵的经历将成为我未来人生旅途中的财富。带着这份感激，我将继续前进，不断追求更高的目标。