西南交通大学

本科毕业设计（论文）

体质健康数据管理系统设计与实现

年 级： 2018级

学 号： 2018112674

姓 名： 黄纯峰

专 业： 计算机科学与技术

指导教师： 钟灿

二零二二年五月

西南交通大学

本科毕业设计（论文）学术诚信声明

本人郑重声明：所呈交的毕业设计（论文），是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的作品成果。对本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。本人完全意识到本声明的法律结果由本人承担。

作者签名：

日期： 年 月 日

西南交通大学

本科毕业设计（论文）版权使用授权书

本毕业设计（论文）作者同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权西南交通大学可以将本毕业设计（论文）的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本毕业设计（论文）。

**保密**□，在年解密后适用本授权书。

本论文属于

**不保密**☑。

（请在以上方框内打“🗸”）

作者签名： 指导教师签名：

日期： 年 月 日 日期： 年 月 日

院系 计算机与人工智能学院 专 业

年级 2018级 姓 名

题目

指导教师

评 语

指导教师 (签章)

评 阅 人

评 语

评 阅 人 (签章)

成 绩

答辩委员会主任 (签章)

年 月 日

**毕业设计（论文）任务书**

班 级 计算机2018-02班 学生姓名 黄纯峰 学 号 2018112674

发题日期：2021年12月10日 完成日期：2022年5月28日

题 目

1、本论文的目的、意义

2、学生应完成的任务

3、本论文与本专业的培养目标达成度如何？（如在知识结构、能力结构、素质结构等方面有哪些有效的训练。）

此为计算机科学与技术专业的最新毕业要求，其他专业应替换为各自专业的毕业要求！

本论文支撑本专业以下毕业要求的达成：**（1）**能够通过查阅和分析文献，为计算机系统及工程的问题求解寻找方案，并认识到所求解的问题具有多种可能的解决途径（指标点2.3）；**（2）**能够针对特定需求确定目标，设计计算机系统框架、组成模块，合理组织/存储数据，基于适当的模型进行系统设计与实现，并体现一定的创新意识（指标点3.3）；**（3）**能够在解决方案中从技术、非技术（如经济、社会、健康、安全、法律、文化以及环境等）角度，对设计方案的可行性进行评价和分析（指标点3.4）；**（4）**能够采用科学方法对计算机系统及工程问题进行研究，通过实验对比、文献综合、归纳整理得到合理有效结论，并对其进行规范表述（指标点4.3）；**（5）**能够利用开发环境和工具，对计算机系统及工程问题进行模拟仿真和数据分析（指标点5.3）；**（6）**能识别、分析、评价特定需求的计算机系统在设计和实现中对社会、健康、安全、法律以及文化的影响，并明确自己应承担的责任（指标点6.2）；**（7）**能够评价计算机系统设计、开发、运行和维护对环境保护和社会持续发展的影响（指标点7.2）；**（8）**能够通过口头、文稿、图表等方式、陈述和表达自己的观点，能够就计算机系统及工程问题与同行和相关人员进行交流（指标点10.1）；**（9）**能够根据对工作内容和过程的记录与整理，撰写技术报告和设计文稿、陈述发言或回应质询（指标点10.2）；**（10）**了解计算机系统工程管理原理与经济决策方法，理解计算机系统项目的组织模式和实施过程，掌握项目管理原理和内容（指标点11.1）；**（11）**正确认识自主学习的必要性和重要性，认识到本专业是一个发展迅速的学科，具有自主学习和终身学习的意识（指标点12.1）；**（12）**具备自主学习新技术和新方法的能力，能够通过学习不断提高、适应信息技术和职业的发展（指标点12.2）。

4、论文各部分内容及时间分配：（共17周）

第一部分 ( 周)

第二部分 ( 周)

第三部分 ( 周)

第四部分 ( 周)

第五部分 ( 周)

评阅及答辩 (1周)

备 注

指导教师： 2021年12月10日

审 批 人： 2021年12月10日

摘要

近年来经济社会发展，人们生活水平不断提高，青少年身体发育水平明显提升，但其身体运动素质发展却在停滞甚至下滑，对此国家出台了一系列政策促进提高青少年体质健康，社会、体育部、学校对于青少年体质健康越来越重视。因此，本文从青少年体质健康数据入手，设计与实现了数集数据存储、分析、上报、体质素质综合评价功能的Web管理系统。

本文设计实现的体质健康数据管理系统主要服务于学校管理者、体育教师等用户群体，涵盖基本信息管理、体质测试和成绩分析上报的一系列功能。系统采用微服务架构、前后端分离模式开发，主要包括权限管理、基础数据管理、科目个性化、体质测试和统计分析模块，具有架构灵活、数据安全、可扩展性强等特点。基于此系统，各类用户可以以不同角色参与到体质健康数据管理的流程中，更好地了解评估体质健康状况。

本文首先调研了青少年体质测试标准和当前体质健康数据管理系统，针对数据管理效率低、灵活性不高等不足确定了研究方向，并对完成研究涉及到的Spring、Vue等相关技术进行了介绍，而后根据实际需求对系统设计划分了各个模块并使用用例图对各模块进行了详细的需求分析和整体概要设计，接着设计系统关键技术方案，基于RBAC模型的细粒度权限控制和RESTful风格的系统架构提升了系统的灵活性和可扩展性；在模块功能方面，使用类图、时序图等进行详细设计，扩展了个性化定制科目和动态报表功能，拓宽了系统的适用场景，最后完成系统前后端开发并对系统界面功能进行测试和介绍。该体质健康数据管理系统解决了体质健康数据管理低效的痛点，能更好地支撑体质健康大数据平台的建设，服务于青少年体质健康的发展，为个人健康、科学管理、政策制定服务提供帮助。

关键词：体质健康；数据管理；系统设计实现；Spring；Vue

Abstract

In recent years, with economic and social development, people's living standards have been continuously improved, and the physical development level of adolescents has been significantly improved, but the development of their physical fitness has stagnated or even declined. In response, the country has issued a series of policies to improve the physical health of adolescents. The Ministry of Society, Sports, Schools pay more and more attention to the physical health of young people. Therefore, this paper starts with the physical health data of adolescents, and designs and implements a Web management system with functions of data storage, analysis, reporting, and comprehensive evaluation of physical quality.

The physical health data management system designed and implemented in this paper mainly serves user groups such as school administrators and physical education teachers, covering a series of functions such as basic information management, physical fitness testing, and performance analysis and reporting. The system adopts a micro-service architecture and is developed in a front-end and back-end separation mode. It mainly includes permission management, basic data management, subject personalization, physical fitness testing and statistical analysis modules. It has the characteristics of flexible architecture, data security, and strong scalability. Based on this system, all kinds of users can participate in the process of physical health data management in different roles to better understand and evaluate physical health status.

This paper firstly investigates the youth physique test standards and the current physique health data management system, determines the research direction for the inefficiency and low flexibility of data management, and introduces the Spring, Vue and other related technologies involved in the completion of the research, and then The system design is divided into various modules according to the actual needs, and the use case diagram is used to carry out detailed demand analysis and overall outline design of each module, and then design key technical solutions of the system, fine-grained permission control based on RBAC model and RESTful style system architecture improvement The flexibility and scalability of the system are improved; in terms of module functions, class diagrams, sequence diagrams, etc. are used for detailed design, which expands the functions of personalized custom subjects and dynamic reports, broadens the applicable scenarios of the system, and finally completes the front-end and back-end development of the system. And test and introduce the system interface function. The physical health data management system solves the pain point of inefficient physical health data management, can better support the construction of a physical health big data platform, serve the physical and healthy development of young people, and provide help for personal health, scientific management, and policy formulation services.

**Keywords:** Physical health; Data management; System design and implementation; Spring; Vue

目录

[第 1 章 绪论 1](#_Toc103847550)

[1.1 背景与意义 1](#_Toc103847551)

[1.2 国内外发展（应用）现状 2](#_Toc103847552)

[1.2.1 青少年体质健康现状 2](#_Toc103847553)

[1.2.2 国外体质健康管理系统现状 2](#_Toc103847554)

[1.2.3 国内体质健康管理系统现状 3](#_Toc103847555)

[1.3 论文所做工作及思路 4](#_Toc103847556)

[1.4 论文章节安排 4](#_Toc103847557)

[第 2 章 相关技术分析 6](#_Toc103847558)

[2.1 Spring 6](#_Toc103847559)

[2.2 Vue 7](#_Toc103847560)

[2.3 MySQL 7](#_Toc103847561)

[2.4 MyBatis 7](#_Toc103847562)

[2.5 Redis 7](#_Toc103847563)

[2.6 本章总结 8](#_Toc103847564)

[第 3 章 体质健康系统需求分析 9](#_Toc103847565)

[3.1 系统需求概述 9](#_Toc103847566)

[3.2 系统功能性需求 10](#_Toc103847567)

[3.2.1 用户与权限 11](#_Toc103847568)

[3.2.2 基础数据 12](#_Toc103847569)

[3.2.3 科目个性化 13](#_Toc103847570)

[3.2.4 体测模块 14](#_Toc103847571)

[3.2.5 统计分析 15](#_Toc103847572)

[3.3 系统非功能性需求 16](#_Toc103847573)

[3.4 本章总结 16](#_Toc103847574)

[第 4 章 体质健康系统方案设计 18](#_Toc103847575)

[4.1 系统整体方案设计 18](#_Toc103847576)

[4.2 数据库设计 20](#_Toc103847577)

[4.3 系统权限设计 26](#_Toc103847578)

[4.3.1 界面权限设计 27](#_Toc103847579)

[4.3.2 接口权限设计 28](#_Toc103847580)

[4.3.3 数据权限设计 28](#_Toc103847581)

[4.3.4 RBAC模型 29](#_Toc103847582)

[4.4 前后端接口设计 29](#_Toc103847583)

[4.5 系统功能设计 31](#_Toc103847584)

[4.5.1 用户与权限模块设计 31](#_Toc103847585)

[4.5.2 基础数据模块设计 34](#_Toc103847586)

[4.5.3 科目模块设计 37](#_Toc103847587)

[4.5.4 体质测试模块设计 38](#_Toc103847588)

[4.5.5 统计分析模块设计 40](#_Toc103847589)

[4.6 本章总结 40](#_Toc103847590)

[第 5 章 体质健康系统实现与测试 41](#_Toc103847591)

[5.1 系统开发环境搭建 41](#_Toc103847592)

[5.2 系统主要功能实现 41](#_Toc103847593)

[5.2.1 用户与权限功能实现 41](#_Toc103847594)

[5.2.2 基础数据功能实现 42](#_Toc103847595)

[5.2.3 科目功能实现 43](#_Toc103847596)

[5.2.4 科目组功能实现 45](#_Toc103847597)

[5.2.5 体质测试模块实现 45](#_Toc103847598)

[5.3 系统测试 47](#_Toc103847599)

[5.4 本章总结 50](#_Toc103847600)

[第 6 章 总结与展望 52](#_Toc103847601)

[致谢 53](#_Toc103847602)

[参考文献 54](#_Toc103847603)

# 绪论

## 背景与意义

随着社会发展和城市化进程推进，人们物质生活条件日益改善，但体质健康状况却不容乐观，尤其是对于青少年来说。青年兴则国兴，青年强则国强。青少年阶段不仅是学习知识、增长技能的黄金时段，更是塑造青年精神气质最重要的阶段，但由于学业压力、电子产品等因素影响，青少年户外活动、体育锻炼的时间偏少，整体身体素质特别是耐力、速度、力量不断下滑[1]。

青少年是祖国未来的希望，一直以来，国家对于青少年的体质健康问题也给予了高度重视。1985年以来，国家先后组织七次全国学生体质与健康调研；我国2007年颁布并在2014年修改的《国家学生体质健康标准》对我国学生体测项目作出了具体要求[2]；2010国务院于2010年审议通过了《国家中长期教育改革和发展规划纲要（2010-2020年）》[3]、2012年颁布了《国务院办公厅转发教育部等部门关于进一步加强学校体育工作若干意见的通知》[4]。2016年6月国务院公布的《全民健身计划（2016—2020年）》明确指出要提高青少年的身体素质，同年印发了《“健康中国2030”规划纲要》推进建设全民健康的目标[5-6]；2021年9月，教育部召开发布会介绍第八次全国学生体质与健康调研有关情况，而各地教育厅、学校对进一步加强学生体质测试工作的通知更是屡见不鲜。

在这样的背景下，青少年的体质健康越来越受到社会的关注，体质测试成为学校的重要工作和学生的考核指标，以更好地了解学生体质健康情况，促进提高身体素质。当今学生在各个年龄阶段都有不同规模程度的体质测试，体测数据将上传到省市体质监测平台，北京、上海、广东、江苏、浙江、河南、河北等多个重点省市已经成功搭建建立起了地方性的体质健康数据服务平台，旨在通过数据分析结果为学生体质健康提供建议和指导，为国家政府出台政策提供数据支持和咨询建议[7]。在整个过程中，体质测试数据保存和处理一直是整个工作的重要环节，但是目前部分地区对于体质测试数据的管理方式仍然比较低效，部分较为先进的地区对于数据也仅仅局限在存储，对测试数据的管理效率较低，数据的分析和再利用一直是未能解决的问题[8]。学生和学校仅仅在测试过程中能直观的对体质健康素质有直观的感受，但无法依靠数据对问题进行更深的理解和制定相应的措施。

因此，如何更好的对体质测试数据进行管理和有效利用是当前需要解决的问题。本课题针对目前国内外对学生体质健康理论、体质健康管理系统现状的研究分析，以青少年体质健康测试数据为基础搭建数据管理软件系统，服务于学校体育工作者，向上对接大数据体质监测平台，使青少年体测数据的录入、管理、上报更加智能化、统一化、高效化；同时，对于体测数据进行有效处理分析、可视化等，为学生和学校提供直观的体质健康情况，客观地展现出青少年整体体质健康状况，对研究青少年的真实健康状态具有实际的应用价值。整个系统的数据最终应用于青少年体质健康的发展，为个人健康、科学管理、政策制定服务提供帮助。

## 国内外发展（应用）现状

### 青少年体质健康现状

国家对青少年体质健康状况十分重视，并对此发布了一系列战略方针。但可惜的是，多种政策出台下，青少年体质健康情况仍不容乐观。根据2014年全国学生体质健康调研结果[9]，大学生身体素质正在不断下降;视力不良问题严重且呈现低龄化倾向;各年龄段学生肥胖检出率持续上升。2019年，教育部部署展开了第八次全国学生体质与健康调研工作[10]，调查结果显示学生握力水平有所下降、大学生身体素质下滑等是亟待解决的问题。青少年体质健康问题越来越严重，缺少运动、视力、久肥胖等问题尤其显著。针对我国青少年体质健康下降的趋势，国家也在积极出台政策促进和改善青少年体质健康现状，教育部门、学校体测工作越来越被重视，但是政策的实施主体是学校，过度地关注了体质监测过程中的测量与数据上传[11]，而忽视了体测前期学生意识教育和后期的指导实施，在体质健康的监测方面存在漏洞，相关学者提出构建青少年体质健康评价的监测及数据处理方案。在青少年体质健康的监测和数据管理中呈现出“数据化“、”信息化“的趋势[12]。加之计算机技术飞速发展，大数据时代到来，体测数据也成为重要的、可分析的资源，而高效地存储、管理、上报青少年体测数据，是从中挖掘有效信息的前提。

### 国外体质健康管理系统现状

在日本家庭普遍享有健康管理服务并建立档案，日本学生在入学时都会进行健康体检并存有档案，美国对青少年体质健康的关注比较早，在20世纪90年代，美国教育界就提出了终身化、个性化的教育计划[13]。1982年，美国就开发了FITNESSGRAM[14]青少年体质健康测评系统，目前仍普遍作为学生健康体能的评价工具，该系统除了具有保存测评数据和统计分析的功能外，还能生成个性化的测评报告。同时，该系统还能评估学生的体力活动情况和营养状况，给出的面向家长的测评报告可以让家长更多的了解学生的体质状况，从而针对性地制定运动方案和饮食方案。法国的体质测试管理软件是通过网络将数据上传到服务系统上，用户可以在诸如IOS、Android的App上查看相关健康建议[8]。近年来随着大数据、深度学习等技术的发展，相关学者开始考虑到将技术应用到体质健康系统中。2019年计算机科学与电子技术国际学会指出，必要实时监测大学生的健康状况和体质，收集和处理大学生体质健康系统信息，利用数据分析、数据挖掘相关技术构建体质管理系统[15]。WangHui[16]等考虑利用大数据技术搭建学生健康管理模型，将学生体质健康的数据采集、监督、数据分析、数据应用有机结合，利用计算机对学生以往所有的医学信息进行日常管理，构建基于网上问答的医生推荐模型,为不同体质的学生给出具体的健康建议。

### 国内体质健康管理系统现状

我国在体质健康管理方面起步较晚，从最早的仅用于学校上报和汇总统计的单机系统逐渐发展为可容纳每年全国2亿多学生数据的网络版系统，但我国的体质健康管理系统偏向于对数据的宏观统计分析，对学生体质健康的反馈和指导功能有待提高[17]。2013年，教育部发布《关于建立健全全国学生体质健康监测与评价体系的通知》，国内青少年体质健康研究变成热点问题，各省市开始重视学生体质健康管理平台的建设，相关研究也越来越多。刘小卉[18]针对学生体质测评成绩管理的低效性，采用了J2EE分层架构、Struts框架设计实现了中学生体质测试测评管理系统，这是较早的使用IT技术解决手工整理体质测试数据问题的实践，减少了人工操作，降低了体测数据管理的出错率，并提供了小部分统计分析的功能。贺慨[19]设计和实现了基于.NET框架开发的C/S架构的学生体质健康运动处方系统，在日常体育教学中，大大减轻了学校教师授课负担，使得体育教学更加科学，同时为体质健康系统的设计提供了参考。徐刘柱[20]基于Java语言和Flex前端技术，设计和开发完成了湖北省大学生体质健康管理平台，为湖北省大学生体质健康信息的“收集、诊断、评价、实施”提供可操作性的参考依据。鉴于当时计算机技术发展限制，这些系统功能比较偏向于对数据的汇总，自动上报和宏观的统计分析上，对于测试数据的利用程度有限，系统没有对政府政策以及学生个人的体质健康提供较有效的参考和指导。2015年，朱广涛[21]在体质健康测试系统的设计实现中采用了数据挖掘技术，挖掘出数据中隐含的规则,为决策者制定决策提供数据支持，同时为后续学者的研究提供了借鉴和参考。随后体质健康系统的相关研究如雨后春笋般涌现。2017年，黄雨博[22]基于Loushang5平台设计了集数据采集、管理、应用等功能于一体的系统，具有按各种要求统计分析的功能，各级教育管理部门和学校可以全面了解评估大中小学校及学生个人的体育教育状况。2020年，张亚琦[23]利用了决策树C4.5、关联规则Apriori等算法对体能指标之间的相关性进行研究，使体质健康系统对于学生的指导建议更加科学、合理。除学术方面之外。由此可见，体质健康管理系统的构建已经是顺应时代需要、国家政策的必然趋势，并且将会朝着数据化、智能化的方向迈进。

## 论文所做工作及思路

本文对基于Web的体质健康数据管理系统的设计与实现工作进行了详细论述。首先充分调研了青少年体质测试标准及现有体质数据管理系统，分析了当前体质健康工作存在的问题，针对当前痛点确定了业务需求，查阅和学习了基础理论和相关技术，对系统进行需求分析和功能设计，采用前后端分离开发模式，使用SpringBoot、Vue等技术，完成系统代码编写工作，最后进行测试，最终实现了包含体质健康数据的存储、分析、上报、体育综合素质评价等功能的体质健康系统。具体进行了以下工作：

1. 对体质健康数据管理系统进行了需求分析，调研了当前青少年体质健康测试的流程、测试科目及成绩评定标准，分析整理作为本研究的基础参考。
2. 确定了系统的功能性需求和非功能性需求，利用用例图分析并设计各功能模块，然后基于此设计了系统总体架构。根据系统的实际需求、技术特点和业界主流技术，采用B/S架构，MySQL数据库、Java语言、Spring框架、Vue等技术实现。
3. 在详细的需求分析上，对数据库和前后端接口进行设计，分析系统整体工作流程，设计各部分之间的数据流向，结合RBAC[24]模型设计系统的页面权限、功能权限、数据权限。通过类图、时序图等方式对系统各个功能模块进行详细设计。
4. 对系统设计的功能进行具体实现，包括用户和权限、基础数据、科目模块、体质测试和统计分析功能，完成了前后端开发工作。
5. 根据系统运行的环江对系统功能进行了测试工作，编写测试用例进行测试，验证体质健康数据系统是否满足了需求分析的功能性与非功能性需求。

## 论文章节安排

论文共分5章，描述了从项目背景到体质健康数据管理系统设计、实现和测试的整体流程。

第1章，绪论。主要说明了本文的项目研究背景及意义，介绍了体质健康系统的国内外现状，最后阐述了本文的主要工作及实现思路。

第2章，相关技术介绍。针对系统实现的相关基础理论，技术框架等进行了分析介绍。

第3章，系统需求分析。对业务流程进行分析和抽象，设计了多个模块，完成系统的功能性和非功能性需求分析。

第4章，系统设计。对系统进行概要设计和详细设计，进行了系统的整体架构设计，数据库设计、前后端接口设计、系统权限设计合各功能模块的详细设计。

第5章，系统实现与测试。对系统实现的功能进行测试工作，展示介绍系统的功能页面和测试用例，验证系统的实现结果。

第6章，总结与展望。总结本文的研究内容，分析了体质健康数据系统的完成情况与不足之处，并展望未来发展。

# 相关技术分析

B/S架构的体质健康数据管理系统按照当前主流前后端分离的思路进行平台的搭建和功能的研发，前端用户交互使用TypeScript开发，后台服务器使用Java语言。本章主要对体质健康系统设计实现中运用到的技术进行详细阐述，包括Web应用开发所用到的Spring后端框架、Vue前端框、MySQL数据库，Redis缓存中间件等相关技术知识。

## Spring

Spring是一个支持快速开发JavaEE应用程序的框架。它提供了一系列底层容器和基础设施，并可以和大量常用的开源框架无缝集成，可以说是开发JavaEE应用程序的必备。Spring在JavaEE几乎具有完全的统治力，已经成为JEE事实上的标准，全世界的开发人员都在使用Spring框架开发各种应用。Spring是一个封装很清晰的分层架构，包括SpringCore、SpringDAO、SpringAOP、SpringWeb等模块。Spring的CoreContainer包含了Core、Beans、Expression等模块，其中Core和Beans是Spring设计的核心基石，这两者的结合使其具备了控制反转和依赖注入的特性，减少了应用开发的繁杂配置与程序设计的耦合度。下图是Spring架构如下图2-1所示：

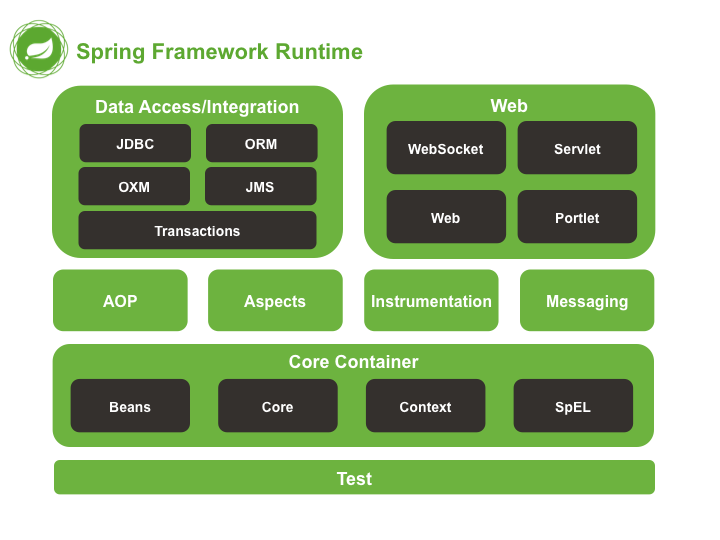


图2‑1Spring架构

注：图片来自Spring官方网站

随着Spring越来越受欢迎，在SpringFramework基础上，又诞生了SpringBoot、SpringData、SpringSecurity等一系列基于项目，集成了开发过程中所有常见的功能，大大简化了开发步骤，提高了开发效率。

## Vue

Vue.js是一款轻量MVVM框架，主要是数据驱动+组件化的前端开发。在MVVM框架下，视图和数据是不能直接进行交互的，通常都是通过ViewModel来进行通讯。而ViewModel需要实现一个观察者，来监听数据的变动，并且通知对应的视图进行改变。当我们用户来操作视图ViewModel也需要通知对应的数据来做持久化操作，这就是所谓数据的双向绑定。使用MVVM模型可以针对复杂交互逻辑的前端应用可以提供基础的架构抽象，并且可以通过Ajax数据持久化，保证前端用户体验。MVVM模型如图2-2所示：

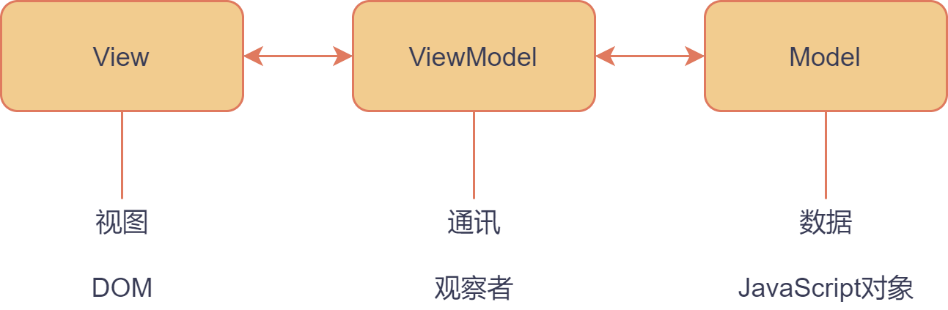


图2‑2MVVM模型

## MySQL

MySQL是当下最流行的一种关系型数据库管理系统，它功能强大，提供了多种数据库存储引擎，适用于不同的应用场合，支持跨平台，运行速度显著，使用成本低，是目前使用最广泛的一种数据库。同时，其社区支持度高，有众多的学习者和使用者，方便地对接各种编程语言。因此，本系统采用MySQL对结构化的数据进行存储和管理。

## MyBatis

MyBatis是一款优秀的持久层框架，工作在应用程序与数据之间，它支持自定义SQL、存储过程以及高级映射。MyBatis免除了几乎所有重复繁杂的JDBC代码以及设置参数和获取结果集的工作，同时，MyBatis将SQL语句集中在XML文件中配置和编写，分离了应用程序业务逻辑和数据库访问逻辑，方便功能的实现、维护和扩展。

## Redis

Redis是现在最受欢迎的NoSQL数据库之一，常用来做数据库、缓存中间件以及消息中介。它使用key-value存储系统，包含多种数据结构，如字符串、哈希表、列表、集合等。基于内存运行，性能高效，同时支持不同级别的持久化。Redis也支持分布式、数据库事务，高并发读写、Lua脚本等。被广泛地用于缓存中间件、提供高可用服务。

## 本章总结

本章主要阐述了体质健康系统设计实现过程中涉及到的相关技术以及其原理、特点和优势。通过对技术原理的分析为系统的开发实现提供了相关技术经验，为后续系统的实现奠定了基础。

# 体质健康系统需求分析

在体质健康数据管理系统设计实现中，需求分析是非常重要的阶段，是后续的设计实现参考，主要是对系统整体业务进行分析，需要解决哪些问题，达到系统使用者的要求，从功能性与非功能性方面对系统需求进行详细讨论。

## 系统需求概述

体质健康监测平台服务于体育管理部门、学校管理者、体育教师等用户群体。为用户群体提供体质健康数据的采集、分析、上报、体育综合素质评价等应用体系。本系统为整个体质健康监测平台的子系统，旨在便于体育老师、体育管理部门管理学生体质测试数据，提高体质测试数据的录入效率，提高录入准确率，形成体质健康数据轨迹，更好地服务于体质健康检测平台。

系统的需求围绕体质测试展开，管理员录入学生、班级等基础数据到系统中，教师选择科目、班级等创建体质测试，参与测试的学生可查看体测完成情况，体测完毕后老师将体测数据导入到本系统中。该部分包括体质测试的管理、体测详细信息，体测数据部分包括成绩录入与管理、体测成绩的统计、运动能力分析，体测报告查看，这些统计报告要求以图表、PDF文档等格式展现，并支持导入导出。系统工作流程及各角色的工作如下图3-1所示。

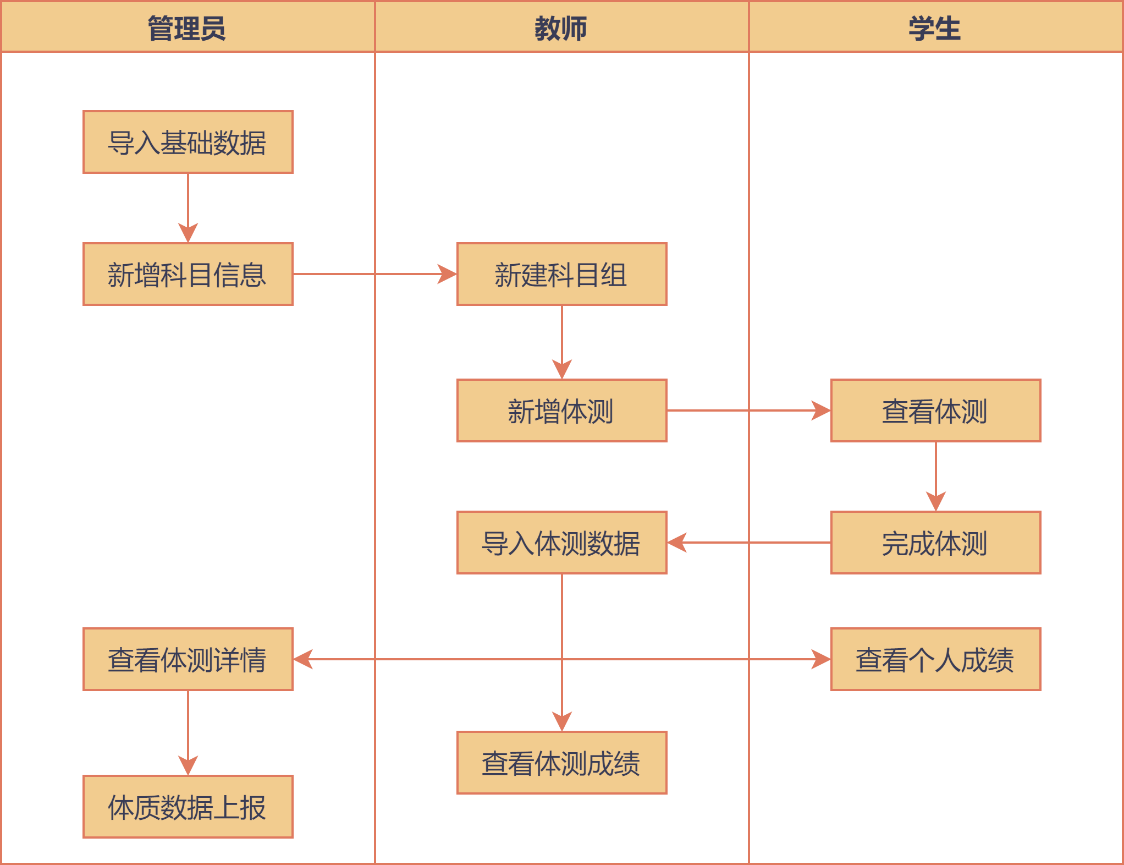


图 3‑1系统工作流程

要完成以上工作，系统需要以下信息作为支撑：班级信息、教师信息、学生信息、科目信息等与体质测试相关的数据信息，系统中数据的流动如下图3-1所示：

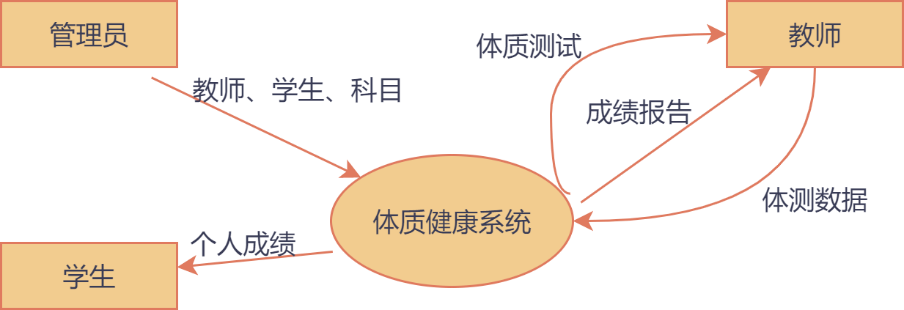


图3‑2系统数据流动

通过分析，系统各类数据总结如下：

1. 基础数据：包含管理员、教师、学生、班级、学院，用户及其所属部门组织信息。
2. 角色与权限：对于不同的用户，其角色和权限不同，用户页面和操作权限也不同。
3. 科目：体质测试的科目信息，考虑到系统的灵活性，科目又包含评分标准，科目组等信息。
4. 体质测试：依赖于基础数据、科目等，是系统成绩数据的来源。

对于系统的各类角色所拥有的功能可大致总结如下：

1. 学生：查看系统班级、教师、科目、体质测试、成绩信息，体质健康分析，维护个人基本信息，对于系统的数据几乎都为查看权限等。
2. 教师：查看基础数据、科目等信息、维护科目组、体质测试信息，体制数据导入导出，学生成绩分析、维护个人信息等。
3. 管理员：基础数据的维护，学生、班级、教师等信息的导入导入，系统角色与权限控制，科目管理，成绩查看、导出和分析、维护账号信息等。管理员具有系统绝大多数数据的修改权限。

系统的主要功能即对上述数据实现管理，主要包含用户权限、基础数据、科目管理、体测管理、统计分析这几个部分，从而更好地开展体质测试、记录和管理体测结果数据。下面将对各个部分的详细功能需求进行介绍。

## 系统功能性需求

本系统主要包含五大功能模块，各模块下又包含若干子功能模块，对应用户主要有管理员、教师和学生，不同用户拥有不同的页面权限、菜单权限和数据权限。下图3-2为系统功能模块图，其中子模块未画出。

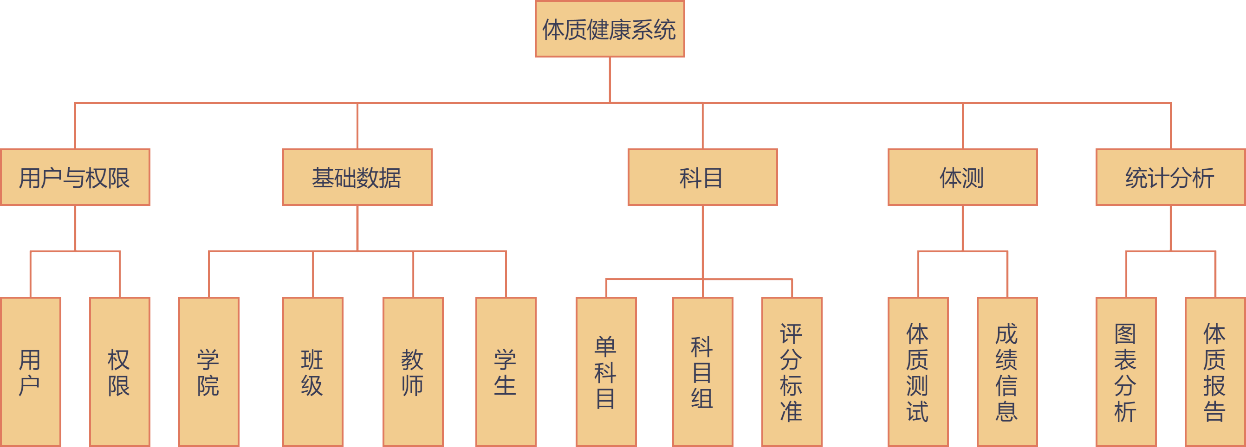


图3‑3系统功能模块

通过上图可以加深对系统整体功能模块的认识，接下来将对各个模块详细的需求进一步阐述。

### 用户与权限

体质健康数据系统使用用户主要有管理员、教师、学生，均需要登录认证后，系统识别身份，提供不同的用户界面与功能权限。该模块中，用户子模块包含所有用户通用的功能，如登录、找回密码、个人信息。教师、学生在该模块的功能权限相同。对于管理员来说，负责整个系统的管理和维护，所以需要控制不同用户的访问权限，可对其他用户访问不同资源的权限进行修改。用户与权限模块的用例图如下3-3所示。

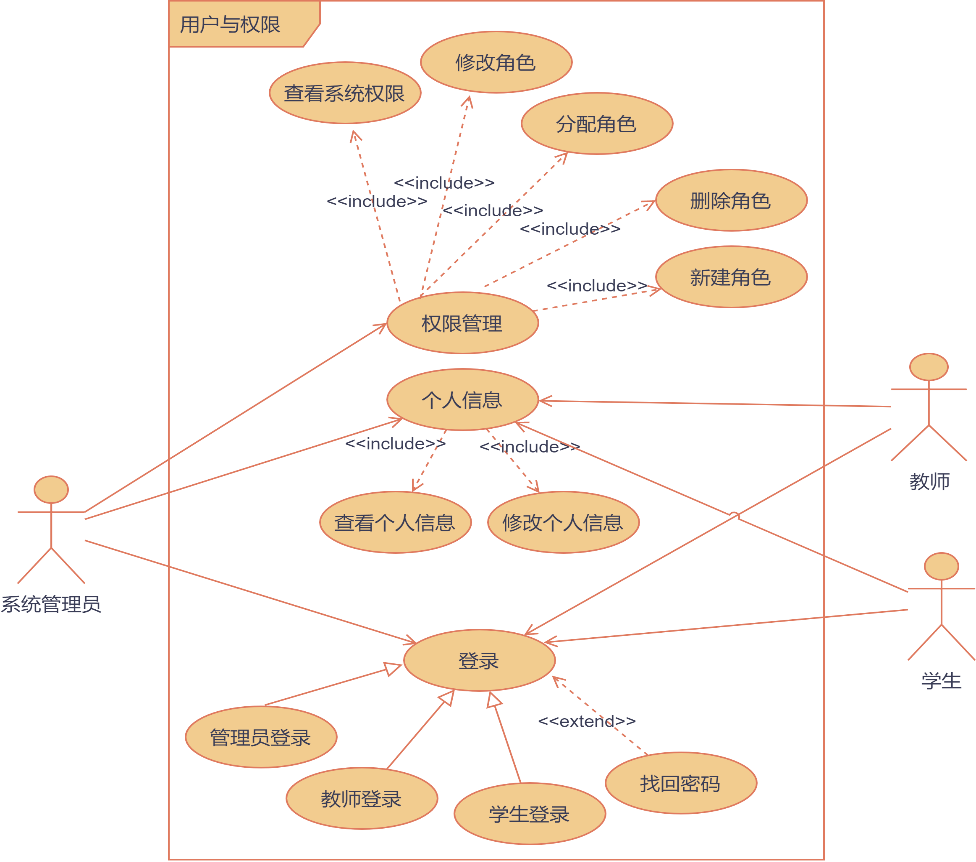


图3‑4用户与权限用例

管理员所操作的用户权限包括其他管理员与教师，该系统中学生默认只有部分数据的查看权限，即权限是固定的。管理员通过管理角色与权限的关联关系，再将角色分配给教师为教师配置权限。对权限的控制可具体到某个按钮的权限管理，例如可以设置某位教师具有下载学生基础数据的权限但不能上传该数据，在前端页面上则是控制下载按钮的可见性，后台则需要对控制该教师对于该接口的访问。

登录及个人信息等功能是所有用户共有的，但对于不同的角色，其看到的信息、可修改的范围存在细微差异。

该模块是用户访问本系统及保障系统资源安全的基础，未登录认证的用户无法访问系统的任何资源。

### 基础数据

基础数据是系统正常运行及体测业务的基础，包括学院、班级、教师和学生四个部分，教师按班级组织学生参与体质测试，这是系统体测数据的基础支撑。系统正常运行必须先由管理员将上述数据导入，教师、学生才能参与使用。

基础数据模块下包含学生管理、班级管理、学员管理、教师管理四个子模块。每个模块包含文件模板下载、Excel数据上传以及删改查功能，查询功能支持多条件查询，需求结构比较接近。下图3-5为基础数据模块中的教师管理用例图，其他子模块的用例图大致相同。

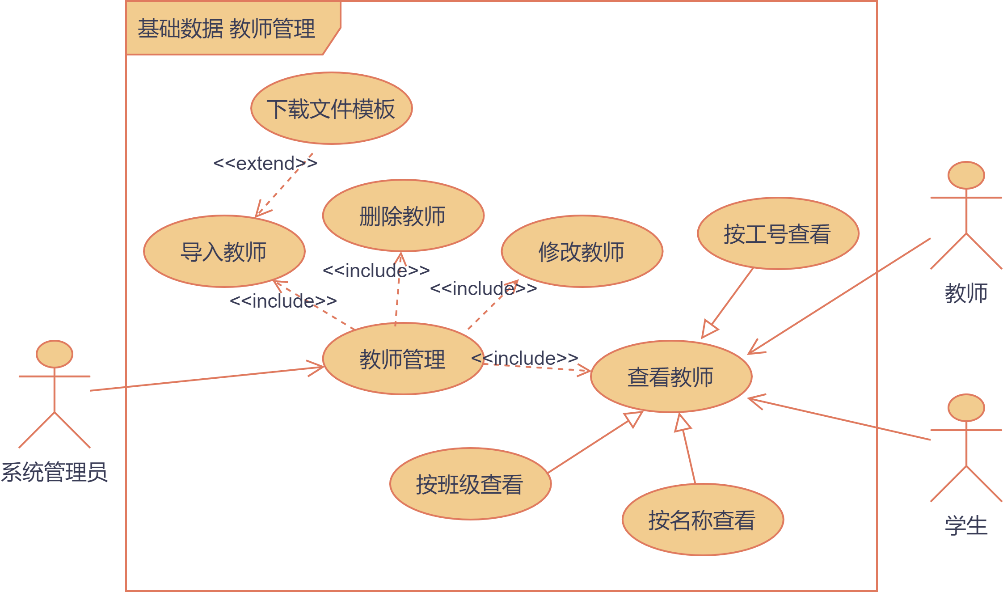


图3‑5基础数据教师管理用例图

基础数据模块的主要功能在管理员，管理员可以对学院、班级、教师、学生数据进行录入、查看、修改、删除等。数据录入之后，教师和学生方可登录使用本系统。系统管理员拥有系统中所有基础数据的操作权限。

对于教师，在管理员未授予权限的情况下，只能查看学院、班级、学生、教师信息，无法进行任何修改操作。考虑到教师可以是学院普通教职工或者负责人，两者应当拥有不同权限，后者允许所负责学院下的修改功能。其中，所有查看、修改都需要数据权限的控制，即教师只能操作所任课的班级或者负责学院下的数据。

对于学生，在该模块中，拥有的权限很少，只能对学院、教师、班级三个部分进行查看，无法查看其他学生的信息。下图3-6为基础数据模块中学生管理用例图。

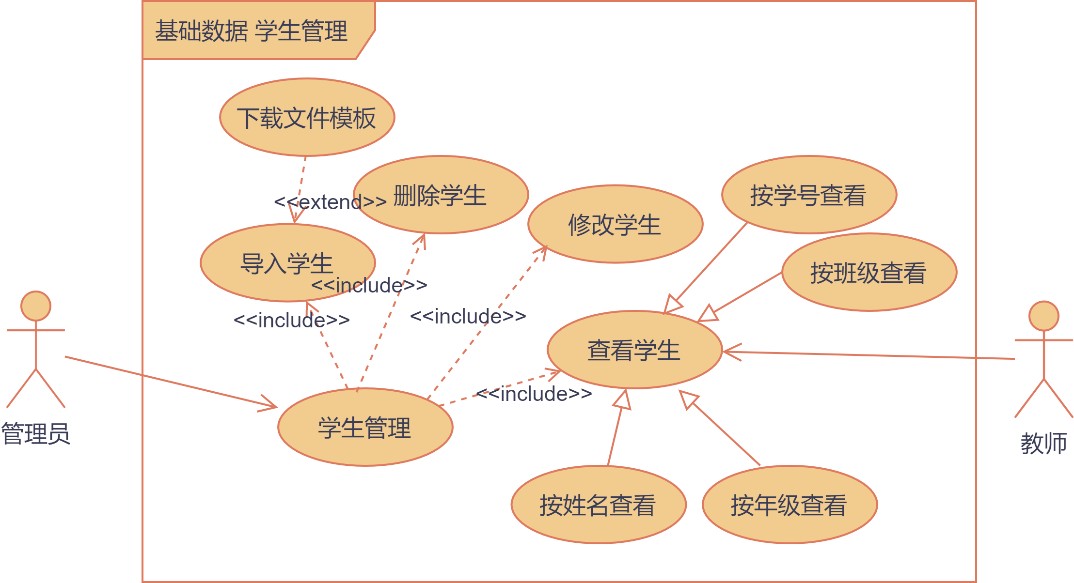


图 3‑6基础数据学生管理用例图

上图中没有学生这个参与者，学生不能查看其他学生的信息。

### 科目个性化

现有的体质测试中，不同年级测试科目不尽相同，相同科目对于不同性别、不同年级，评分标准也不相同。对于指定年级性别的学生，需要测试的科目是一定的。对于体育教师而言，希望能有针对性地对学生地运动体能素质进行考察检验，所以在体质测试中，教师希望可以个性化选择科目；同时，随着体质健康管理的发展，未来测试的科目可能动态变化，例如某一年级性别的测试科目进行改动或其评分标准发生变化，此时系统应当能跟随上述变化继续使用。除此之外，当教师选择科目组创建体质测试后，系统应当能够自动识别每个学生需要测试的科目，当教师进行体测数据导入时，系统需要自动判定每个科目的成绩与评价等级。

综合上述分析，为了满足科目基本信息、测试对象、评分标准的动态变化需求，实现个性化科目配置，故将科目相关信息单独抽取为一个模块，维护和管理科目基本信息、测试对象、评分标准和科目组信息，以更好的服务于体质测试工作。

本模块的基础为单个科目信息，科目关联的测试对象为某年级的男生或女生，不同的科目和测试对象关联多条评分标准。在体质测试中，可选择多个科目组成科目组，方便复用和管理。考虑到测试对象为固定的小学到大学的学生，并未单独抽取用例，但测试标准为分段区间，每个科目有对应年级、性别的评分标准，故单独抽取为一个子模块。所以，科目模块的用例图如3-7所示。

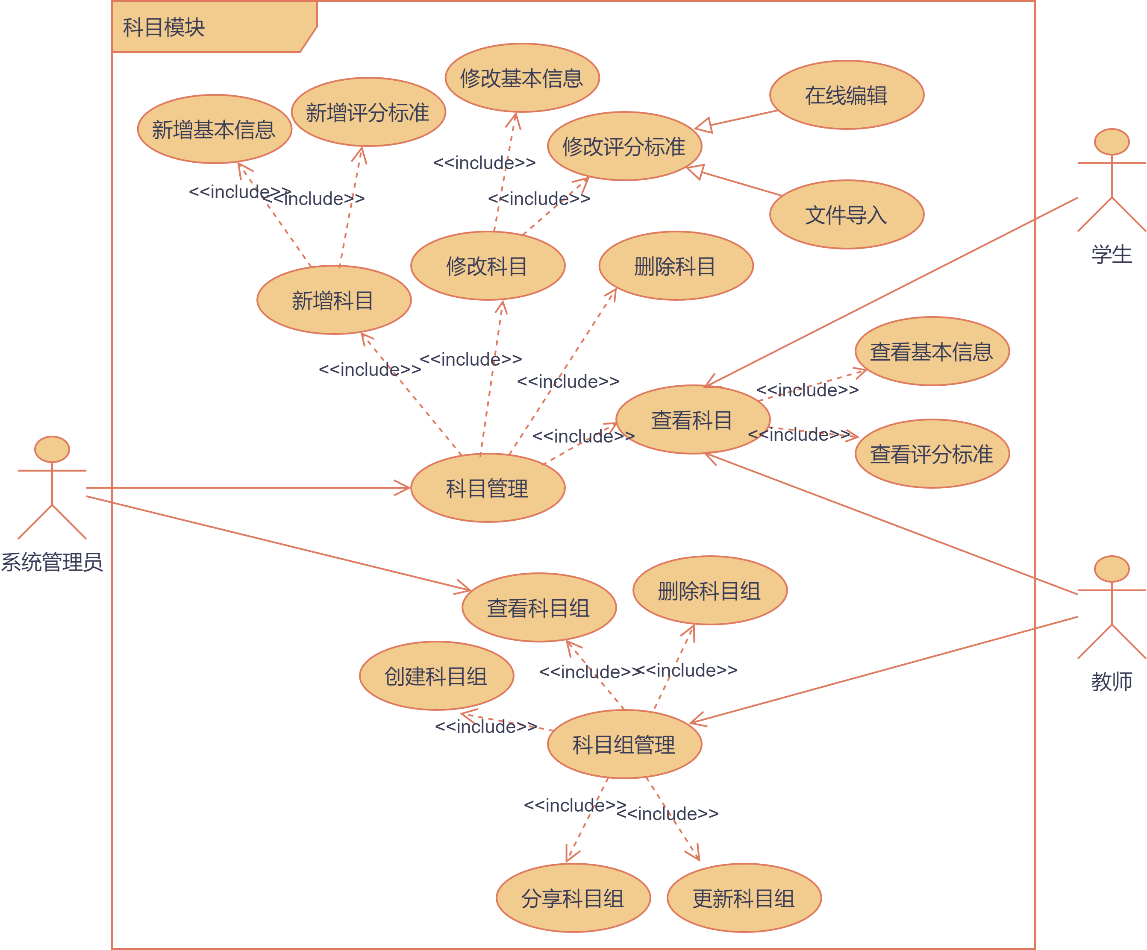


图3‑7科目模块用例图

所有的科目基本信息及其对应的关联年级、性别的评分标准，只能由管理员操作，教师和学生只能查看，没有修改的权限。

管理员对于科目的管理，均包含科目基本信息及其评分标准两个部分，其中，修改评分标准支持在线编辑和从外部文件导入。

科目服务于体测，管理员和教师可选择系统中已有的科目组合成科目组，对科目组进行管理，其中，管理员创建的科目组默认对所有教师可见，教师创建的科目组仅对自己可见，科目组分享之后对其他教师可见。所有科目组的修改权限只有创建者本人拥有。

学生在该模块中只具有科目基本信息和评分标准的查看权限。

### 体测模块

体质测试模块为系统的核心功能，基础数据和科目模块本质上为该模块服务，一次体质测试涉及到教师、班级、学生、科目、科目组等数据，是系统中业务最集中、需求和实现最复杂的一个部分，而该部分的体测数据上传、存储、体质测试成绩导出、统计分析、数据上报等则是系统设计实现的根本需求。该模块作为系统的核心业务，其用例如下图3-8所示。

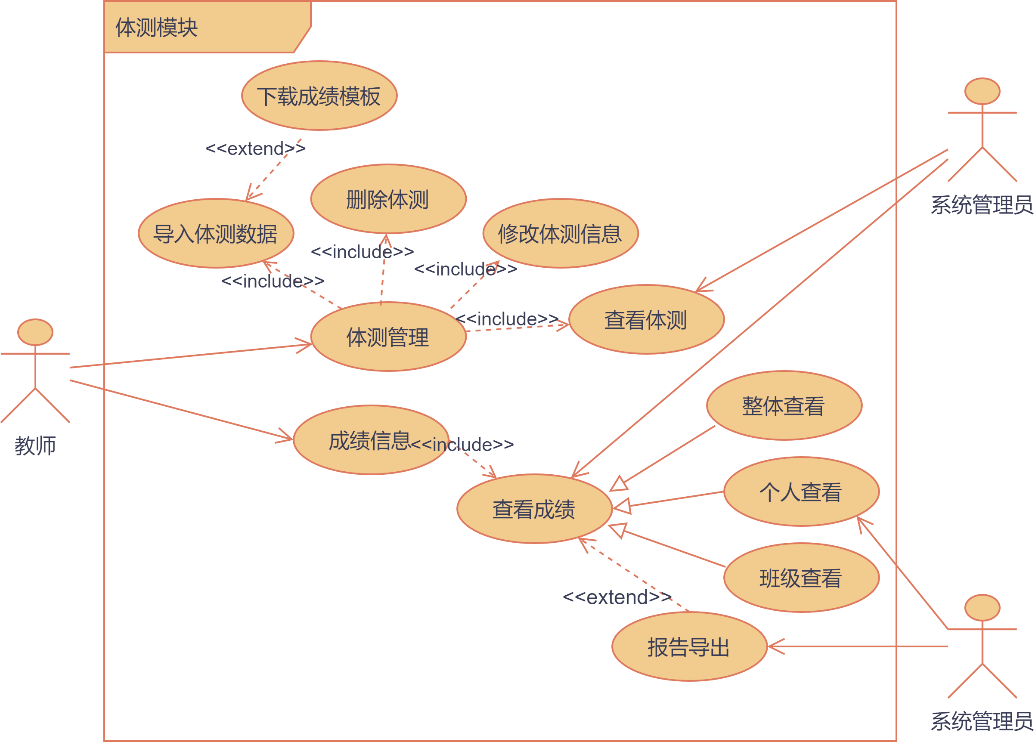


图3‑8体测模块用例图

体质测试仅能由体育教师管理，在体测信息界面，教师可查看参与测试人数、科目数、完成情况等基本信息，体测完成后，教师可下载模板文件填充体测数据并导入系统，统自动根据年级、性别、科目评定成绩。教师可按多维度查看成绩情况，包括整体情况、班级情况和学生个人成绩详情。

学生即可在体测信息中查看完成情况，包括已完成、未完成体测、未完成科目等。待教师导入体测数据后，学生同可以在体测成绩界面查看个人成绩信息、运动素质分析、测试报告导出等。

管理员在本模块中仅有体测基本信息和成绩的查看权限，不参与任何修改功能。

### 统计分析

体质健康数据管理系统除了可以对数据进行存储、上报之外，还具备一定的统计分析功能。为了实现对体测成绩数据的有效利用，从中挖掘更深层次信息，智能生成运动处方，对学生个人的体质健康提供参考和建议，为学校展开体育工作，促进提高学生身体素质给与指导。

统计分析模块包含两个子模块，数据统计和数据分析。

数据统计即对体测成绩数据进行简单宏观的统计分析，例如优秀率、及格率、平均分等，在界面上以表格、柱状图、条形图等方式展现，可以对体质测试成绩有一个整体的分析情况，并能将对某一科目、班级或者体测的统计结果进行导出。

数据分析则是在数据统计的基础上，利用数据挖掘、机器学习等方法，对体质健康数据进行建模深度分析，智能生成运动处方[25]，对个人体质健康提供指导建议。

下图3-9展示了统计分析模块的用例。

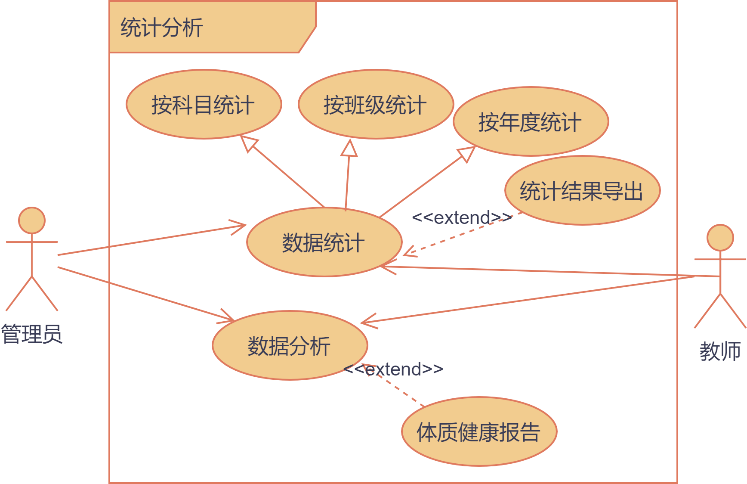


图 3‑9统计分析模块用例

该模块不涉及任何系统数据的修改，只有教师和管理员有查看权限。

## 系统非功能性需求

体质健康数据系统的功能性需求对管理员、教师、学生的功能通过用例图形式化描述，是系统功能的直观体现，而系统非功能性需求是指系统在满足体质健康数据管理的业务需求以外的功能特性，不能通过直观的方式表达出来。非功能性需求包括易用性、安全性、高效性等。根据体质健康数据管理系统的使用场景和用户需求，将其非功能性需求描述总结如下：

1. 安全性

系统中包含用户个人信息等重要数据，系统必须对外界攻击和干扰具有较强的自我保护能力，主要通过对前端代码进行防护以及后台服务严格的认证鉴权实现。除此之外，系统在操作过程中会对操作人员实现有效的操作记录跟踪，确保系统信息安全。

1. 易用性

系统使用应尽可能简单，同时，系统的图表类功能需要保证图表的美观性、文件类功能需要对用户文件进行严格的数据校验并给予友好的消息提示。

1. 高效性

系统的响应时间必须在可接受的范围内，由于体质测试关联其他数据较多，且随着系统的使用，学生的成绩数据条数增长迅速，需要保证数据库的查询时间。可通过优化SQL语句、建立查询索引来保证。

## 本章总结

本章节首先以数据流图分析了体质健康数据系统的整体需求，对系统的需求进行功能性与非功能性的划分。其中，对于功能性需求的各个模块进行了细致的需求分析，从多方面对于非功能性需求进行了陈述。

# 体质健康系统方案设计

前文已经较为详细地介绍了系统的需求，本章将针对系统需求进行总体方案设计，总体设计需要考虑实现难度、安全性、易用性等因素，系统设计经过多次修改最终确定。

## 系统整体方案设计

通过对体质健康数据管理系统的需求分析，本系统为主要应用于学校内部，服务于体育工作的Web系统，考虑到当前Web开发主流技术、系统实现难度，本系统采用了基于Vue、Spring、MySQL等技术的分层体系架构，前后端分离模式开发，系统架构图如下所示：

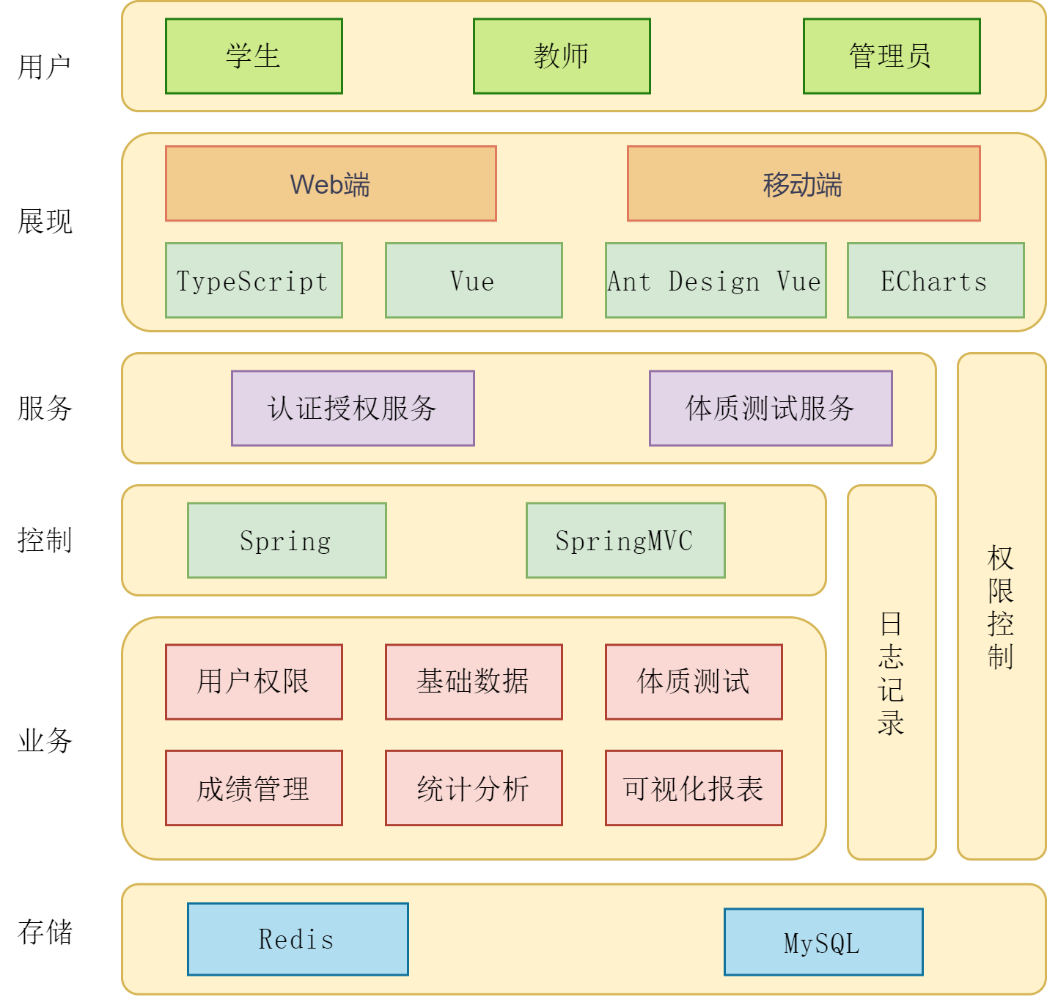


图4‑1系统整体架构

体质健康数据管理系统包含三类用户，用户角色、权限的不同，系统界面、可操作资源也不同，系统支持配置教师对某一接口功能的访问权限。

体质健康数据管理系统的前端采用Vue和Antd开发。Vue采用组件化的开发思路和数据双向绑定，面向数据编程，使前端开发变得简单高效。且Vue3经过优化，建议并提供了对TypeScript的支持，为JavaScript添加类型批注、编译时类型检查、类型擦除等特性的语言扩展，使JavaScript向静态语言靠近，大大有利于应用的开发和后期的维护。结合基于Vue的UI组件库Antd、相关插件工具和Ajax等成熟的技术，开发的单页面应用不仅可以实现系统的复杂需求，同时兼容不同的浏览器内核，页面结构也可以自动适配Web端和移动端。

系统主要使用场景在学校，日常访问量较稳定。由于教学活动时间较为集中，在体质测试等时段，服务器的并发压力较大。所以系统拟采用微服务架构，使用Nginx作为代理服务器。其中学生、班级、教师等基础数据，系统的认证鉴权由认证授权服务提供，体质测试服务专门用于提供系统主要业务功能。微服务设计有利于系统模块间的解耦和扩展，多个应用可以降低系统的压力。

系统后端使用后台开发脚手架Springboot、持久层框架MyBatis、关系型数据库MySQL、缓存中间件Redis等技术。下图4-2为系统服务于每一个用户请求的整个流程：

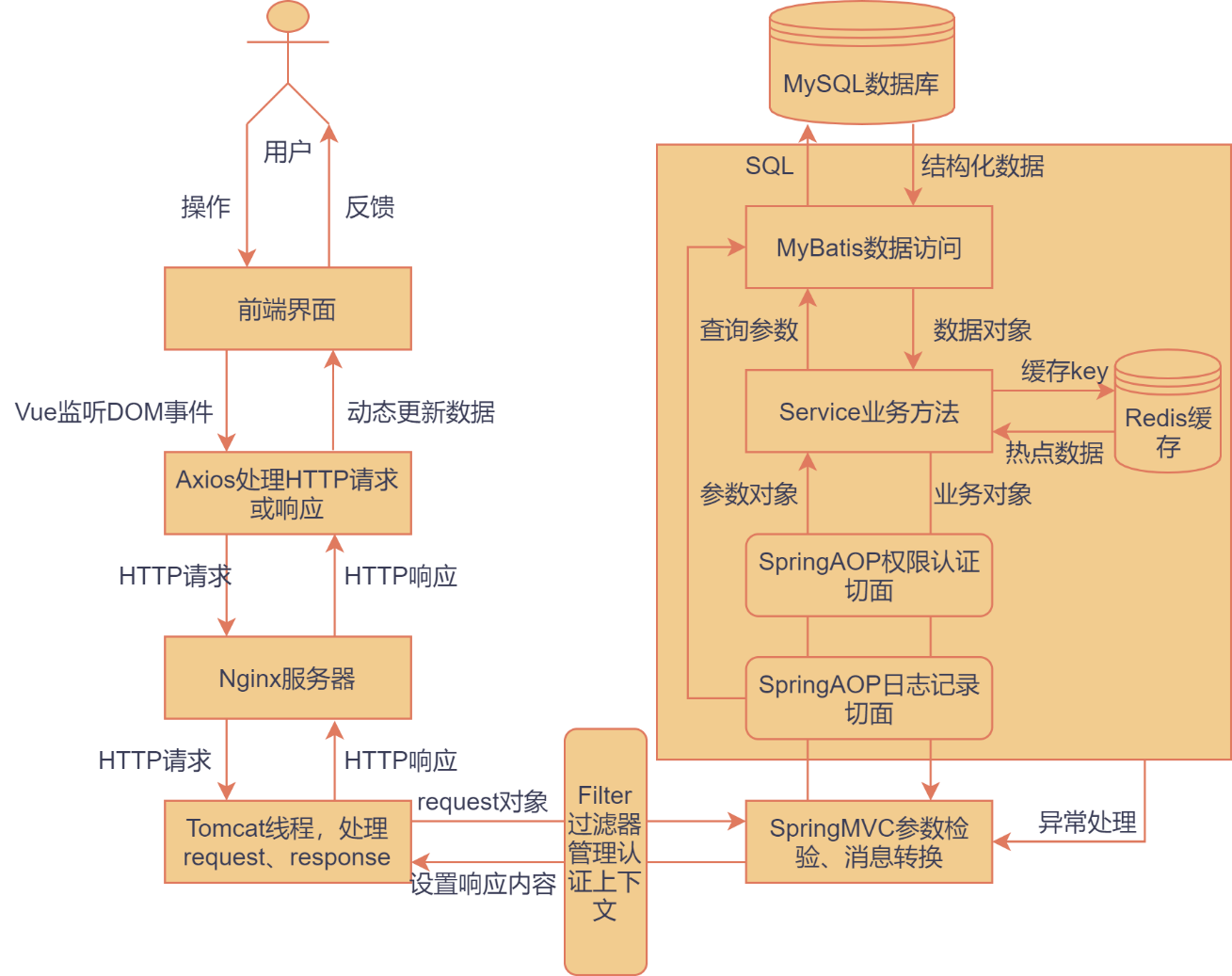


图4‑2系统工作流程

Springboot由传统SSM架构的后台框架发展而来，简化了SSM的配置和开发过程，整合了几乎所有当前业界后台开发的主流技术，极大地减少了开发人员的工作量，实现了从环境搭建、配置、开发、测试和打包的一系列自动化操作。系统与数据库的交互采用MyBatis持久层框架，屏蔽了重复的JDBC操作，支持动态SQL，实现了数据访问与业务逻辑的分离。数据库方面使用关系型数据库MySQL存储业务数据，非关系型数据库Redis作为缓存中间件。系统的日志功能采用SpringAOP实现，对每一个需要记录日志的功能进行统一拦截。权限认证使用SpringSecurity框架完成，SpringSecurity是Spring项目之中的一个安全模块，可以非常方便与项目无缝集成。

使用框架为开发提供了极大的便利，使开发人员只需要专注于业务层和数据库设计，而数据库设计与后台业务逻辑、前后端接口、前端展现都关联紧密。下面将对系统数据库和业务功能设计进行详细阐述。

## 数据库设计

数据库设计使体质健康数据系统设计实现的开始。只有将系统的需求分析详细、将各个实体的关系清晰化、规范化，才能设计出合理的数据库表结构，进而后续开发顺利进行。本系统实际开发过程中，数据库经过多次修改，导致系统前后端代码、SQL语句、接口设计均发生变化，需要耗费大量的时间精力维护，经过多次尝试之后，最终设计了较为合理的表结构。下图4-3为数据库部分ER图，其中实体属性仅画出一部分：

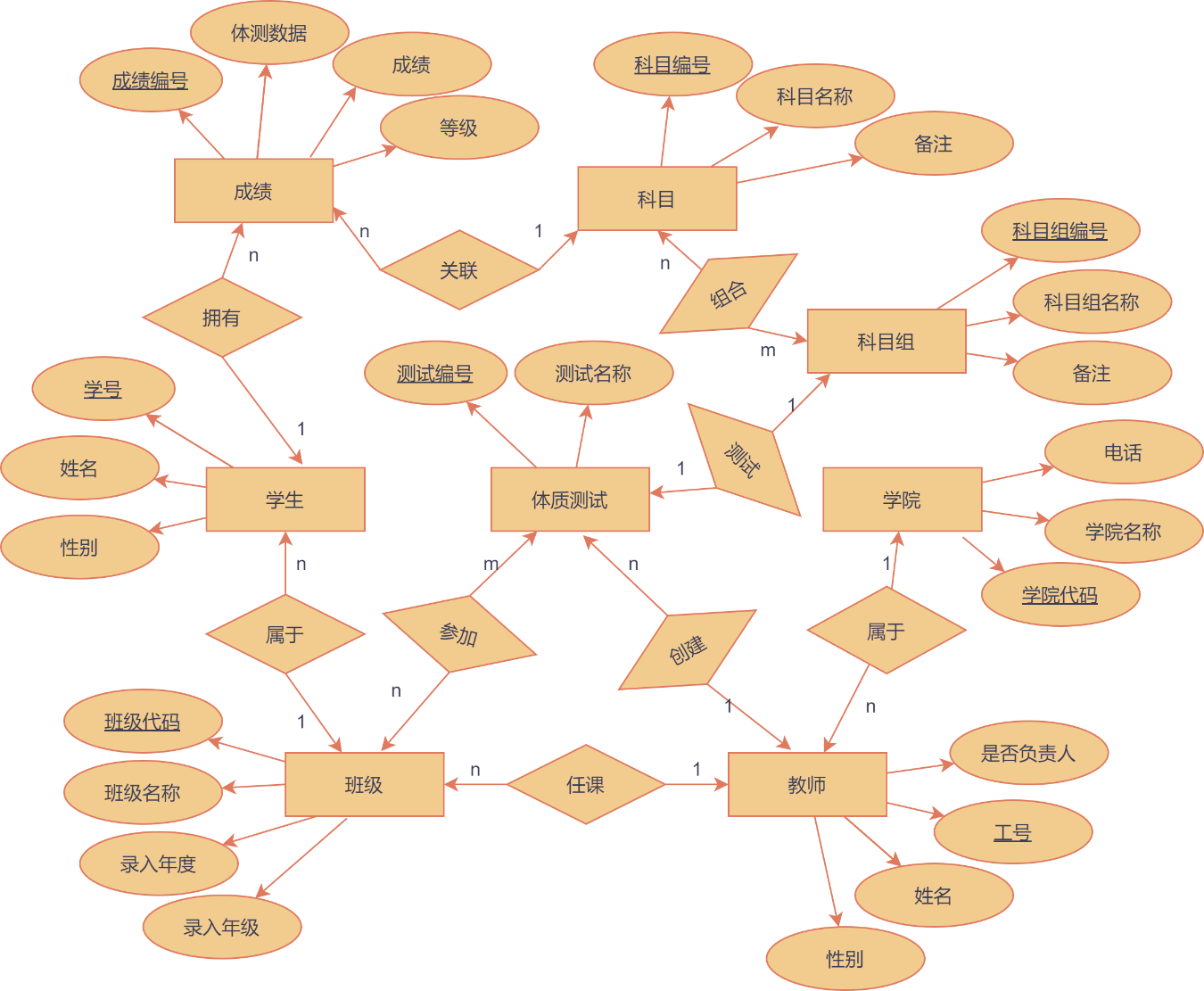


图4‑3数据库部分ER图

设计过程中，遵循以下设计规范：

1. 表名、字段名均使用小写，下划线分隔，尽量设置统一前缀。ORM框架自动将小写下划线分隔单词映射为驼峰命名，字段名统一前缀可方便连表查询。
2. 每个表设计时尽量包含两个时间字段，创建时间和修改时间；
3. 必须要有主键，且主键尽量不包含任何业务含义。主键用以保证实体的完整性，且可以加快数据库的响应速度；
4. 字段尽可能为非空，MySQL要求索引字段不能为空，且为空的字段，判断会变得复杂；
5. 使用InnoDB存储引擎，InnoDB是默认的表存储引擎。其特点是行锁设计、支持MVCC、支持外键、提供一致性非锁定读、同时被设计用来最有效的利用以及使用内存和CPU；
6. 字符集使用utf8mb4，排序规则使用utf8mb4\_bin，utf8mb4相比于utf8支持部分特殊字符；
7. 为表名、字段名加上备注信息，有助于理解数据库设计，生成程序实体类也会自动注释。

系统数据库包含共20张表，如下图所示



图4‑4系统数据库表

下面将对主要数据库表进行详细介绍

1. 学生表，pt\_student

性别字段并未使用枚举，而使用了占存储较小的char(1)类型，数据库密码不存储明文，varchar(255)用以存储加密后的字符串，学生的年级信息关联在班级表中。

表4‑1学生表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 类型 | 描述 | 备注 |
| sid | bigintunsigned | 无含义 | 自增、唯一、不可为空 |
| stu\_id | varchar(64) | 学号 | 主键 |
| stu\_name | varchar(64) | 姓名 | 不可为空、索引 |
| stu\_gender | char(1) | 性别 | M或F、不可为空 |
| stu\_birth | date | 出生日期 | 不可为空 |
| cls\_code | varchar(64) | 班级代码 | 外键 |
| password | varchar(255) | 密码 | 不可为空，有默认值 |
| avatar | varchar(255) | 头像 | 不可为空，有默认值 |
| stu\_created | timestamp | 创建时间 | 不可为空，有默认值 |
| stu\_modified | timestamp | 上次修改 | 可为空 |
| stu\_last\_login | timestamp | 上次登录 | 可为空 |
| stu\_desp | varchar(255) | 备注信息 | 不可为空，有默认值 |

1. 教师表，pt\_teacher

因中小学无学院部门，故学院外键可为空。教师身份可为学院负责人或普通教职工，默认为普通教职工。

表4‑2教师表

| 字段名 | 类型 | 描述 | 备注 |
| --- | --- | --- | --- |
| tid | bigintunsigned | 无含义 | 自增、唯一、不可为空 |
| tea\_id | varchar(64) | 工号 | 主键 |
| tea\_name | varchar(64) | 姓名 | 不可为空、索引 |
| tea\_gender | char(1) | 性别 | M或F、不可为空 |
| tea\_birth | date | 出生日期 | 不可为空 |
| clg\_code | varchar(64) | 学院代码 | 外键、可为空 |
| principal | char(1) | 是否负责人 | 不可为空，默认0 |
| password | varchar(255) | 密码 | 不可为空，有默认值 |
| avatar | varchar(255) | 头像 | 不可为空，有默认值 |
| email | varchar(64) | 邮箱 | 可为空 |
| phone | varchar(64) | 手机 | 可为空 |
| tea\_created | timestamp | 创建时间 | 不可为空，有默认值 |
| tea\_modified | timestamp | 上次修改 | 可为空 |
| tea\_last\_login | timestamp | 上次登录 | 可为空 |
| tea\_desp | varchar(255) | 备注信息 | 不可为空，有默认值 |

1. 班级表，pt\_class

表4-2教师表（续表）

由于中小学无学院这个部门，故学院可为空。考虑到班级的年级信息是随时间变化的，设计了录入年度和录入年级两个字段用于动态计算年级。

表4‑3班级表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 类型 | 描述 | 备注 |
| cid | bigintunsigned | 无含义 | 自增、唯一、不可为空 |
| cls\_code | varchar(64) | 班级代码 | 主键 |
| cls\_name | varchar(64) | 班级名称 | 不可为空、索引 |
| clg\_code | varchar(64) | 学院代码 | 外键、可以为空 |
| cls\_entry\_year | int | 录入年度 | 不可为空 |
| cls\_entry\_grade | int | 录入年级 | 不可为空 |
| tea\_id | varchar(64) | 任课教师工号 | 外键、不可为空 |
| cls\_created | timestamp | 创建时间 | 不可为空，有默认值 |
| cls\_modified | timestamp | 上次修改 | 可为空 |

1. 体质测试表，pt\_measurement

体质测试只能由教师创建，关联一个科目组和多个班级，多对多班级信息存储在中间表中。

表4‑4体质测试表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 类型 | 描述 | 备注 |
| ms\_id | bigintunsigned | 无含义 | 自增主键 |
| ms\_name | varchar(64) | 体测名称 | 唯一、索引、不可为空 |
| ms\_desp | varchar(255) | 备注信息 | 不可为空、有默认值 |
| grp\_id | bigintunsigned | 科目组id | 外键、不可为空 |
| ms\_created\_tea | varchar(64) | 创建教师 | 外键、不可为空 |
| ms\_created | timestamp | 创建时间 | 不可为空，有默认值 |
| ms\_modified | timestamp | 上次修改 | 可为空 |

1. 班级、体测中间表，pt\_class\_measurement

学生以班级为单位参加体质测试，班级与测试之间为多对对关系，故需要一张中间表维护。

表4‑5班级测量中间表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 类型 | 描述 | 备注 |
| cms\_id | bigintunsigned | 无含义 | 自增主键 |
| cls\_code | varchar(64) | 班级代码 | 外键、不可为空 |
| ms\_id | bigintunsigned | 体测编号 | 外键、不可为空 |
| cms\_created | timestamp | 创建时间 | 不可为空，有默认值 |
| cms\_modified | timestamp | 上次修改 | 可为空 |

1. 成绩表，pt\_score

每一条成绩记录由学号、体测编号、科目id唯一确定，同一次体测中有多个科目，每个科目可多次参与测试。体测成绩、等级导入数据时由系统自动计算生成。

表4‑6成绩表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 类型 | 描述 | 备注 |
| sco\_id | bigintunsigned | 无含义 | 自增、唯一 |
| stu\_id | varchar(64) | 学号 | 外键、不可为空、主键 |
| ms\_id | bigintunsigned | 体测编号 | 外键、不可为空、主键 |
| sub\_id | bigintunsigned | 科目id | 外键、不可为空、主键 |
| sco\_data | decimal(7,3) | 体测数据 | 不可为空、索引 |
| score | int | 体测成绩 | 不可为空、索引 |
| sco\_level | varchar(64) | 等级 | 不可为空、索引 |
| sco\_created | timestamp | 创建时间 | 不可为空，有默认值 |
| sco\_modified | timestamp | 上次修改 | 可为空 |

1. 评分标准表，pt\_score\_sheet

该表维护科目的评分标准，由于体测成绩与科目、年级、性别有关，且评价标准为分段形式，经多次修改后最终设计为下面的结构，每一条记录由年级、性别、科目、上区间、下区间唯一标识，当体测数据上传后，需要从该表分段查询成绩与等级。查询时分段区间为左闭右开。评分标准若无上限，则设置为9999.999；若无下限，则设置为-9999.999，不在该区间的数据则认为是异常数据。

表4‑7评分标准表

| 字段名 | 类型 | 描述 | 备注 |
| --- | --- | --- | --- |
| id | bigintunsigned | 无含义 | 自增、唯一 |
| grade | varchar(64) | 学号 | 外键、不可为空、主键 |
| gender | bigintunsigned | 体测编号 | 外键、不可为空、主键 |
| sub\_id | bigintunsigned | 科目id | 外键、不可为空、主键 |
| upper | decimal(7,3) | 上区间 | 不可为空、主键 |
| lower | decimal(7,3) | 下区间 | 不可为空、主键 |
| score | int | 体测成绩 | 不可为空、索引 |
| level | varchar(64) | 等级 | 不可为空、索引 |
| created | timestamp | 创建时间 | 不可为空，有默认值 |
| modified | timestamp | 上次修改 | 可为空 |

1. 接口操作表，pt\_operation

表4-7 评分标准表（续表）

该表中的每一项对应系统的某一功能，教师可拥有多个角色，每一角色可关联多个功能，通过对教师-角色，角色-接口操作表的修改维护则可以实现动态控制功能的目的。该表在系统开发完成之后即是不可变化的。

表4‑8接口操作表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 类型 | 描述 | 备注 |
| id | bigintunsigned | 无含义 | 自增、唯一 |
| oper\_id | varchar(64) | 操作id | 外键、不可为空、主键 |
| oper\_summary | varchar(64) | 操作概要 | 外键、不可为空、主键 |
| oper\_desp | varchar(255) | 操作描述 | 外键、不可为空、主键 |
| created | timestamp | 创建时间 | 不可为空，有默认值 |

1. 操作日志表，pt\_oper\_log

该表用于记录用户对于系统资源的修改记录，使用户对于系统的操作均可查询，有利于系统的安全和稳定。

表4‑9操作日志表

| 字段名 | 类型 | 描述 | 备注 |
| --- | --- | --- | --- |
| oper\_id | bigintunsigned | 无含义 | 自增主键 |
| oper\_desp | varchar(255) | 操作描述 | 不可为空 |
| oper\_req\_param | text | 请求参数 | 可为空 |
| oper\_exp | text | 异常信息 | 可为空 |
| oper\_user\_type | varchar(10) | 用户类型 | 不可为空 |
| oper\_user\_id | vachar(64) | 用户id | 不可为空 |
| oper\_uri | varchar(255) | 请求uri | 不可为空 |
| oper\_ip | varchar(64) | 请求ip | 不可为空 |
| oper\_time | timestamp | 操作时间 | 不可为空，有默认值 |
| oper\_status | char(1) | 操作状态 | 不可为空、默认为1 |
| oper\_type | varchar(10) | 操作类型 | 不可为空 |

## 系统权限设计

表4-9 操作日志表（续表）

系统用户主要有三类，管理员、教师、学生，三类用户在查看、修改系统资源上有不同权限，大致概括如下：

表 4‑10用户权限概览表

| 用户类型 | 可查看权限 | 可修改权限 | 权限可扩展 |
| --- | --- | --- | --- |
| 管理员 | 个人信息、基础数据、单科目、体测、科目评分标准、成绩、统计分析 | 个人信息、基础数据、教师权限、单科目、评分标准 | 否 |
| 教师 | 个人信息、基础数据、单科目、体测、科目评分标准、成绩、统计分析、科目组 | 个人信息、体测、成绩、科目组 | 是 |
| 学生 | 个人信息、基础数据（部分）、单科目、体测、科目评分标准、个人成绩 | 个人信息 | 否 |

可修改包括一切非幂等操作，即增加、删除、更新。

从表中可以看到，不同用户类型呈现的用户界面、操作菜单、按钮等也应当是不一样的。对于教师而言，权限是可扩展的，所有教师默认有基本权限，基本权限无法更改，管理员可以为某位教师授予其他权限。

按照上述分析，系统的权限要求总结如下：

1. 对于不同用户，系统应呈现不同的界面、可操作菜单等，属于界面权限或模块权限。
2. 对于不同用户，系统应对界面的可操作按钮进行控制，接口权限。
3. 对于同种类型的不同用户，系统需要过滤不属于该用户的数据，例如教师只能选择自己任课的班级参与体测、教师只能查看但无法修改他人创建的科目组等，属于数据权限。
4. 对于教师用户，需要动态扩展权限。
5. 所有的权限除了在前端控制用户无法正常之外，都需要对后台接口进行权限认证，无权限的请求无法进入到业务，因为所有的前端权限控制主要在于用户体验，真正的权限控制需要后端来保证。下面将分别介绍上述要求的具体设计。

### 界面权限设计

对于多数系统而言，管理员与普通用户应当有两套不同的用户界面，因为多数情况管理员和普通用户的功能存在一定的差异。但在体质健康数据管理系统中，管理员、教师、学生对于数据的访问权限差异较小，主要在于数据的修改权限，同时考虑到系统的复杂度、个人完成时间等因素，在界面权限上并未针对不同用户编写不同的用户界面，而是对决定对菜单做主要的控制。目前主要的解决方案有以下两种：

1. 前端模式。在前端固定写好路由的权限，指定路由有哪些权限可以查看。只初始化通用的路由，需要权限才能访问的路由没有被加入路由表内。在登录获取用户权限后，通过权限去遍历路由表，获取该角色可以访问的路由表，生成路由表。实现页面权限的过滤。该种方式实现简单，但是权限相对不自由，如果后台改动角色，前台也需要跟着改动。适合角色较固定的系统。
2. 后端模式。是通过接口动态生成路由表，且遵循一定的数据结构返回。前端根据需要处理该数据为可识别的结构，动态配置路由表，实现界面权限。该方式配置灵活，但是后端实现较复杂，适合角色变化较多的系统。

本系统中只有教师的权限存在变化，通过上述需求分析发现，教师权限的变化主要在于接口权限，而不存在界面或菜单的变化，即根据用户类型即可判断用户可操作的菜单，属于角色较为固定情况。

综合上述分析，采用前端控制的模式实现界面权限，系统对于界面权限的控制流程如下：

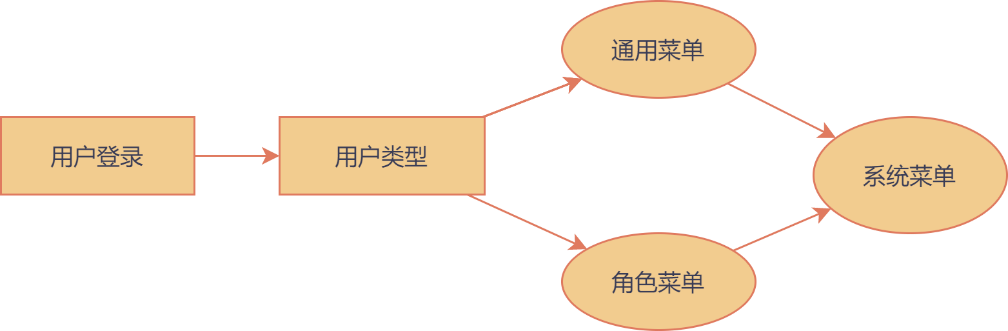


图4‑5界面权限流程

### 接口权限设计

接口权限的控制主要在于后端，前端通过动态渲染菜单、按钮保证了用户正常使用下不会访问无操作权限的接口，但是如果存在不正常使用、恶意攻击则可能访问到无权限的接口，造成系统数据安全问题。所以，后端必须对系统接口进行权限校验。

系统最初使用了SpringSecurity框架能够比较方便地进行接口权限过滤，但获得方便的同时也丢失了灵活性，且SpringSecurity框架较复杂，系统志使用到了极少部分功能，最终选择移除了框架的使用，自己参照框架的设计思想和模式完成了系统的权限校验。系统接口权限的控制流程如下：

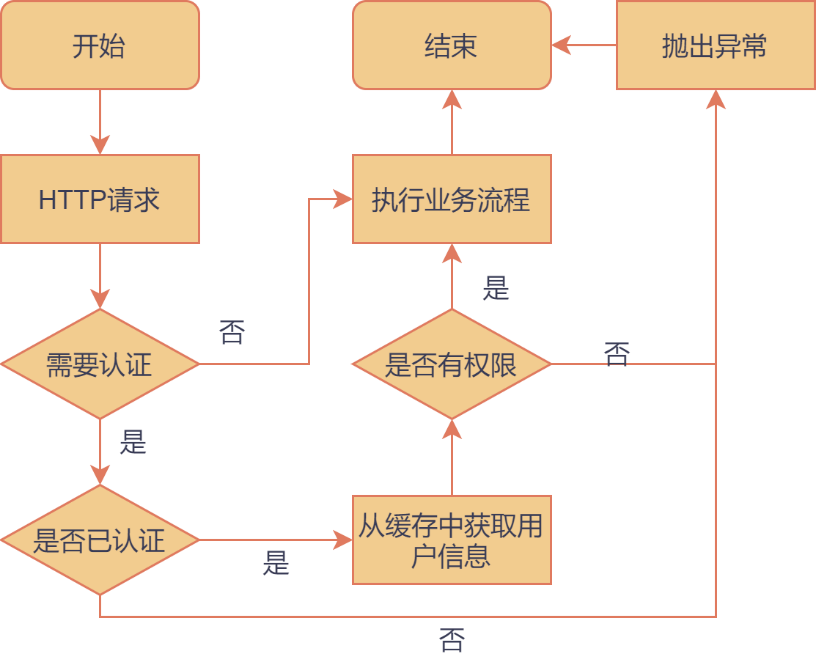


图4‑6接口权限流程

其中，判断是否认证在过滤器中处理，判断是否有接口权限在控制层处理，如此设计保证了在业务流程中只专注于业务，保证了业务与校验的解耦。

### 数据权限设计

数据权限主要控制不同的资源主体有查看不同的数据信息的权限。由于数据权限和系统的业务逻辑关系非常密切，所以不同的系统设计差异性会非常大。从另一方面来说，由于数据权限和业务逻辑关联性非常强，如果系统的业务逻辑非常复杂，数据权限设计起来也会相对复杂，所以关于数据权限的设计一直没有一种相对通用和使用简单的设计方案。

考虑到本系统角色较为固定，数据权限控制点并不多且基本变化很小，例如教师只能选择任课班级参与体测、学生只能查看个人成绩等，综合考虑之后，决定把数据权限的规则限制死在程序中。例如当学院负责人获取班级时，属于上级查看下级的情况，是固定不变的，可直接写死在程序中。

### RBAC模型

教师权限动态扩展的设计基于RBAC（Role-BasedAccessControl）模型，基于角色的访问控制，就是用户通过角色与权限进行关联。简单地说，一个用户拥有若干角色，每一个角色拥有若干权限。这样，就构造成“用户-角色-权限”的授权模型。在这种模型中，用户与角色之间，角色与权限之间，一般是多对多的关系若直接关联用户与角色，当用户数量较大且多用户拥有相同的权限，那么用户-权限中间表的数据将会大量重复，出现“权限爆炸”的情况，而引入角色做中间关联比较好的解决了Who可以对What进行How的操作，下图4-8展示了RBAC模型的基础组成元素及其之间的关系：



图4‑7RBAC模型

本系统中，权限对应系统指定接口的访问权限，权限之间设置联动关系，例如可删除某项资源必须先具有查看资源的权限。教师默认拥有保证正常工作的部分权限。管理员可对系统角色进行管理，选择角色关联某些权限，再通过关联角色和教师为教师授予权限，也就是说教师权限为系统内置权限与教师角色权限的并集。

## 前后端接口设计

项目前后端分离，使用Json数据交互，前端使用Axios发送异步请求，后端SpringMVC接受处理请求。前后端接口设计关系到系统功能的实现与扩展性，不合理的接口设计会无形之中会增加前后端对接调试和代码编写的复杂度，结构清晰、符合标准、易于理解、扩展方便的接口风格可以使系统更简洁，更有层次，更易于实现缓存等机制。

REST(Representational State Transfer)ful[26]架构是目前最流行的一种互联网软件架构。它结构清晰、符合标准、易于理解、扩展方便，所以正在得到越来越多的以网络为基础的软件开发的采用。RESTful架构将每个URI当成一种资源，客户端和服务器之间传递资源的信息实体，是资源的表现层，客户端通过某种动作，对服务器端资源进行操作，改变服务器的状态，实现“表现层状态转换“，即REST名字的解释。这里的动作通常表现为HTTP协议，GET用来获取资源，POST用来新建资源（也可以用于更新资源），PUT用来更新资源，DELETE用来删除资源。

本系统接口设计遵循RESTful风格，下图4-9为RESTful Api架构：

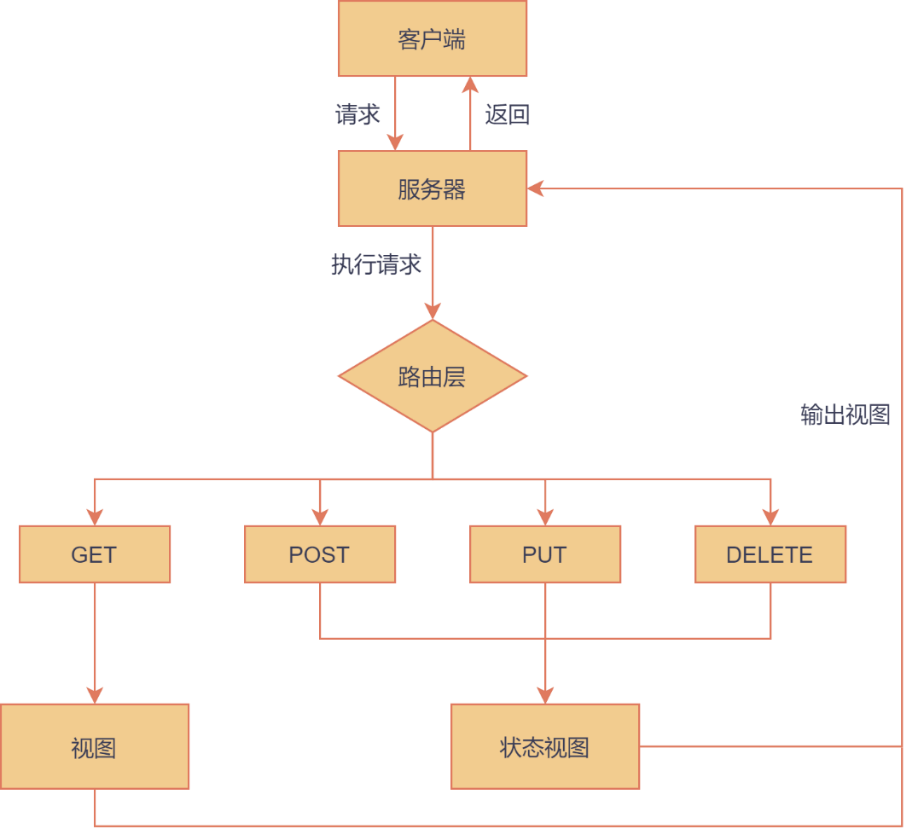


图4‑8RESTful Api架构

HTTP请求的方式即代表了该次操作的语义，所有的GET请求表示从服务器获取资源，都不作任何修改，即是幂等性的，而POST、PUT、DELETE请求对服务器的资源进行了修改，非幂等操作。下表显示了系统的接口设计规范：

表4‑11系统接口风格设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 语义 | 请求方法 | 路径变量 | 是否携带请求参数 | 是否携带请求体 |
| 查询 | GET | 主键 | 是 | 否 |
| 添加 | POST | 无 | 否 | 是 |
| 更新 | PUT | 无 | 否 | 是 |
| 删除 | DELETE | 主键 | 否 | 否 |

1. 查询数据采用GET请求，使用路径变量形式，可以携带请求参数，无请求体。
2. 添加数据采用POST请求，所有参数放在请求体中，请求参数为空。
3. 删除数据采用DELETE请求，采用路径变量形式，请求参数请求体为空。
4. 修改数据采用PUT请求，所有参数在请求体中，请求参数请求体为空。

除此之外，考虑到系统权限校验以及消息友好提示，规定以/api开头的uri都需要权限认证，对于所有请求HTTP状态码统一返回200，操作状态由业务状态码指示。因此，需要定义统一返回体，包括响应代码、响应信息、数据。下表4-12展示了不同的业务状态码表示的含义。

表4‑12业务状态码含义

|  |  |
| --- | --- |
| 状态码 | 含义 |
| 2xx | 本次操作正常，是否成功由消息中的数据判断 |
| 3xx | 客户端参数异常，抵御恶意攻击 |
| 4xx | 未认证或权限不足 |
| 5xx | 服务器异常 |

## 系统功能设计

体质健康数据管理系统主要包含用户与权限模块、基础数据模块、科目模块、体质测试模块、统计分析模块，各个模块下面有若干子模块。各子模块的设计符合统一的规范，概括如下：

1. Controller为控制层，Service为业务层，Mapper为数据访问层。
2. 对于特定的一类系统数据，有且仅有一个Controller类接受用户请求，封装请求参数，并完成数据校验。
3. Controller有且仅有一个负责业务的Service类，调用业务类的指定方法完成请求。
4. Service类有一个或多个数据访问Mapper，Controller层不与Mapper关联。
5. 所有Service类中方法的运行时异常全部不捕获，向上抛出，统一处理。

通过上述设计规范，降低了系统实现的难度，使得代码利于编写、维护，后续的类图设计也很好的体现了上述几点。

下面将对各个模块的详细设计进行阐述。

### 用户与权限模块设计

用户与权限模块主要包含身份认证、个人信息、角色和授权功能等。其中最重要的是身份认证功能包括用户认证和鉴权。系统参照SpringSecurity框架的设计理念，实现了一套自己的认证与授权逻辑。

传统单体应用使用Session来识别用户身份，但是扩展性不好，单机情况下没有什么问题，但是前后端分离，微服务架构下无状态导致传统的Session机制不适用。解决办法是将用户认证信息保留在客户端服务端无需保持任何状态，每次通过客户端信息识别用户身份，基于JWT的登录认证是比较主流的一种实现方式。下图展示了JWT认证的流程：

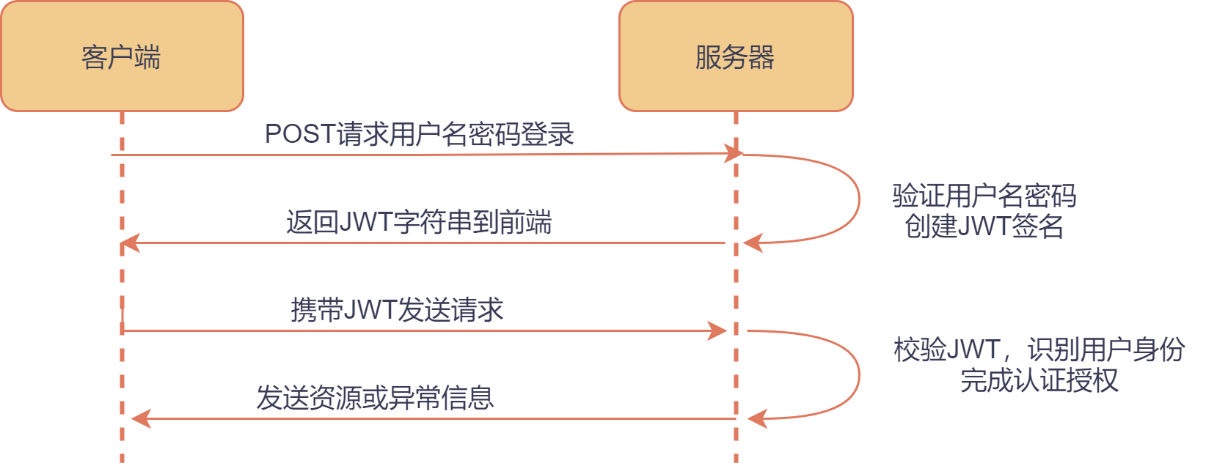


图4‑9基于JWT的认证流程

登录过程中，系统包含三类用户，不同用户的登录逻辑存在差异，故抽象了用户类与登录的统一接口，接口返回抽象用户信息，对于不同类型的用户，执行不同的登录逻辑并返回实际的教师信息、管理员信息或学生信息。下图是登录部分的类图设计，类的get和set方法以及与该模块无关的方法未画出：

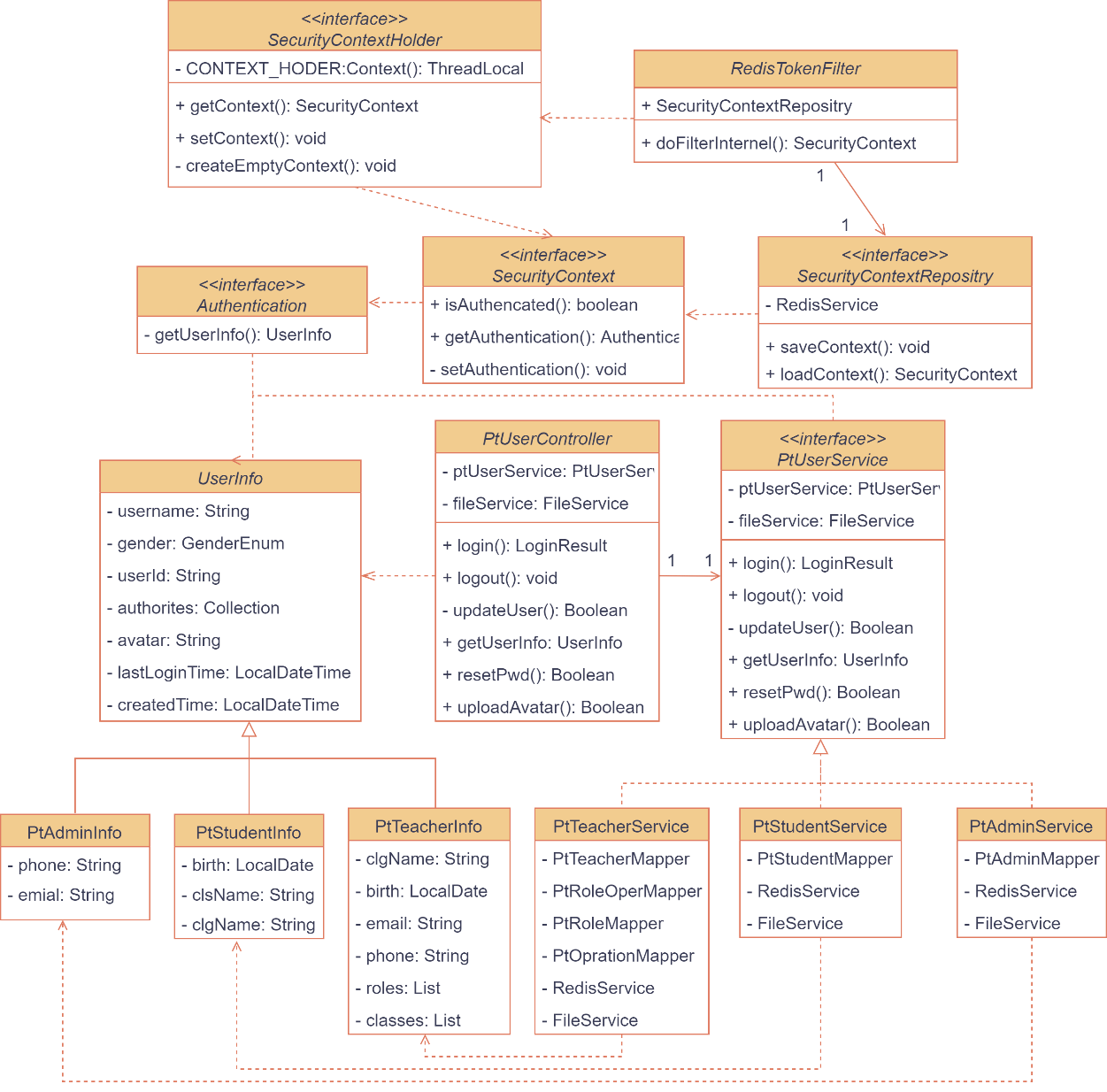


图4‑10登录认证类图

对于每一个登录请求，首先经过RedisTokenFilter，由于用户未认证，通过SecurityContextRepositry加载空的安全上下文并设置到本地线程中，供该次请求使用，然后到达UserController进行登录表单校验，如必须携带的参数、用户名密码不能为空等，校验失败则直接返回错误信息，否则识别当前用户类型调用指定的Service执行登录业务。

以教师登录为例，首先根据用户名查询教师信息，其次校验密码，发生错误则返回教师不存在或密码错误。否则继续查询教师角色信息并与系统内置权限合并作为该教师的所有权限。对于教师而言，还需要查询其所任课的班级。至此，登录基本完成，将用户信息存入到缓存中，生成JWT字符串返回，再次经过RedisTokenFilter会清空安全上下文，表示本次请求完成。此时，缓存在已存在用户信息，当用户携带JWT字符串访问时，从中提取信息并获取缓存，则成功识别了用户身份。

下图为教师登录成功的时序图（不包含校验失败等异常情况）：

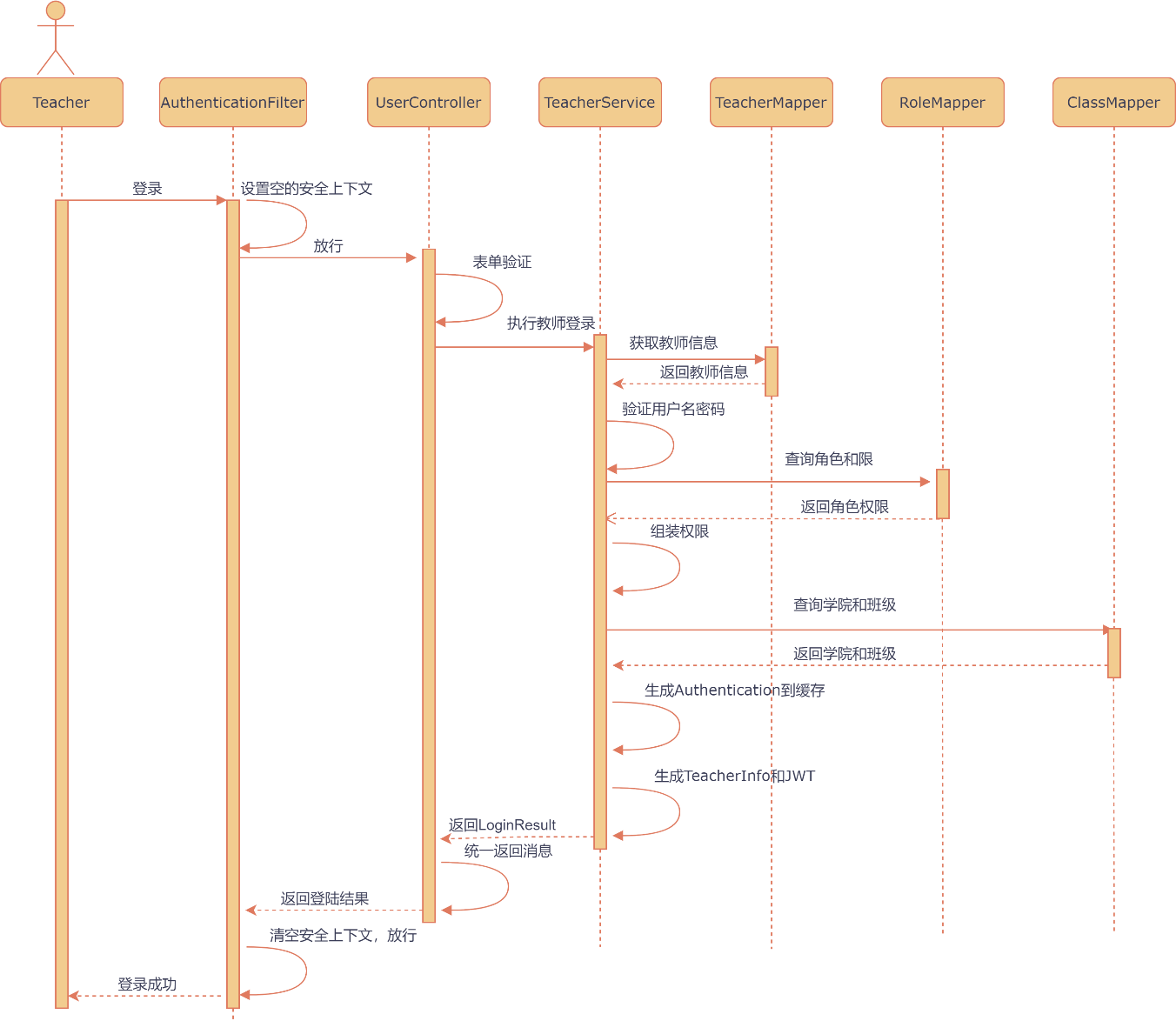


图4‑11教师登录时序图

教师登录是三类用户登录过程最复杂的，其中关联的学院信息、班级信息以及教师角色信息是管理员和学生登录不具有的，所以时序图中需要涉及到学院、班级等的数据访问，而管理员和学生的登录流程则相对简单，管理员仅需校验用户名密码，学生还要查询班级和学员信息，三类用户登录成功后返回统一的用户信息。下图4-12展现了系统各类用户登录流程。

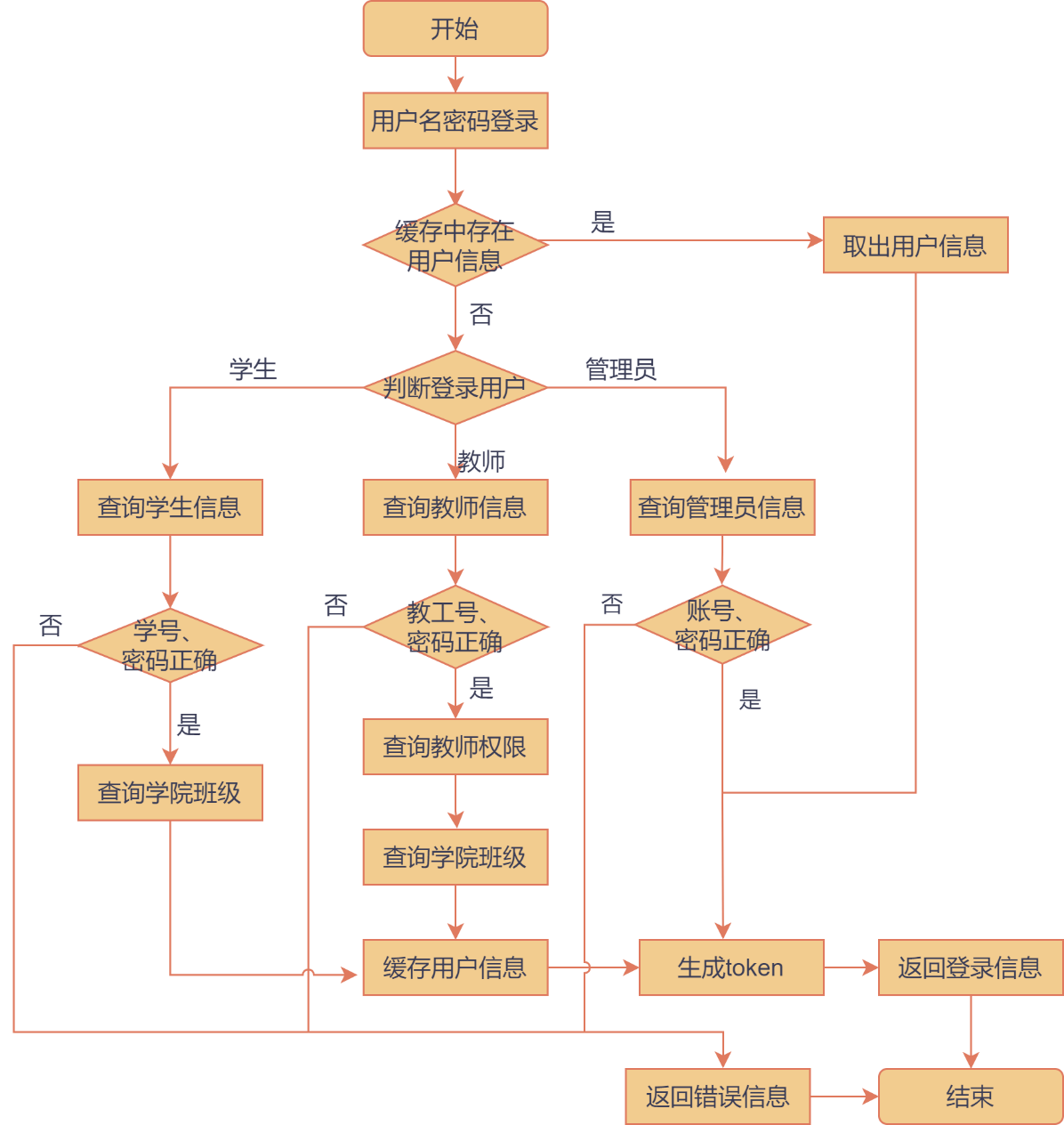


图 4‑12系统各类用户登录流程

### 基础数据模块设计

基础数据可以由学生、教师、管理员查看，但其修改、删除操作只有管理员有权限。管理员可以进行学生管理、教师管理、班级管理、学院管理以及各类数据的文件导入导出等功能。四个子模块的设计实现大同小异，都包括增删改、分页查看、文件模板和数据上传方法。

以当中的教师模块为例，类图设计如下所示：



图4‑13基础数据教师模块类图

所有的方法命名规范，有见名知义的效果。其中PtTeacherExcel类和PtTeacherExcelListener两个类负责完成Excel数据的解析存储。本系统所有的Excel文件解析由阿里巴巴开源包EasyExcel完成，EasyExcel是阿里巴巴开源的一个Excel处理框架，以使用简单、节省内存著称，其解析数据时并未一次性加载到内存，而是通过一行一行从磁盘读取，并将一行的解析结果以观察者的模式通知ExcelListener。当需要解析的字段固定时，如导入教师数据，可以通过定义类来快速实现，在类中定义并存在注解的字段会被框架自动解析成对象，使用简单、方便。

但是对于来自网络的数据，服务器不可能完全信任地处理到数据库中，必须经过严格的校验，但文件流在控制层校验测不符合程序的规范性，且消耗系统性能，只能在数据一行行解析时校验。除了校验内容是否安全外，对于系统数据的完整性、唯一性等约束同样需要校验，事实上，直接将数据插入到数据库也会被数据库拒绝，但是这样做存在两个弊端：

1. 消耗资源和增加数据库压力，数据校验的工作应当在程序中完成，数据库仅负责存储数据。
2. 不能提供有好的用户提示信息，对于数据库的报错，程序难以从中提取出用户提示信息，例如某行某列某单元格发生了某某错误，只能笼统地告知用户文件内容有误，用户体验较差。

本系统设计时对于所有的字段校验都在程序中完成，一旦校验不通过，则抛出异常，提示详细的信息到前端用户。如上图所示，Listener类中的validtate方法完成校验工作，同样以教师数据为例，需要完成校验工作：

1. 教工号不能为空且只能包含数字和字母
2. 教工号不能重复
3. 姓名、性别、出生日期不能为空
4. 学院不能为空且必须存在于数据库中，学院负责人唯一。
5. 所有的字符串数据有长度限制

一旦上述校验地任何一条不通过，则本次数据上传直接失败，否则将其添加到缓冲区。当缓冲区数据达到指定数量后，插入到数据库中，然后继续解析文件流。

下图4-14展示了教师数据上传的时序关系：

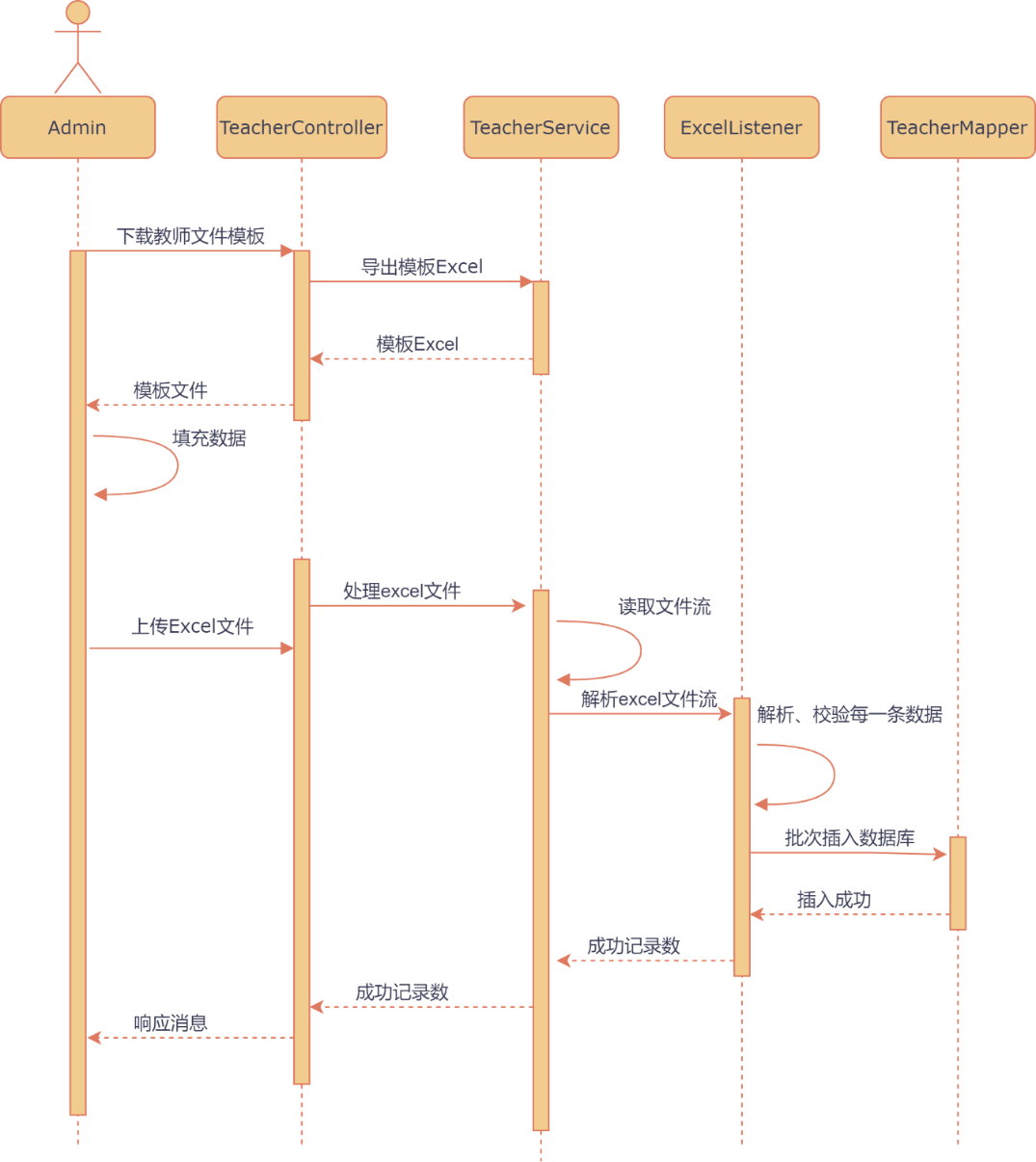


图4‑14教师Excel数据上传时序图

### 科目模块设计

科目模块主要包含单科目管理、科目组管理、科目评分标准管理三个子模块。每个模块包含分页查看，增删改功能，其中对于每个科目的评分标准提供了在线修改和文件上传的功能。在科目组管理中，教师可以查看自己创建的其他教师分享的科目。管理员仅有查看权限，学生无权限查看。教师，学生和教师只能查看科目和科目的评分标准，而管理员可对二者进行添加、修改、删除等操作。下图为科目模块的类图设计：



图4‑15科目模块类图

科目模块完全为体质测试服务，其个性化配置的设计是系统中的难点，故独立设计成一个模块。体质测试关联科目组，科目组关联多个科目，每个科目关联测试对象（某年级某性别学生），每个测试对象有不同的评分标准，故将四者分开设计，模块对外呈现为科目组。这样支持对每一个部分进行定制化配置。

该模块对于系统用户来说只有两个部分，科目以及科目组，评分标准、测试对象和分享人都是作为属性存在，所以修改科目组包含了修改基本信息、包含的科目、分享人，修改科目则包含了修改基本信息、测试对象、评分标准，而删除科目或者科目组则要联动删除其关联信息。每一个涉及到多表更新的业务都需要开启数据库事务以保证数据的完整性。下图展示了修改科目信息的流程图：



图4‑16更新科目信息流程图

对于成绩评价标准的excel文件上传，要经过严格的数据验证：

1. 性别、年级、分数、等级不能为空且有长度限制
2. 若无上限，默认上限为9999.999；若无下限，默认下限为-9999.999，上限严格小于下限，上下限区间不能有交叉部分。
3. 该年级该性别的学生必须为该科目的测试对象。

### 体质测试模块设计

体质测试模块为系统的核心功能，包含体质测试的管理、成绩信息两个子模块。体质测试管理包含增删改，教师查询体测信息与学生查询体测信息，二者的数据权限不同，查询的结果也不一样，但都只能看到与自己相关的体测信息。除此之外，还可下载某次体测的成绩模板，包含了所有学生及其需要测试的科目，只需填充数据即可上传到系统中并生成成绩。成绩模块只能添加与查看，无法对成绩数据进行修改和删除。该模块类图设计如图4-16所示。



图 4‑17 体质测试模块类图

从图中可以看到，体质测试业务几乎需要关联其他所有的数据，体质测试创建人为教师，参与单位为班级，参与个体对象为学生，测试项目为科目组，每条成绩记录为学生所有，每次体测每个科目可有多条成绩。所以该部分的业务实现起来是最复杂的，需要关联查询系统中绝大部分表并处理成符合要求的数据格式。

以体测数据导入为例，因为每次体测需要测试的科目是不确定的，科目是否需要评分也是不确定的，所以无法像上面那样通过定义类的方法快速解析Excel，除此之外，在同一次体测中，不同年级、不同性别的学生需要测试的科目也不尽相同，相同科目对于不同学生的评分标准也不尽相同，这些都需要系统自动判断，增加了业务的复杂度。除此之外，由于每一科目的数据都需要查询评分标准，考虑到性能和数据库压力，需要对使用到的评分标准做缓存处理，这些在类图的PtScoreExcelListener中都有所体现。

综和以上考虑，该业务需要进行的步骤较多，流程图如下，左图为导入某次体测的整体流程，右图为解析每一项体测数据的流程：

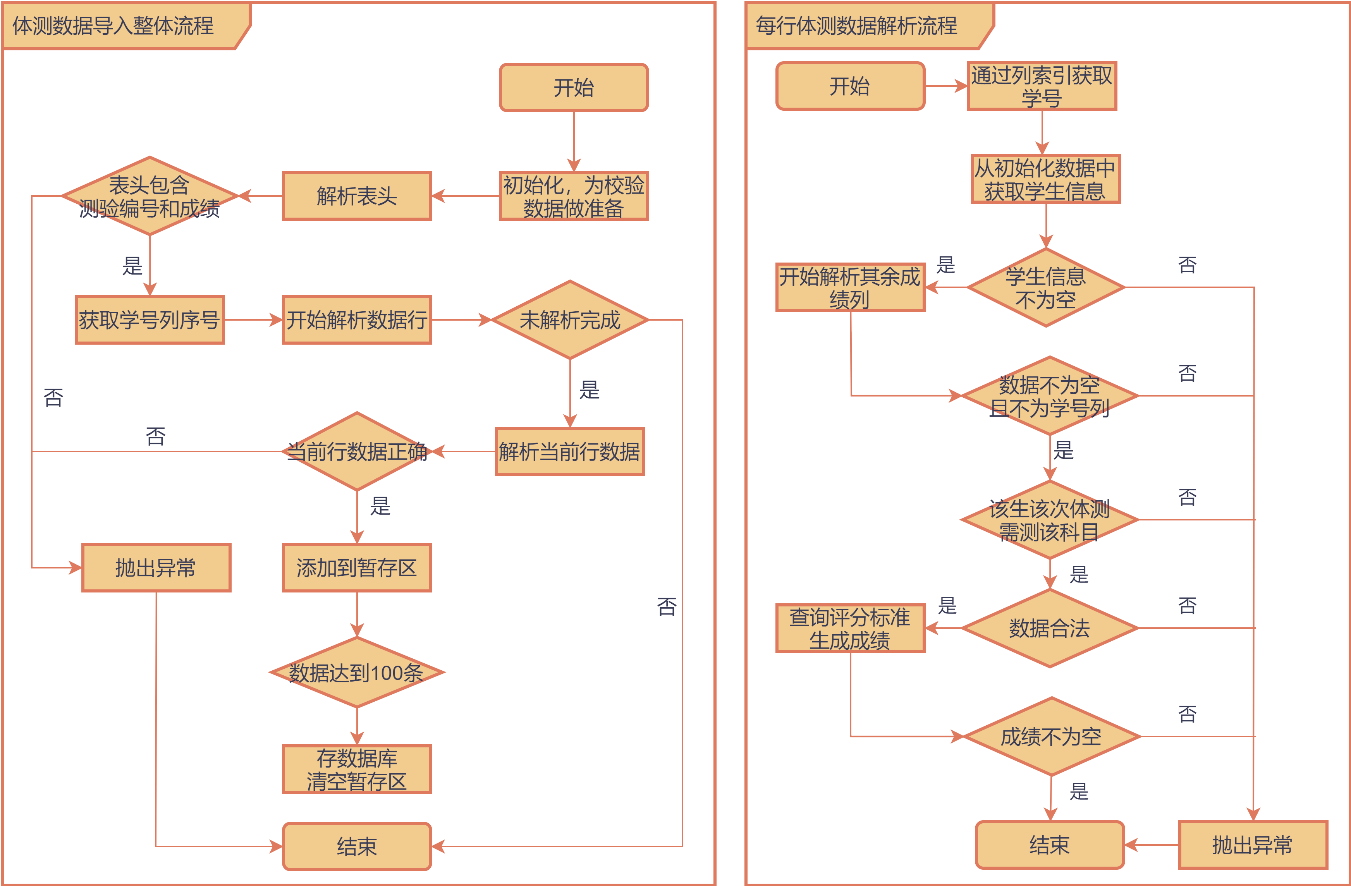


图4‑18体测数据导入流程

### 统计分析模块设计

该模块仅涉及数据的查询，没有任何修改操作。对于数据统计部分，实现较为简单。后端只需要设计多条件查询的接口，将数据返回前端即可，后端不做任何数据处理操作。前端接收到数据，根据各种图表需要，处理成需要的格式由EChart完成渲染。数据分析模块本研究做了需求设想，但并未实现。

## 本章总结

本章主要针对第三章的需求分析进行系统设计，阐述了系统的整体设计思路，然后对于系统的数据库设计、权限设计和接口设计进行了详细描述，接着分模块针对每个需求点对系统的功能以类图、时序图、流程图等进行了详细的设计。

# 体质健康系统实现与测试

经过前四章对于体质健康数据系统的设计分析，本章将对系统主要功能的实现进行介绍，对于体质健康数据系统实现而言，就是将登录与权限、基础数据、科目与体测等各功能得到体现，供用户进行操作。对于体质健康系统测试，除了需要测试功能是否能满足用户需求外，用户的使用体验、界面的美观程度也是需要考虑的方面。

## 系统开发环境搭建

系统开发过程中采用了多种技术框架，其名称、所用版本和说明信息如下表5-1所示。

表5‑1系统开发所用技术框架

|  |  |  |
| --- | --- | --- |
| 名称 | 版本 | 说明 |
| Java | 11 | 后台开发语言 |
| MySQL | 8.0.19 | 数据库 |
| SpringBoot | 2.6.2 | 容器+MVC框架 |
| EasyExcel | 3.0.5 | Excel操作工具 |
| Redis | 5.0.10 | 缓存中间件 |
| TypeScript | 4.6.3 | 前端开发语言 |
| Vue | 3.2.21 | 前端框架 |
| Antdv | 2.2.8 | 前端组件库 |
| IntellijIDEA | 2021.2.3 | 后台开发IDE |
| VsCode | 1.66.2 | 前端开发IDE |
| Python | 3.8 | 模拟测试数据 |
| Git | 2.34.0.windows.1 | 版本控制 |

由于本系统测试需要较多的测试数据，使用Python脚本生成，本章实现和测试部分所展示的数据均为模拟数据，不包含任何真实信息。

## 系统主要功能实现

### 用户与权限功能实现

用户登录界面如下所示，可以选择登入用户类型，用户不存在、密码错误都会有相应的提示，登录成功自动跳转到主页。



图5‑1系统登录页面

该模块还包括角色和权限功能的控制，对于教师可以扩展其权限，系统管理员登录后可查看系统所有接口功能，创建角色并分配功能。下图5-2展示了系统的权限树，使用Echarts绘制生成。

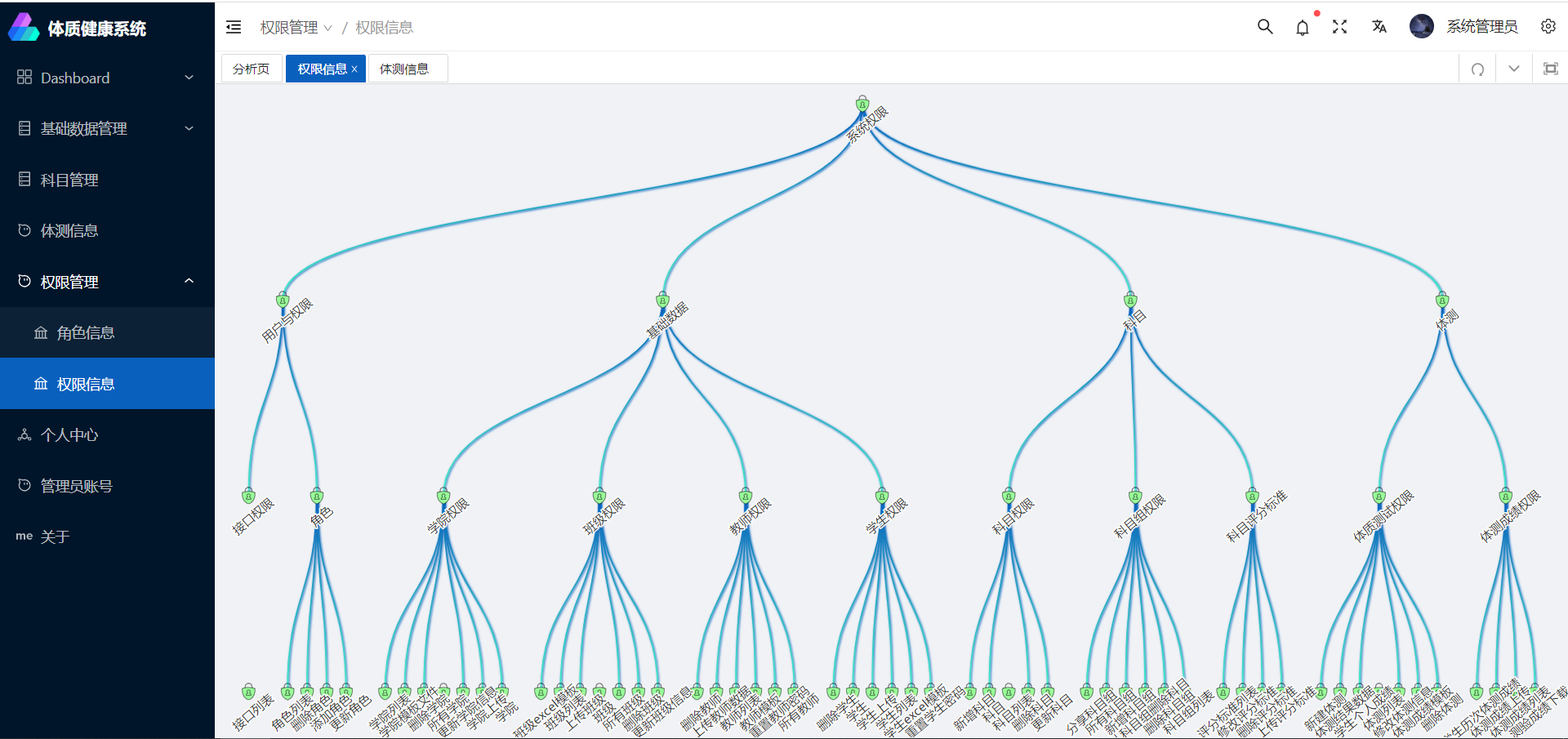


图5‑2系统权限树

### 基础数据功能实现

基础数据模块界面如下图5-2所示，当前登入用户为系统管理员，在菜单栏基础数据管理下有学院、班级、教师和学生信息管理，当前页面为教师列表，其余三个模块的页面布局大致相同。

可以看到，页面上方为搜索表单，用户可以根据学院、姓名、工号多条件组合搜索教师信息。教师列表的右上方为工具栏，从左到右的功能分别为Excel文件模板、教师数据上传、刷新表格、调整表格密度、表格列配置、全屏功能。

表格最后一列为操作列，会根据当前登录用户权限动态调整，由于当前登入用户为管理员，故拥有基础数据模块的所有权限，从左到右依次为删除教师信息、为教师分配角色、修改教师信息、重置密码和教师详情。



图5‑3基础数据教师列表页

### 科目功能实现

科目模块包含科目与科目组两个菜单，但只有教师拥有科目组模块的权限。

下图5-3为科目列表页，页面顶部左侧为标题与搜索框，可以根据名称搜索科目信息，右侧为添加科目按钮。所有科目信息以卡片列表的形式分页查看，显示了包括科目名、运动能力、备注信息、测试对象、创建时间等信息。卡片左下方为操作栏，从左到右依次为查看科目评分标准，编辑科目信息，删除科目、导入评分标准。

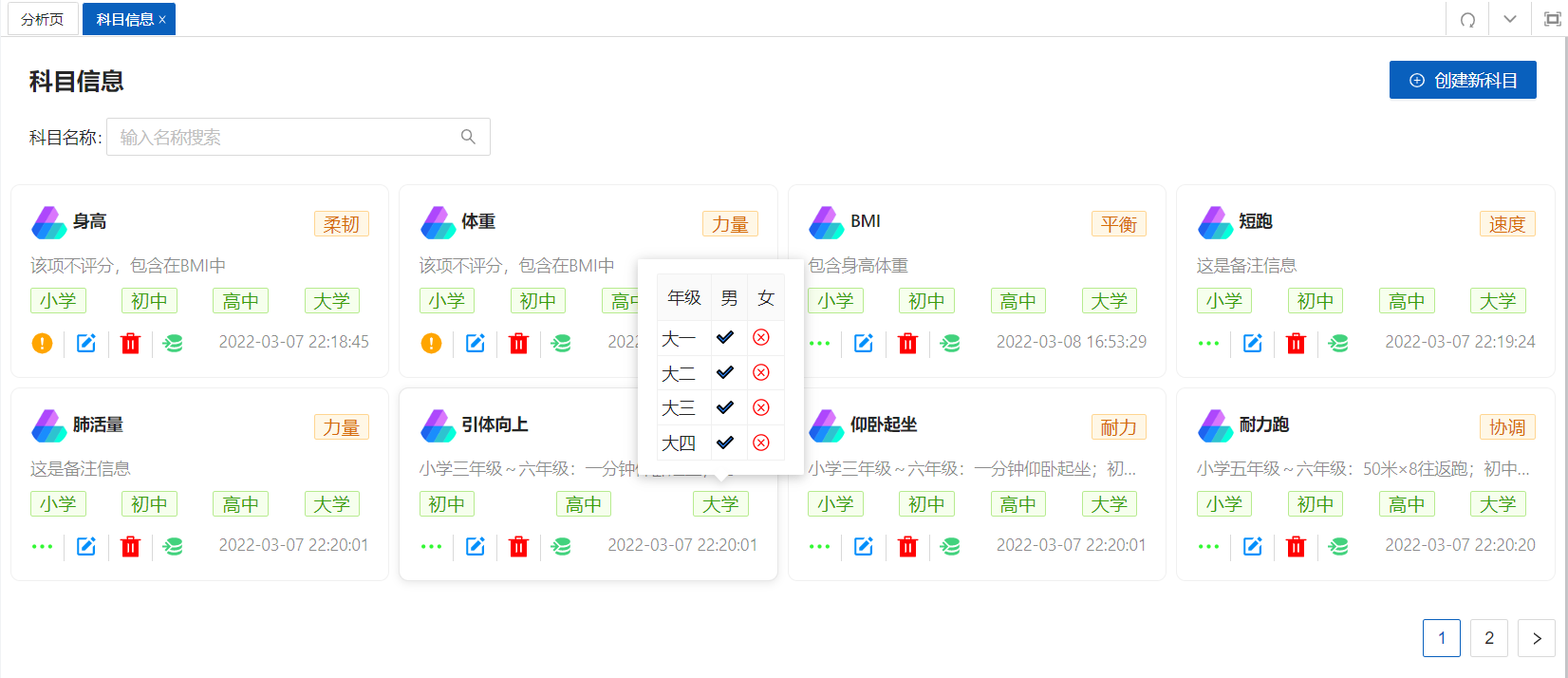


图5‑4科目列表页

由于当前系统登录用户为管理员，所以上述所有功能都具有权限，都会在界面中显示。执行科目的增删改功能都会弹出相应的操作对话框，有较好的交互体验。下图5-4为更新科目信息的对话框。



图5‑5更新科目信息

对于用户的输入前端后端都做了严格的数据校验，修改科目信息包括名称、运动能力、测试对象等。

对于评分标准既支持从Excel文件导入，也可以在科目的评分标准界面在线修改。下图5-5为指定科目的评分标准界面，上方搜索框可搜索指定的评分标准，教师和学生默认只能对该页面的数据进行查看。点击右侧操作列的编辑或删除按钮即可修改数据，除了等级未选择框外，其余三项为数字输入框，当数据不符合规范时，会拒绝用户提交，当某一行处于编辑状态的的时候，当前页面的其他功能是禁用的，页面设计中考虑到了许多细节。操作列同样做了权限的控制。



图5‑6评分标准界面

### 科目组功能实现

切换教师账号登录查看科目组信息，下图5-6展示了科目组列表页。

页面上方包含搜索区域和新增按钮，科目组卡片左下方包含了详情、编辑、删除和分享操作。其中对于被分享的科目，当前教师无修改权限，仅可以查看和在体测中使用该科目组。



图5‑7科目组列表页

### 体质测试模块实现

当系统中录入学生、班级、教师、科目等数据后，教师才可以创建体质测试。下图5-8为体质测试列表页。

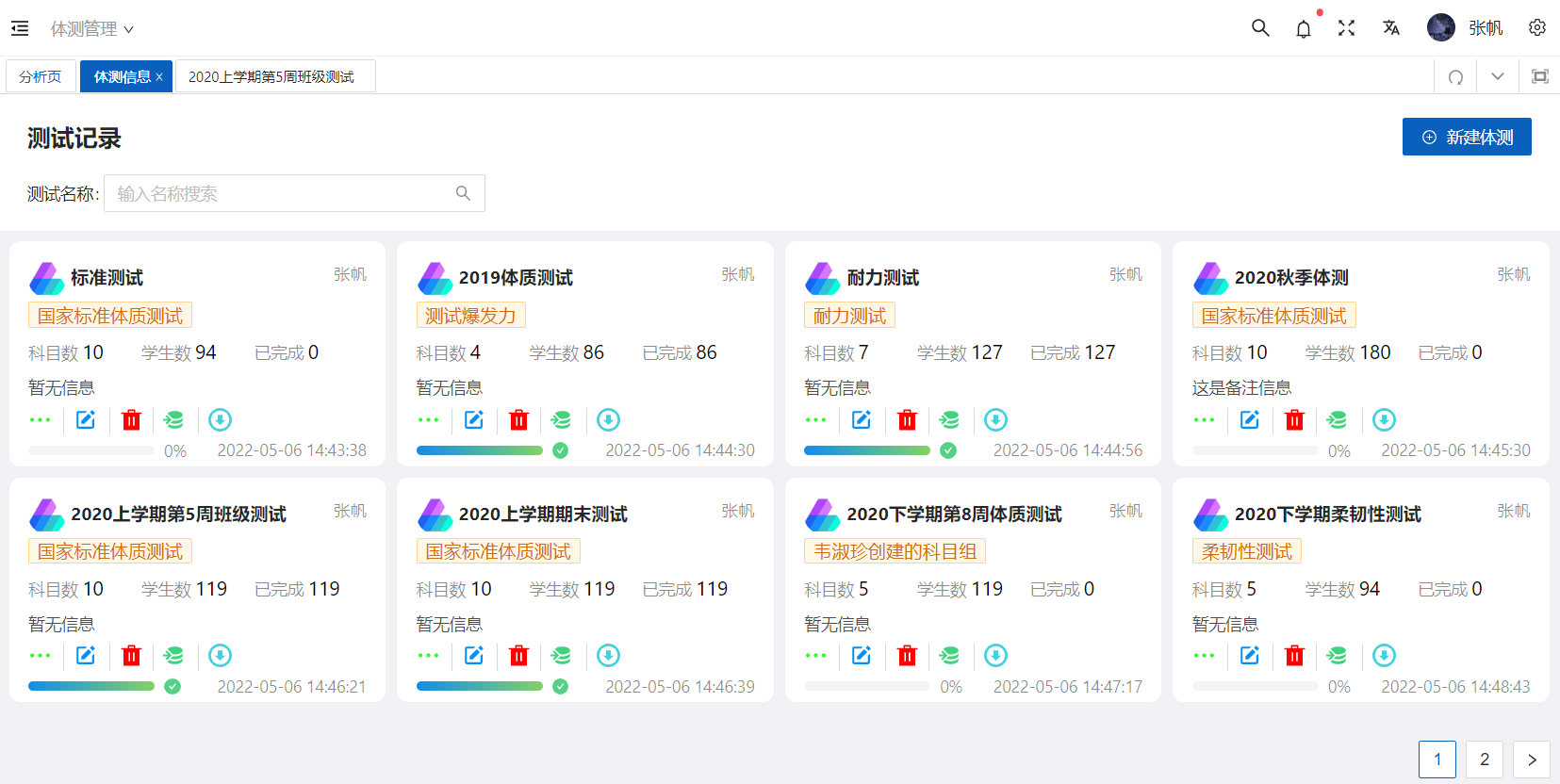


图5‑8体质测试页面

管理员和教师都可以访问，但是只有教师有操作权限。页面布局与上述页面相同，上方为搜索区域和新建按钮。每一项体测列表卡片显示了体测相关的详细信息，标题、科目组、科目数、参数测试的学生数、完成进度、创建时间等。操作按钮从左到右以此为体测成绩详情、编辑体测信息、删除体测、导入体测成绩和下载成绩模板。体测卡片显示的布局美观，信息饱满，能直观地了解体质测试的情况。

新建体质测试后，学生和教师完成体测工作，教师导入数据后由系统自动生成成绩，下图5-9展示了体测成绩详情页面。



图5‑9体质测试详情页面

页面上方为体质测试详细信息，下方为成绩数据搜索区域和展示表格，可以通过学号姓名搜索某个学生的成绩，每行数据可以点击左边展开箭头，查看成绩数据详情。考虑到成绩列较多，可以通过表格工具栏的列操作显示、隐藏、固定列，表格中部分数据为空表示该生无需测试该科目。体测数据导入系统“打分”后，教师可以导出体测成绩Excel表格。

切换学生账号登录，可以查看个人的所有体测信息。下图5-10为学生个人体测列表页面。



图5‑10学生个人体测列表页面

教师导入数据后学生可查看体测信息，包括已完成的、未完成的，其中已完成体测点击展开箭头可以查看成绩详情并通过图表对个人成绩进行展示分析，使学生对于个人的体质情况有良好的认识。

除此之外，虽然系统只有Web端，但是系统做了页面适配，采用响应式布局自动根据当前设备的尺寸调整页面，在移动设备上也能正常显示。如下图5-11所示。

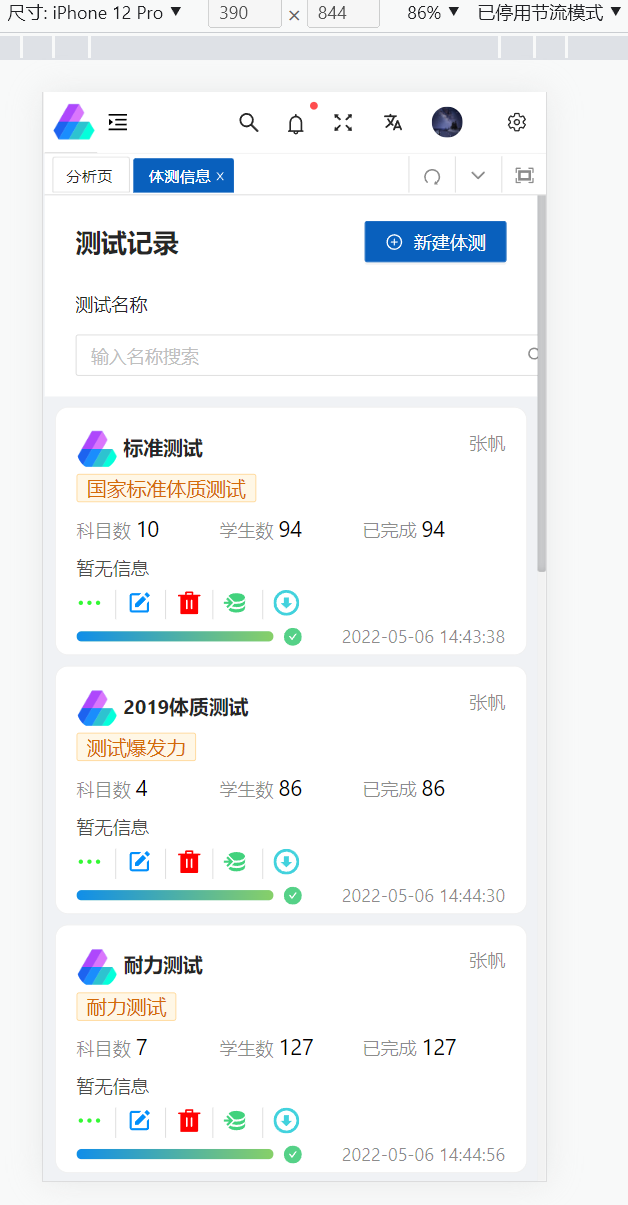


图 5‑11 系统页面适配

## 系统测试

为了保证体质健康数据管理系统满足需求分析的要求，找出与需求规格不符或矛盾的地方，及时发现潜在的漏洞，对已完成的体质健康数据管理系统进行了系统测试。测试的环境如下表所示。

表 5‑2系统测试环境配置

|  |  |
| --- | --- |
| 名称 | 配置 |
| 处理器 | Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz |
| 内存 | 8GB |
| 操作系统 | Windows 10 家庭中文版 |
| 操作系统版本 | 19043.1645 21H1 |
| 浏览器 | Google Chrome 101.0.4951.54 |

系统测试分为功能性与非功能性测试。前者是根据需求分析的内容，对系统已实现的功能进行验证，检查系统是否达到用户的要求，通过功能性测试还可以发现系统中设计交互不合理、影响用户体验的地方。功能性测试需要编写完整、全面、细致的测试用例，针对任何正常或非正常情况下可能出现的操作进行测试，以保证系统的安全稳定性。非功能性测试是对系统除了功能以外的其他性能指标进行测试，如性能测试、压力测试、安全性测试等。

在本次系统测试中，分模块对系统进行了功能性与非功能性测试，其中以功能性测试为主。下面对系统重要功能的测试情况进行介绍。

下表5-3为用户与权限模块功能测试用例。

表 5‑3用户与权限核心功能测试

|  |  |  |  |
| --- | --- | --- | --- |
| 用例名称 | 用户与权限模块功能测试 | | |
| 测试目的 | 用户与权限模块的各项功能是否完整和正确 | | |
| 功能名称 | 测试步骤 | 预期结果 | 测试状态 |
| 登录 | 1. 进入系统主页 2. 输入用户名密码，以管理员、教师、学生身份点击登录按钮 3. 登录成功后，参考需求分析，验证用户的权限 | 1. 主页正常打开 2. 登录成功或失败有提示信息 3. 用户权限控制正常 | 通过 |
| 系统权限 | 1. 以管理员身份登录 2. 点击权限菜单，查看系统权限 | 系统权限树正常显示 | 通过 |
| 创建角色 | 1. 以管理员身份登录 2. 选择权限菜单，点击新建角色按钮，填写表单 | 1. 表单数据校验 2. 角色创建成功或失败，有相应消息提示 | 通过 |
| 删除角色 | 1. 以管理员身份登录 2. 选择权限菜单，点击角色列表的删除按钮 | 删除成功或失败有相应的消息提示 | 通过 |
| 更新角色 | 1. 以管理员身份登录 2. 选择权限菜单，点击角色列表的更新按钮，填写表单 | 1. 表单数据校验 2. 角色更新成功或失败，有相应消息提示 | 通过 |
| 个人信息 | 1. 进入个人中心页面 2. 点击修改个人信息 3. 填写表单 | 1. 个人中心页面信息正确 2. 表单数据校验 3. 个人信息更新成功或失败消息提示 |  |

按照测试用例对用户与权限模块进行测试，不同用户登录的菜单、功能权限不一样，与需求规格相符，管理员对于系统角色的管理正常。

技术数据模块共包含四个子模块，以教师模块为例，下表5-4为基础数据教师模块功能测试用例。

表 5‑4基础数据教师模块测试

|  |  |  |  |
| --- | --- | --- | --- |
| 用例名称 | 基础数据教师模块功能测试 | | |
| 前置条件 | 以管理员身份登录 | | |
| 测试目的 | 基础数据教师模块的各项功能是否完整和正确 | | |
| 功能名称 | 测试步骤 | 预期结果 | 测试状态 |
| 数据查看 | 1. 进入基础数据教师界面 2. 搜索并查看教师信息 | 1. 页面显示正常 2. 搜索、分页功能正常 | 通过 |
| 模板下载 | 1. 进入基础数据教师界面 2. 点击教师信息页面的模板下载按钮 | 1. 成功下载Excel模板 2. Excel文件内容符合要求 | 通过 |
| 数据导入 | 1. 进入基础数据教师界面 2. 点击教师页面的导入数据按钮 3. 确定选择各种Excel文件 | 1. 文件数据成功导入，表格数据刷新 2. 提示文件异常信息，异常信息提示正确 | 通过 |
| 更新教师 | 1. 进入基础数据教师界面 2. 点击教师表格的更新按钮 3. 填写表单 | 1. 表单数据校验 2. 教师更新成功或失败，有相应消息提示 | 通过 |
| 删除教师 | 1. 进入基础数据教师界面 2. 点击教师信息的确认按钮，确认删除 | 删除成功或失败有相应的消息提示 | 通过 |
| 分配角色 | 1. 进入基础数据教师界面 2. 点击教师信息的分配角色按钮，选择角色分配 3. 切换该账号重新登录 | 1. 分配角色成功 2. 切换账号后对应角色的功能正常 | 通过 |

该模块涉及到主要是模板下载、数据导入、对教师信息的删改查，管理员在该模块中可以对教师进行角色分配。学院、班级、学生三个子模块的功能以及测试用例相似。经测试，各模块功能均正常。

下表5-5为体质测试模块的功能测试

表 5‑5体质测试模块功能测试

|  |  |  |  |
| --- | --- | --- | --- |
| 用例名称 | 体质测试模块功能测试 | | |
| 前置条件 | 以教师身份登录 | | |
| 测试目的 | 体质测试模块的各项功能是否完整和正确 | | |
| 功能名称 | 测试步骤 | 预期结果 | 测试状态 |
| 数据查看 | 1. 进入体测管理页面 2. 查看体测信息 | 1. 页面显示正常 2. 搜索、分页功能正常 | 通过 |
| 模板下载 | 1. 进入体测管理页面 2. 点击新建体测按钮 3. 填写表单 | 1. 表单数据校验 2. 体测更新成功或失败，有相应消息提示 | 通过 |
| 成绩模板下载 | 1. 进入体测管理页面 2. 点击成绩模板下载按钮 | 1. 成功下载Excel模板 2. Excel文件内容符合要求 | 通过 |
| 更新体测 | 1. 进入体测管理页面 2. 点击更新体测按钮 3. 填写表单 | 1. 表单数据校验 2. 体测更新成功或失败，有相应消息提示 | 通过 |
| 删除体测 | 1. 进入体测管理界面 2. 点击删除体测按钮，确认删除 | 删除成功或失败有相应的消息提示 | 通过 |
| 体质测试详情 | 1. 进入体测管理界面 2. 点击查看体测详情按钮 | 1. 打开新页面 2. 体测和成绩信息正确 | 通过 |
| 成绩导出 | 1. 进入体测管理页面 2. 点击体测详情，点击导出成绩按钮 | 1. 文件成功下载 2. 文件内容正确 | 通过 |
| 导入成绩 | 1. 进入体测管理界面 2. 点击导入成绩按钮 3. 确定选择各种Excel文件 | 1. 文件数据成功导入，表格数据刷新 2. 提示文件异常信息，异常信息提示正确 | 通过 |

体质测试模块作为系统的核心功能，上述功能测试用例全部通过。

## 本章总结

本章主要介绍了体质健康数据管理系统的实现与测试情况，并对主要的功能界面进行了介绍展示，然后对系统的主要功能编写测试用例进行了测试，保证了系统功能的完整性和正确性。

# 总结与展望

体质健康数据管理系统以青少年体育工作实际需求为背景，以软件工程的设计思想，结合SpringBoot、Vue、MySQL等技术，完成了从项目背景调研、系统需求分析、系统详细设计，编写代码实现和系统测试的整个过程，系统实现了体质健康数据的存储、分析、上报、体育综合素质评价等功能，使体育工作者对体测数据的录入、管理、统计上报更加智能化、统一化、高效化。本文主要论述体质健康数据管理系统设计实现过程中的的以下内容。

1. 充分调研了体质健康现状与相关系统，查阅国家体质测试标准，针对现有体质测试工作存在的问题提出解决方案。
2. 通过分析总结，对系统整体进行模块划分，通过用例图描述了各个模块中管理员、教师、学生用户的详细需求，并根据需求，有针对性地选用相关的技术框架，确定了系统的整体设计方案。
3. 针对系统需求分析和整体设计，确定了系统使用到的技术框架，通过实体关系图进行了复杂的数据库设计，基于RBAC模型对系统权限进行了设计，系统前后端交互采用了RESTful架构，最后对系统各个模块功能通过类图、流程图等进行了详细设计。
4. 按照设计，编码实现体质健康数据管理系统，并对系统已实现的功能进行了测试工作，保证了系统功能满足需求规格的定义。

除此之外，在整个设计实现过程中，也存在一些问题需要改进。

1. 对于统计分析中的数据分析，需要结合机器学习、数据挖掘等技术以实现智能的运动处方推荐，本文仅做了需求设想，限于个人水平，并未实现该功能。
2. 随着系统服务的复杂度提高，系统架构可以向微服务架构升级，增强系统的扩展性和可靠性。

# 致谢

几个月的毕业设计接近尾声，在本次课题中，真的收获良多。查阅资料、设计系统、学习技术、编写代码、撰写论文，过程中有功能实现时的满足，也有遇到困难时的疲惫，最后还是完成了从问题调研、系统设计实现到完成论文的整个过程，对如何利用所学知识系统性的分析解决问题有了更多的体验。本次工作中感受最深刻的一点就是做研究必须首先对问题有一个清晰的认识，明白需求是什么，盲目地设计实现并不能解决任何问题，反而徒增工作量，本人就走进了这个误区，本次毕业设计是一次宝贵的经历。

在这里，感谢西南交通大学给我提供了这样一个学习平台，在这里我学习计算机相关的知识和系统性地解决问题的方法，对个人能力的提升很大。

非常感谢我的指导老师钟灿老师，钟老师丰厚的学识，负责的态度，在我工作的过程中给予了细致的指导，帮助我走出误区，完成毕业设计。对老师表示衷心的感谢，祝老师身体健康，万事如意！

# 参考文献

1. 邢文华. 中国青少年体质的现状及加强青少年体育的紧迫性[J]. 青少年体育，2012(01):5-6.
2. 中共中央国务院关于加强青少年体育增强青少年体质的意见[J]. 中国学校卫生,2007(06):481-483.
3. 中共中央国务院. 国家中长期教育改革和发展规划纲要(2010-2020年)[N]. 中国教育报,2010-07-31(01).
4. 国务院办公厅转发教育部等部门关于进一步加强学校体育工作若干意见的通知[J]. 中华人民共和国国务院公报,2012(31):13-16.
5. 国务院关于印发全民健身计划(2016—2020年)的通知[J]. 中华人民共和国国务院公报,2016(19):9-15.
6. 中共中央国务院印发《“健康中国2030”规划纲要》[J]. 中华人民共和国国务院公报,2016(32):5-20.
7. 焦怡迪. 天津市学生体质健康大数据平台的设计与实现[D]. 硕士论文. 天津体育学院, 2021.
8. 姜颖. 大数据视域下青少年体质健康数据分析处理系统研究与设计[D]. 山东体育学院,2018.
9. 2014年全国学生体质健康调研结果[J]. 中国学校卫生,2015,36(12):4
10. 第八次全国学生体质与健康调研结果发布[J]. 中国学校卫生,2021,42(09):1281-1282
11. 赖锦松余卫平. 我国青少年体质监测管理成效、问题与对策[J]. 吉林体育学院学报,2016,32(04):55-59.
12. 万小兰. 我国青少年体质健康评价标准研究综述[J]. 当代体育科技,2021,11(28):43-45.
13. 王洋,王旭光,宋岩,谭健,张丰刚. 国外青少年儿童体质健康管理概述与启示[J]. 教育教学论坛,2013(03):187-188.
14. Fitnessgram & Activitygram Test Administration Manual-Updated 4th Edition[M]. Human Kinetics:2010-07-30.
15. Li Min. Design and Implementation of College students' Physical Health Test System based on Data Mining[A]. Institute of Management Science and Industrial Engineering.Proceedings of 2019 3rd International Conference on Artificial intelligence,Systems,and Computing Technology(AISCT 2019)[C].Institute of Management Science and Industrial Engineering:计算机科学与电子技术国际学会(Computer Science and Electronic Technology International Society),2019:6.
16. Wang Hui,Wang Ning,Li MeiJie,Mi Simeng,Shi YaYa. Student Physical Health Information Management Model under Big Data Environment[J]. SCIENTIFIC PROGRAMMING,2021,2021.
17. 陈华卫,窦丽,侍崇艳.中美青少年体质健康监测与后续干预比较[J]. 中国学校卫生,2018,39(10):1443-1448.
18. 刘小卉. 中学生体质测评成绩管理系统的设计与实现[D]. 山东大学,2013.
19. 贺慨. 大学生体质健康运动处方系统的设计与实现[D]. 北京工业大学,2013.
20. 徐刘柱. 湖北省大学生体质健康管理平台的研制[D]. 华中师范大学,2014.
21. 朱广涛. 基于数据挖掘的学生体质健康测试系统的设计与实现[D]. 山东大学,2015.
22. 黄雨博. 基于Loushang5平台的体质健康数据管理系统的设计与实现[D]. 山东大学,2017.
23. 张亚琦. 基于数据挖掘技术的高校学生体能分析系统设计与实现[D]. 华中师范大学,2020.
24. 张卫华. 基于SSM的权限管理系统及数据可视化[D]. 北京邮电大学,2020.
25. 胡世平. 基于机器学习的大学生体质健康的运动处方智能推荐研究[D]. 西华大学,2021.
26. 于洋. RESTful架构风格及其演变与发展[J]. 计算机时代,2020(04):10-13.