

学号：2021010876

哈尔滨师范大学 学士学位论文

题 目	基于 Django 的网课教学系统
学 生	郝镇华
指导教师	石晔琼 讲师
年 级	2021 级
专 业	软件工程
系 别	软件工程系
学 院	计算机科学与信息工程学院

哈尔滨师范大学

2025 年 5 月

基于 Django 的网课教学系统

郝镇华

摘要：随着互联网技术的快速发展和在线教育的普及，传统的教育模式已经无法满足现代学习者的多样化需求。本文设计并实现了一个基于 Django 和 React 的在线教育平台，旨在提供一个功能完善、用户体验良好的在线学习环境。

该平台采用前后端分离架构，后端使用 Django REST 框架构建 RESTful API，前端使用 React 和 TypeScript 开发响应式用户界面。系统主要包含用户管理、课程管理、视频流媒体、练习测试和评论交流等核心模块。在用户管理方面，实现了基于 JWT 的身份认证和权限控制；在课程管理方面，支持教师创建、编辑课程，学生浏览、学习课程；在视频流媒体方面，提供直播和点播功能；在练习测试方面，支持教师创建练习，学生提交答案并获取反馈；在评论交流方面，实现了课程评论、回复和通知功能。

本文详细阐述了软件的需求分析、概要设计、数据库结构、API 设计和前端实现。在开发过程中，采用了模块化设计思想，确保系统的稳定运行。同时，保证了用户的体验，实现了响应式设计，使平台能够在不同设备上提供一致的使用体验。

本系统能够有效支持远程教学、混合式教学等多种教学模式，提高教学效率和学习效果。系统的模块化设计也为未来的功能扩展提供了良好的基础。

关键词：在线教育平台；Django；React；前后端分离；视频流媒体

An Online Education System Based on Django

Abstract: With the rapid development of Internet technology and the popularization of online education, traditional educational models can no longer meet the diverse needs of modern learners. This paper designs and implements an online education platform based on Django and React, aiming to provide a feature-rich and user-friendly online learning environment.

The platform adopts a front-end and back-end separation architecture, with the back-end built using the Django REST framework to construct RESTful APIs, and the front-end developed with React and TypeScript to create a responsive user interface. The system mainly includes core modules such as user management, course management, video streaming, practice tests, and comment interaction. In terms of user management, JWT-based authentication and permission control are implemented; for course management, teachers can create and edit courses, while students can browse and study courses; for video streaming, both live and on-demand functions are provided; for practice tests, teachers can create exercises, and students can submit answers and receive feedback; for comment interaction, course comments, replies, and notification functions are implemented.

This paper elaborates on the software's requirements analysis, high-level design, database structure, API design, and front-end implementation. During the development process, a modular design approach was adopted to ensure the stable operation of the system. At the same time, user experience was prioritized, and responsive design was implemented to provide a consistent experience across different devices.

This system can effectively support various teaching modes such as remote teaching and blended learning, improving teaching efficiency and learning outcomes. The modular design of the system also provides a solid foundation for future functional expansion.

Keywords:Online Education Platform,; Django; React; Front-end and Back-end Separation; Video Streaming

目 录

第 1 章 绪论.....	1
1.1 课题背景及研究的目的与意义.....	1
1.2 国内外研究现状.....	1
1.3 主要研究内容及章节安排.....	2
1.3.1 主要研究内容.....	2
1.3.2 章节安排.....	2
第 2 章 需求分析.....	4
2.1 概述.....	4
2.2 课程模块.....	4
2.3 课程评分模块.....	5
2.4 评论模块.....	6
2.5 试题模块.....	6
2.6 直播模块.....	7
2.7 后端人员管理模块.....	8
第 3 章 概要设计.....	9
3.1 概述.....	9
3.2 课程模块.....	9
3.3 课程评分功能.....	10
3.4 评论功能.....	11
3.5 试题功能.....	11
3.6 直播功能.....	12
3.7 用户管理功能.....	13
第 4 章 详细设计.....	15
4.1 课程模块.....	15
4.2 课程评分功能.....	16
4.3 评论功能.....	18
4.4 试题功能.....	20
4.5 直播功能.....	22
4.6 用户管理功能.....	24
4.7 数据库设计.....	26

4.7.1 数据库概念结构设计	26
4.7.2 数据库逻辑结构设计	27
第 5 章 系统实现	34
5.1 课程模块系统	34
5.2 课程评分功能展示	35
5.3 评论功能展示	36
5.4 试题功能展示	36
5.5 直播功能展示	37
5.6 用户管理功能	38
第 6 章 系统测试	39
6.1 网课教学系统详细测试	39
6.2 课程模块测试	39
6.3 评分模块测试	40
6.4 评论模块测试	41
6.5 试题模块测试	41
6.6 直播模块测试	42
6.7 用户管理测试	43
第 7 章 总结与展望	44
参考文献	45
致 谢	47

第 1 章 绪论

1.1 课题背景及研究的目的与意义

信息技术不断进步，教育行业也随着改变。尤其是在新冠疫情爆发之后，线上教学成为全球范围内的重要教学方式。传统的面对面教学模式已经不能完全满足新时代下灵活、个性化和高效率的教学需求。因此，构建一套高效、稳定、功能完善的在线教学系统已成为现代教育的重要发展方向。

Python 作为一门最近新兴的编程语言，因其语法简洁、应用广泛、学习曲线平缓等优势，已被广泛应用于人工智能、数据分析、web 开发等多个领域，也成为众多高校和培训机构的必修课程。与此同时，学习 Python 的人群不断扩大，涵盖了中学生、大学生、职场转型者及编程爱好者等多个层次。如何为这一庞大的学习群体提供一个高质量的在线学习平台，成为了一个值得深入研究和开发的方向。

基于此，设计并开发一个专注于 Python 教学的在线网课系统，不仅具有现实意义，也具备较强的应用前景。本课题的研究目标是构建一套基于 Web 的教学系统，涵盖课程管理、用户权限管理、教学视频上传与播放、直播授课、课后练习、自动批改、评论互动等模块，形成完整的教学闭环，满足管理员、教师、学生三类用户的不同需求。

系统中，管理员主要在后台中进行系统总管理，包括用户管理、课程审核、数据统计等操作；教师作为课程内容的创造者和教学的组织者，能够进行课程的创建，发布教学内容，布置试卷，并根据学生的答题情况分析和评价学生的学习情况；学生则是平台的使用者和学习主体，可以浏览和学习课程内容，观看教学视频，参与直播课堂，完成习题练习，并通过评论系统与教师进行互动交流。系统将采用前后端分离的开发模式，后端基于 Django 框架实现稳定的数据管理与接口服务，前端使用 React 构建响应式的交互界面，以提升用户体验。

本项目的研究不仅为 Python 教学提供了一个实用的教学平台，也有助于推广信息化教育理念，提升教学效率与资源利用率。在当前线上教育需求不断增长的背景下，该系统具备较强的现实适应性和推广价值。更重要的是，通过本系统的实现与推广，能够有效打破地域限制，使优质教育资源更加公平地惠及广大学习者，助力教育公平与终身学习目标的实现。

1.2 国内外研究现状

近年来，随着信息技术的不断发展，网络教育逐步成为教育发展的主流方向。尤其是疫情之后，远程教育和在线学习平台的发展迎来了新的高潮。国内外在线教育系统的研究和开发方面已取得了一定的成果，相关产品和平台在技术实现、教学组织形式、用户交互体验

等方面不断优化升级，为本课题提供了丰富的理论支持与技术参考。

在国外，MOOCs（大规模开放在线课程）平台的发展尤为突出，如 Coursera、edX、Udacity 等平台不仅提供了丰富的课程资源，还在教学方式、互动机制、智能推荐、学习分析等方面进行了大量探索和技术积累。这些平台普遍采用视频授课、自动批改、论坛讨论、测验考试等手段形成完整的教学闭环。尤其在编程教学方面，国外很多平台已经引入了代码实时运行环境、互动式教学和个性化推荐算法，极大提升了学习效率和用户体验。

国内在线教育行业也发展速度迅猛。国内各个在线教育平台逐步形成了完善的在线教学体系，涵盖了基础教育、高等教育和职业培训多个层次。在 Python 教学方面，诸如慕课网和哔哩哔哩等平台提供了大量针对不同学习阶段和实际应用场景的课程资源，具备较强的实用性和灵活性。同时，钉钉、腾讯课堂等工具型平台也在疫情期间承担了大规模远程教学的任务，其技术架构和用户管理经验对于本课题有一定借鉴价值。

然而，无论是国外的成熟平台还是国内的本土化系统，大多为通用型平台，适用于多种学科、专业与课程类型，缺乏对特定知识领域如 Python 编程的深度定制，尤其在练习系统、代码自动批改、个性化练习反馈等方面还存在改进空间。此外，现有多数平台功能模块分散，系统结构复杂，用户学习路径不够清晰，教学反馈机制不够及时，不利于构建一个完整、高效的教学闭环。

因此，基于当前国内外研究成果与应用实践的基础上，本课题试图聚焦于 Python 编程这一具体知识领域，设计并实现一个集课程学习、直播授课、视频教学、习题训练、答题自动评测、师生互动于一体的专业化教学平台。系统将融合现代 Web 开发技术与教育信息化理念，优化用户使用体验，提高教学互动质量，为 Python 编程教学提供一个更具针对性与实用性的解决方案。通过对国内外研究现状的分析可见，本项目具有鲜明的研究意义和实践价值。

1.3 主要研究内容及章节安排

1.3.1 主要研究内容

本论文主要对该系统的需求分析，概要设计，详细设计，系统实现，系统测试进行研究。本系统主要为了力求构建一个功能完备、结构清晰、交互良好的 Python 在线教学平台，满足当前远程教学与在线编程学习的现实需求。

1.3.2 章节安排

本论文分为七个主要部分：

绪论：介绍了系统的国内外研究现状，明确了研究的目的和意义，并概述了系统的主要

研究内容和章节安排。

需求分析：详细描述了系统的整体功能，并通过用例图展示各个模块的主要功能。

概要设计：对系统的功能模块设计进行了详细说明，并通过活动图展示了各模块的设计思路。

系细设计：与概要设计相对应，详细描述了各模块功能的逻辑实现，并展示了时序图、类图以及数据库表结构。

系统实现：通过图片展示了核心模块的实现过程，提供了直观的技术细节。

系统测试：对开发完成的系统进行了全面测试，确保其功能和性能符合预期。

总结与展望：总结了系统开发的全过程，并对未来的改进和发展方向进行了展望。

第 2 章 需求分析

2.1 概述

我们首先明确了系统的整体目标和用户需求。通过对用户角色的分析，识别出系统的主要用户群体，包括学生、教师和管理员。接着，我们对系统的核心功能进行了详细描述，涵盖了课程管理、用户管理、直播功能、练习与评测、数据统计等模块。每个模块的功能需求通过用例图进行了可视化展示，帮助理解各模块之间的交互和依赖关系。如图 2-1 所示。

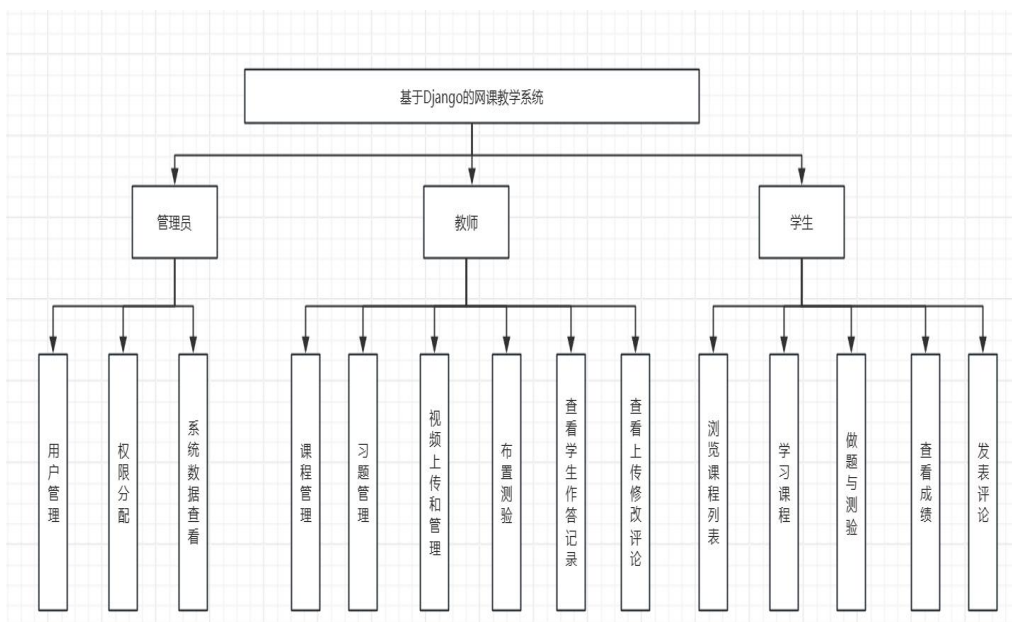


图 2-1 网课教学系统功能图

2.2 课程模块

课程模块是网课系统的核心功能模块，面向学生、教师和游客。用户进入系统后，首先接触到的就是课程模块。该模块通过精心设计的界面，为用户提供多样化的课程资源，支持用视频进行课程学习，以满足不同用户的学习需求。课程模块不仅展示课程信息，还提供课程学习、管理以及互动等功能，是连接教师与学生的重要桥梁。学生可以进行课程的学习与观看，教师可以在这个模块中创建新课程进行授课。课程模块的设计充分考虑了用户体验和功能完整性这个模块通过精心设计的架构和功能，为在线教育提供了完整的解决方案，既满足了学习者的需求，也为教师提供了便捷的教学工具。系统的可扩展性和灵活性，使其能够适应不同规模和类型的在线教育场景。如图 2-2 所示。

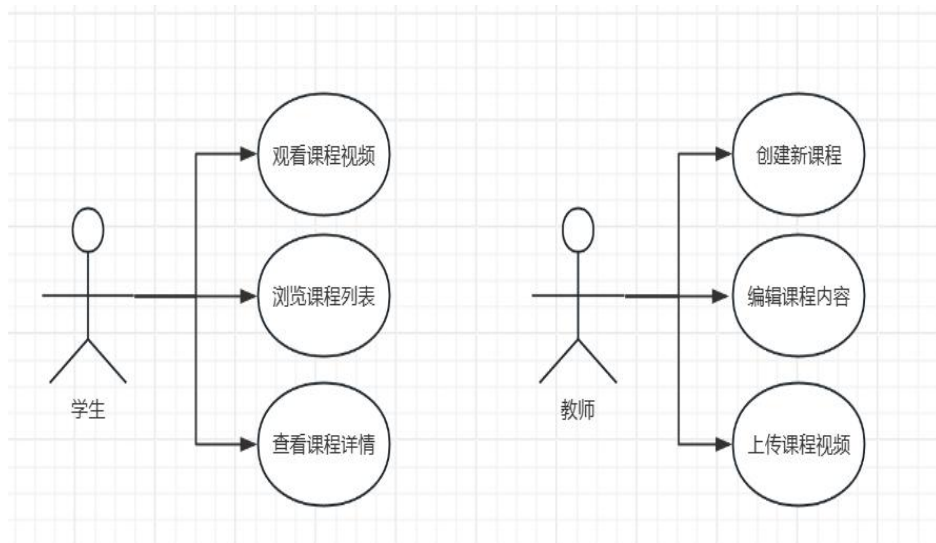


图 2-2 课程模块用例图

2.3 课程评分模块

课程评分模块面向学生跟老师，它是网课系统中至关重要的组成部分，它不仅为学生提供了表达学习体验和反馈的渠道，也为教师提供了改进教学内容和方法的依据，同时帮助平台进行课程质量监控和优化。该模块通过多样化的评分功能和便捷的用户互动设计，增强了学生与教师之间的互动，提升了整个学习平台的用户体验和教学质量。学生在完成课程学习后，可以通过评分模块对课程进行评分和撰写评价。这种评价不仅包括简单的星级评分，还可以包含详细的文字反馈，帮助教师了解学生的学习感受和课程中的不足之处。如图 2-3 所示。

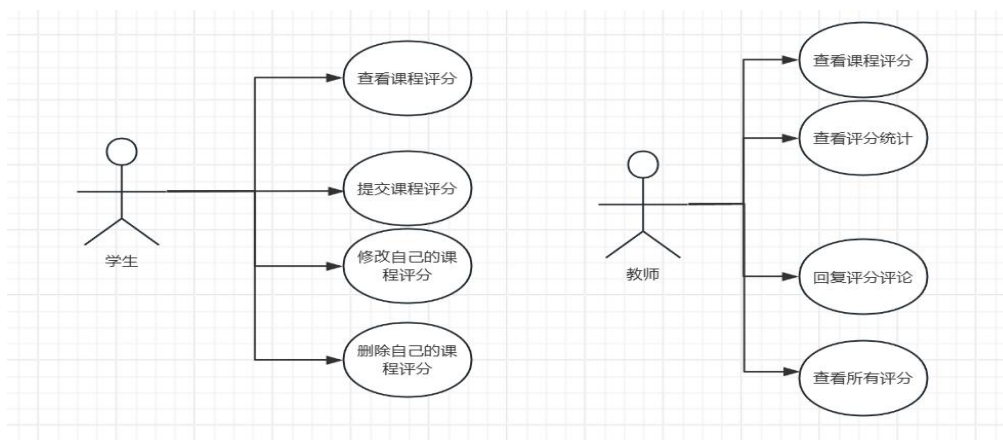


图 2-3 课程评分模块用例图

2.4 评论模块

该模块面向学生和教师，可以进行评论的浏览，评论模块是网课系统中增强用户互动和提升学习体验的重要组成部分。它为学生、教师提供了一个交流的平台，学生可以在这里分享学习方法、提出问题、互相解答，从而形成一个活跃的学习社区。学生可以在课程详情页面或特定内容下方发表评论，分享自己的观点和学习体会。评论内容会实时展示，其他用户可以查看并参与讨论，用户能够对他人的评论进行回复，形成多层次的评论结构，促进深入交流。用户还可以对有价值的评论进行点赞，以表达认可和支持。同时，系统会根据点赞数量对评论进行排序，突出显示热门评论。

如图 2-4 所示。

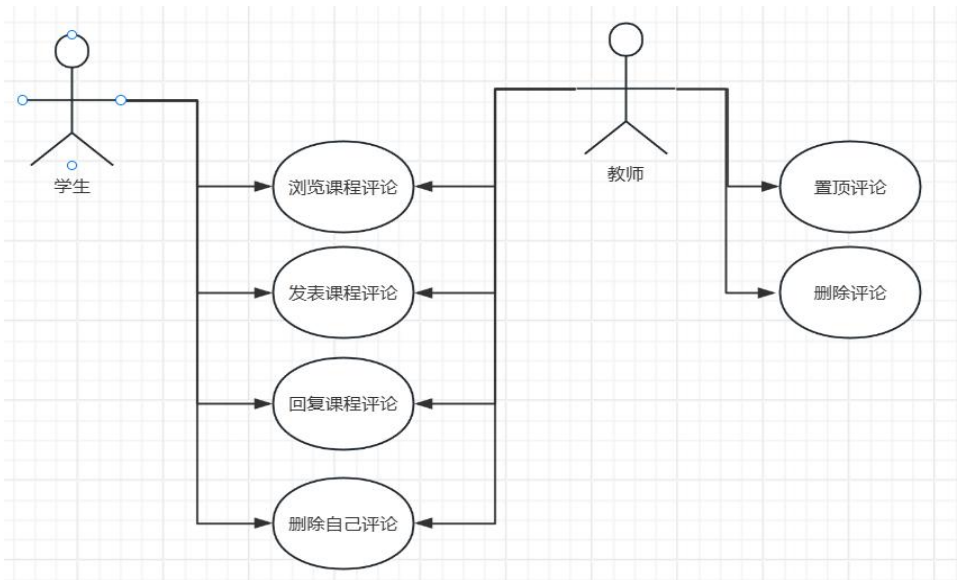


图 2-4 评论模块用例图

2.5 试题模块

试题模块是网课系统中用于评估学习效果、巩固知识和提升教学质量的关键功能模块。它为教师提供了创建、管理和发布试题的工具，同时为学生提供了练习、考试和即时反馈的平台。通过多样化的题型和灵活的测试形式，试题模块不仅帮助学生检验学习成果，还为教师提供了详细的学习分析报告，助力教学优化。教师可以创建不同类型的试题，并对试题进行编辑和管理。系统可以批量导入和导出试题，方便教师快速准备教学资源。学生可以参加教师发布的考试，提交答案后，系统自动评分客观题，教师负责主观题的评分。学生可以在考试结束后查询成绩，并查看详细的考试分析报告。如图 2-5 所示。

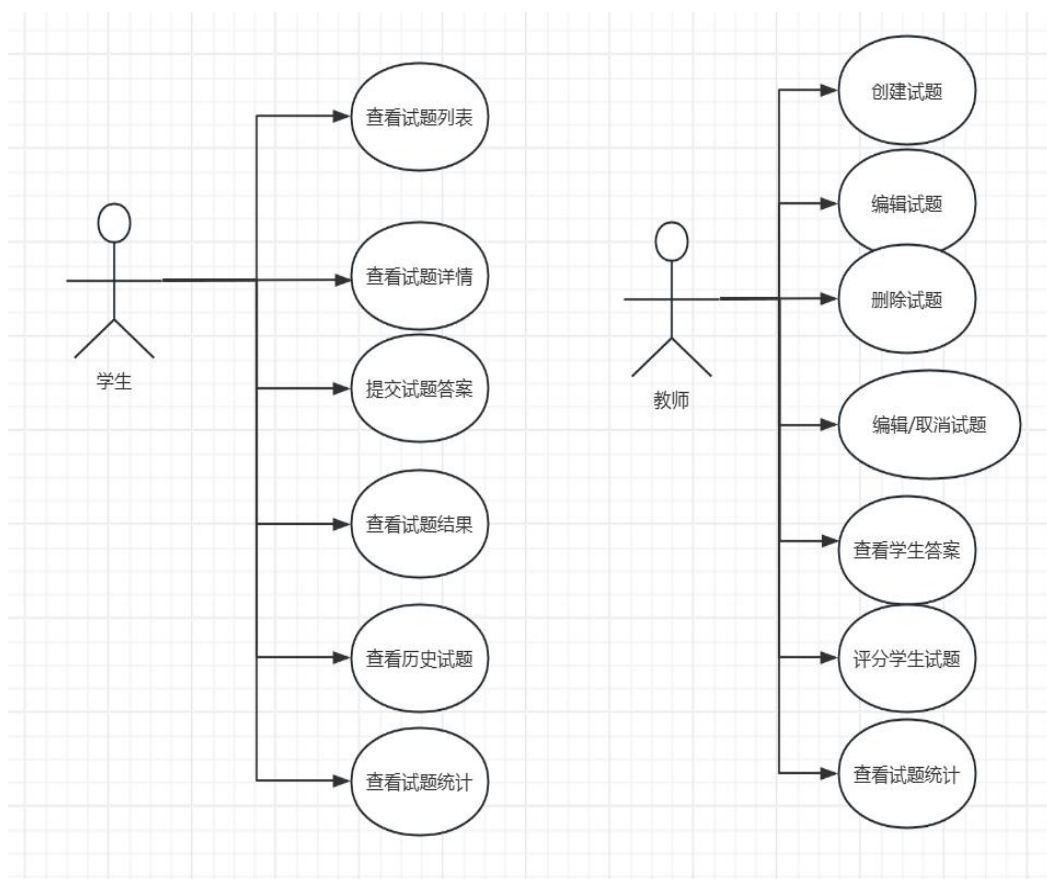


图 2-5 试题模块用例图

2.6 直播模块

直播模块是网课系统中最具互动性和实时性的功能模块，它为教师和学生提供了一个接近传统课堂的在线学习环境。通过直播功能，教师可以在线对学生进行课程教学。直播模块不仅提升了学习体验，还通过回放功能为学生提供了复习的便利，确保每个学生都能跟上学习进度。教师可以创建直播课程，设置直播的基本信息，如标题、描述、开始时间等。直播编辑功能允许教师在直播前对内容进行调整和更新，确保直播内容的准确性和完整性。直播发布后，教师可以实时监控直播状态，确保教学过程顺利进行。直播通过 obs 链接推流，保证了播放的稳定性。如图 2-6 所示。

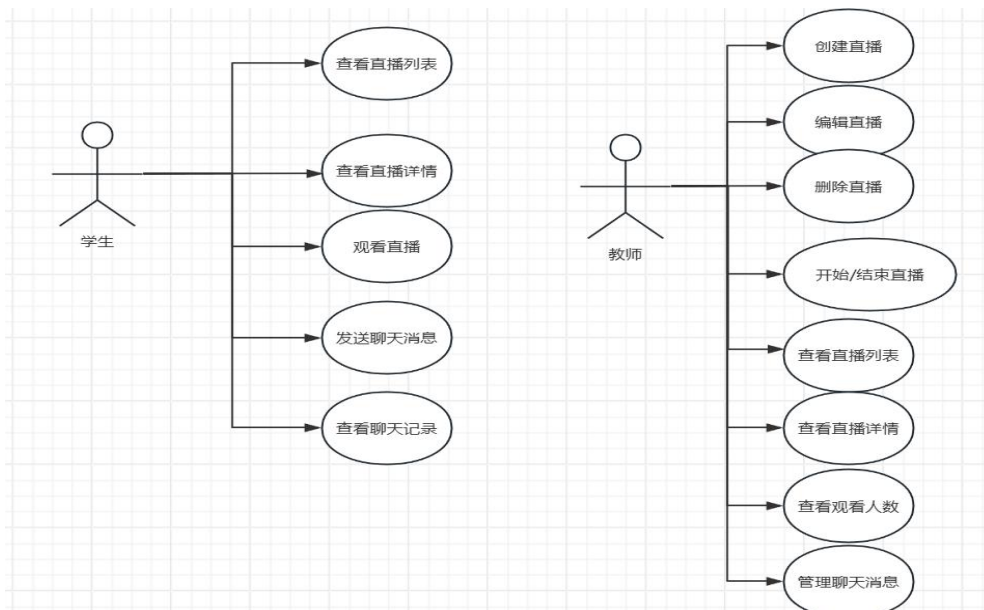


图 2-6 直播模块用例图

2.7 后端人员管理模块

管理模块是网课系统的后台控制中心,为管理员提供了全面的管理功能,包括用户管理、角色与权限管理等功能。此模块确保了系统的安全性、稳定性和可维护性,是保障在线课程系统运行的关键。管理员可以查看系统中所有用户的基本信息,包括用户名、注册时间、联系方式等,编辑用户信息,为用户分配不同的角色和权限,控制用户对系统功能的访问这个管理模块通过完善的功能设计和严格的权限控制,为系统提供了可靠的后台管理支持,确保了系统的安全稳定运行。同时,通过丰富的数据统计和监控功能,为系统运营提供了有力的数据支持。如图 2-7 所示。

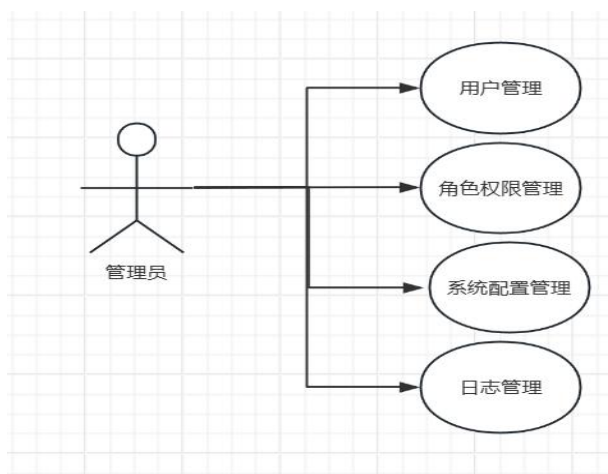


图 2-7 后端人员管理模块用例图

第 3 章 概要设计

3.1 概述

网课教学系统分为前端与后端，前端教师和学生使用，完成课程，课程评分，评论，试题，直播等模块；后台管理者使用，完成用户管理的任务。系统可以构建一个功能完备、结构清晰、交互良好的 Python 在线教学平台。

3.2 课程模块

在主页用户可以选择是否登录。在登录时，用户输入账号密码进行身份验证。在注册时，用户填写注册信息并提交，完成注册后可进入系统。用户可以在个人中心或课程学习页面浏览课程列表。教师或管理员可以管理课程内容，包括更新课程资料、发布课程活动等。对于需要申请的课程，用户提交加入申请，课程模块将为用户提供一个全面、便捷的学习平台，同时为教师提供有效的课程管理和互动工具，提升网课系统的整体教学效果。如图 3-1 所示。

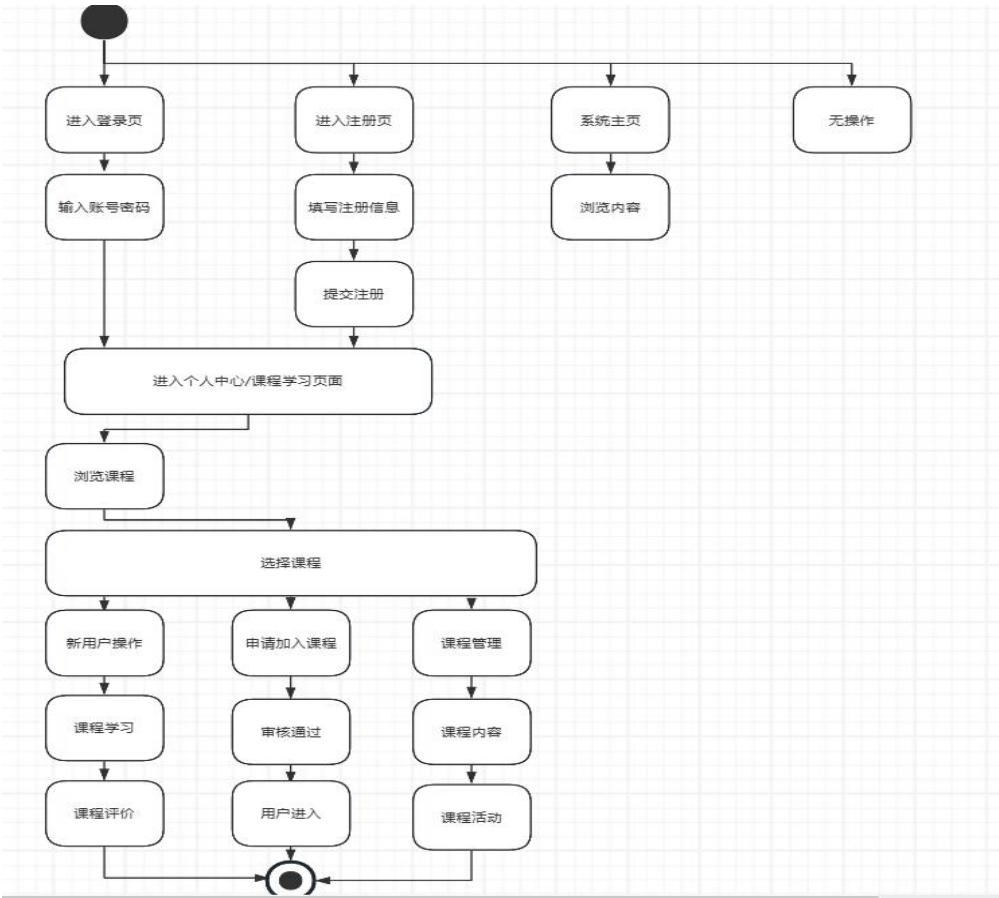


图 3-1 课程模块活动图

3.3 课程评分功能

当学生进入课程详情页并点击评分按钮时，系统首先检查用户是否处在登录状态。如果不处于登录状态，则跳转到主页，提示用户登录后才能进行课程评分。如果已登录，继续进行评分资格检查。用户在评分界面选择评分等级并提交。系统接收到评分后，显示成功提示，确认评分已提交。评分模块将为网课系统提供一个有效的反馈机制，帮助教师改进课程内容，同时为学生提供选择课程的参考，提升整个平台的教学质量和用户体验。如图 3-2 所示。

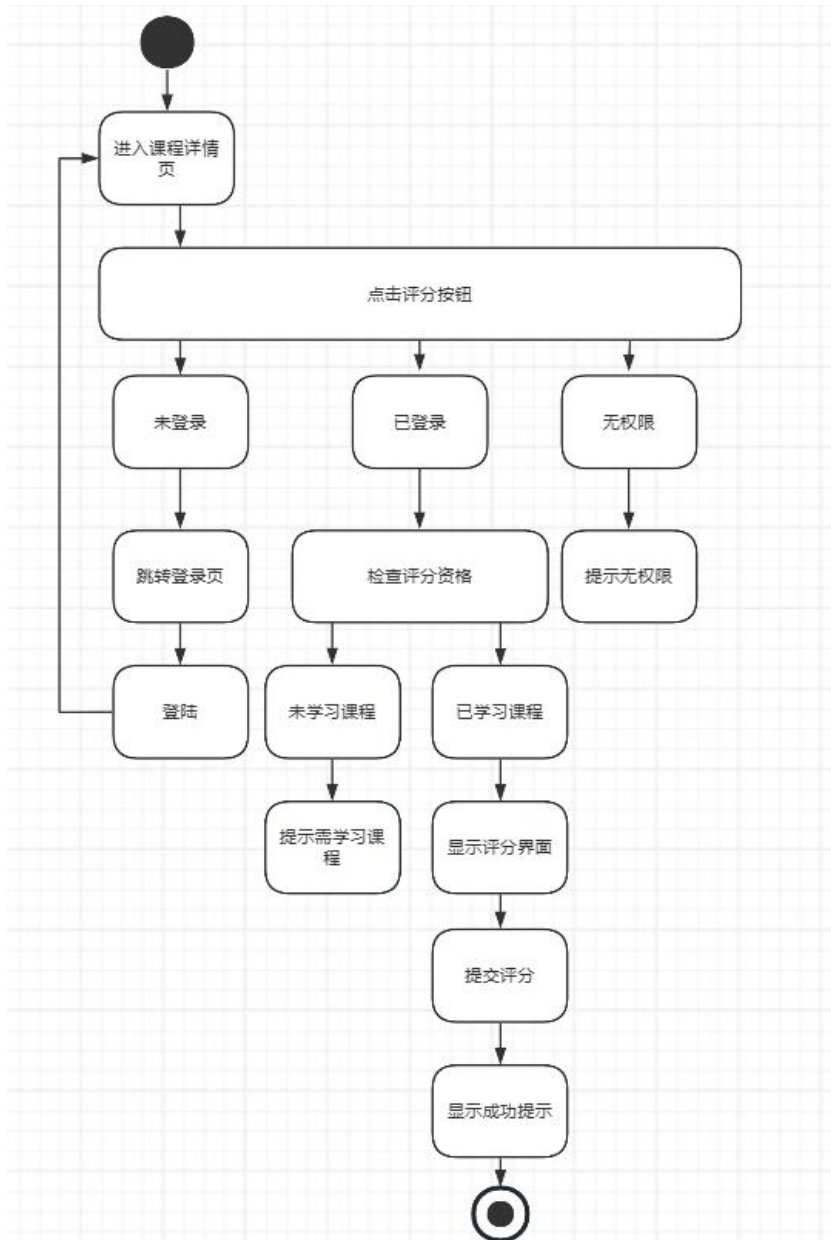


图 3-2 课程模块活动图

3.4 评论功能

该模块面向学生和教师，进入课程的评论区以查看所有评论的列表。评论列表可以按照时间顺序进行排序，方便学生查看最新的评论。学生可以点击评论查看详细的评论内容。可以查看其他用户对某条评论的回复。如果学生想回复某条评论，需要先登录。如果未登录，系统会跳转到登录页面。教师可以查看所有评论，删除违反规定的评论。系统会更新评论列表，确保列表中不包含被删除的评论。如图 3-3 所示。

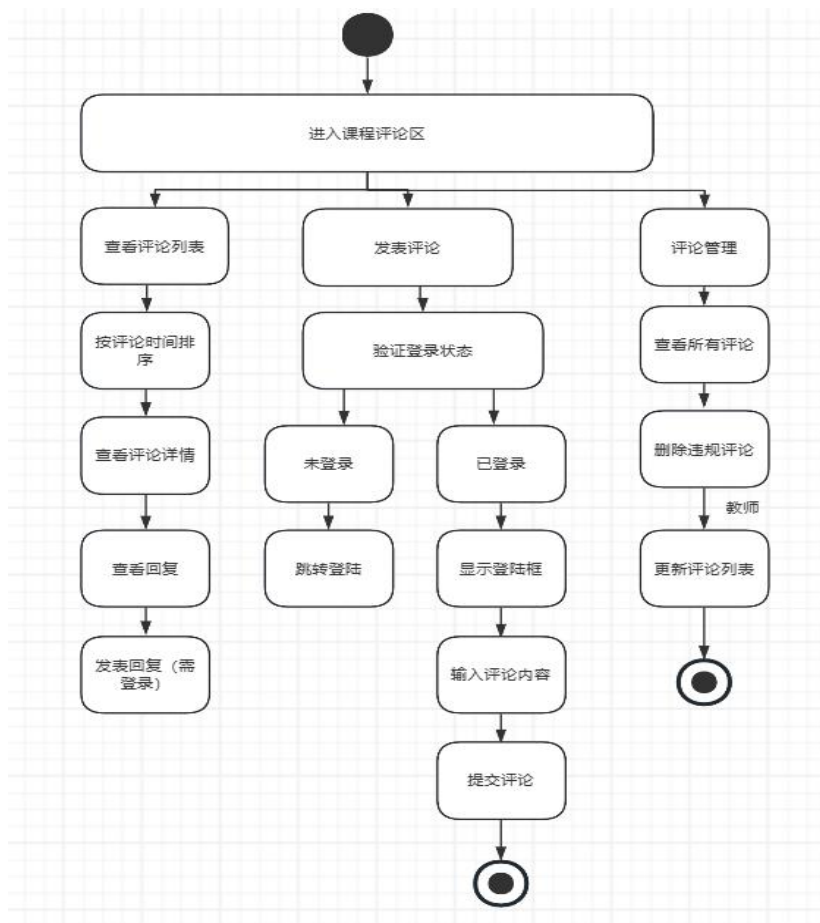


图 3-3 评论功能活动图

3.5 试题功能

模块面向学生和教师，学生首先可以查看教师发布的试题，选择并进入具体的试题或测试，回答试题中的问题。答题结束后，学生可以查看自己的得分。教师的主要流程则是从创建试题到查看学生成绩。编写具体的试题内容，在学生提交答案后，教师开始批改试卷。试题模块是为了提供一个平台，让学生能够在线进行测试和练习，同时让教师能够创建和管理试题，以及评估学生的学习成果。

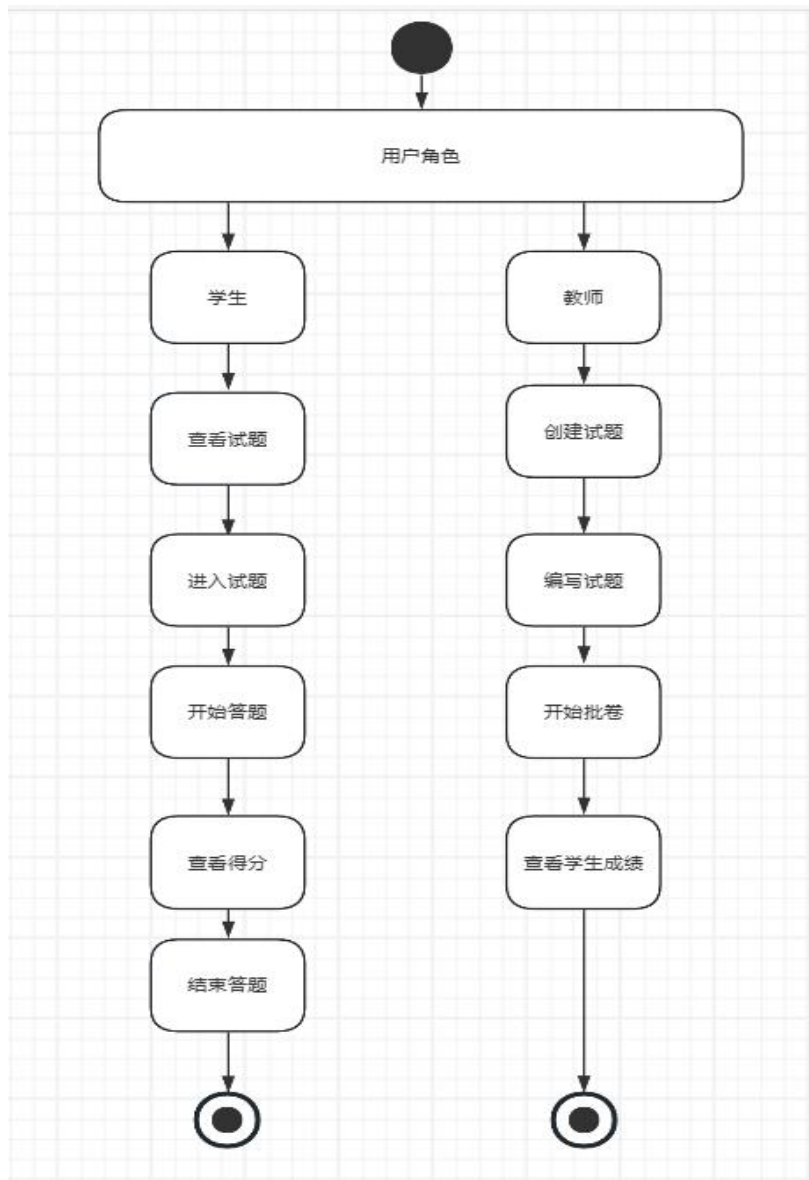


图 3-4 试题功能活动图

3.6 直播功能

直播功能模块是网课系统中实现实时教学和互动的关键组成部分。该模块支持教师进行在线直播授课，学生参与直播学习和互动，从而提供一种接近传统课堂的在线学习体验。教师在系统中创建新的直播课程，设定直播的基本信息。配置直播的具体信息，如直播时间、直播内容、参与规则等，在设定的时间启动直播，开始在线授课。学生可以浏览系统中所有可用的直播课程列表，进入选定的直播间，准备参与直播学习，直播结束后退出直播间

如图 3-4 所示。

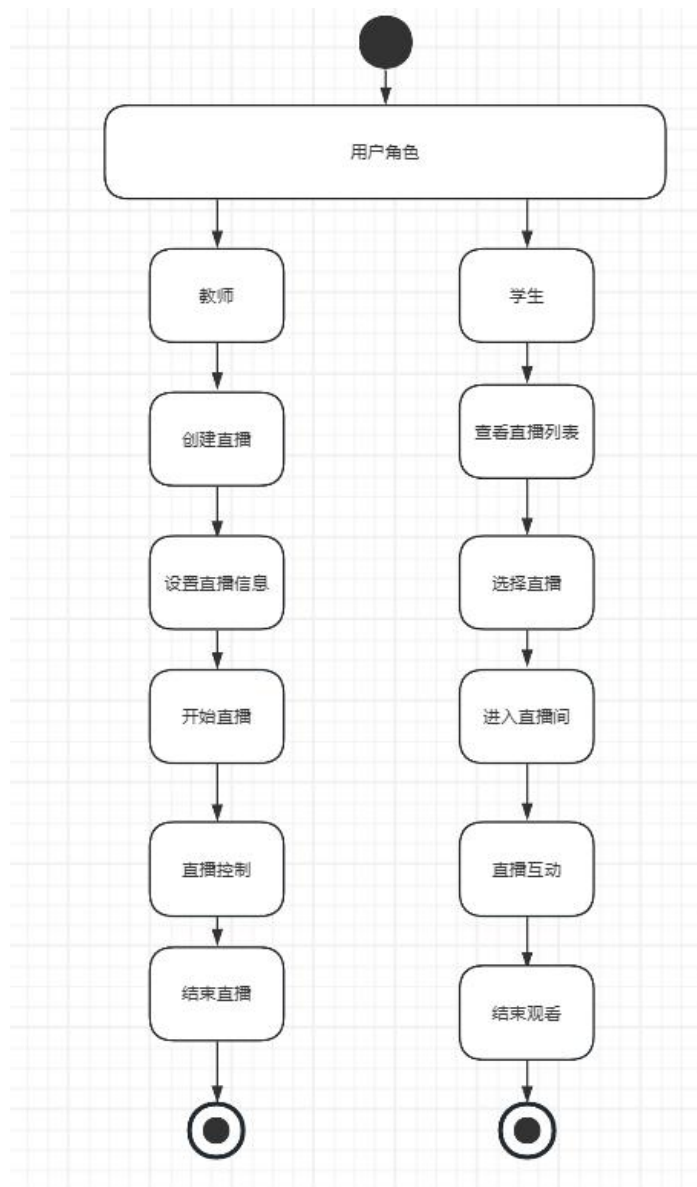


图 3-5 直播功能活动图

3.7 用户管理功能

首先管理员登陆进入后台之后可以登录后，管理员可以选择不同的管理模块进行操作。这些模块包括用户管理、课程管理、评论管理、评分管理、反馈管理。在选择具体的管理模块后，管理员可以对模块中的数据进行增加、查看、修改和删除等操作，允许管理员对系统中的数据进行维护和更新。这个流程图清晰地展示了管理员在系统中的主要操作流程，从登录到选择管理模块，再到对数据进行增删改查，最后记录操作并结束管理活动。

如图 3-6 所示。

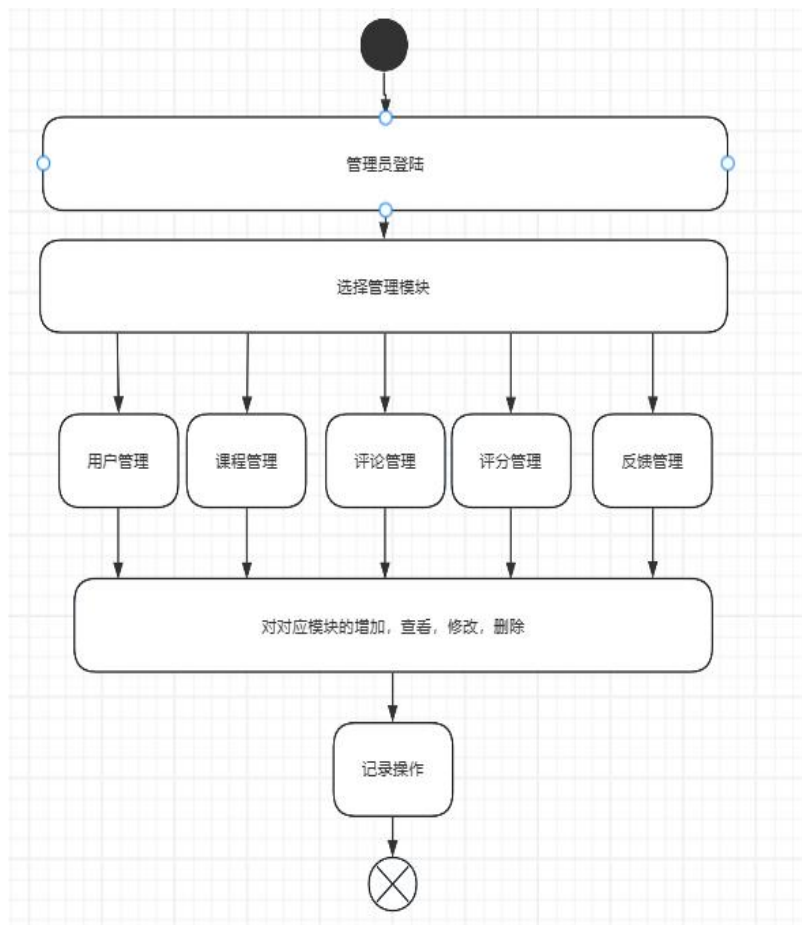


图 3-6 后台人员管理功能活动图

本模块对系统的各模块概要设计进行了详细说明，网课教学领养系统分为前端与后端，前端教师和学生使用，完成课程学习，课程评分，评论，试题模块，直播模块等；后台管理者使用，完成用户管理、数据管理等任务。系统用于满足当前远程教学与在线编程学习的现实需求。

第 4 章 详细设计

4.1 课程模块

系统提供两种身份验证方式：登录和注册。用户可以根据自己的情况选择进入登录页面或注册页面。用户登录系统后，可以通过个人中心或课程学习页面浏览课程列表。课程管理功能教师和管理员拥有课程管理权限，可以更新课程资料，发布课程活动，管理课程内容等。对于需要申请的课程，用户需要提交加入申请。教师审核申请审核通过后，用户才能进入课程学习。

系统价值如图 4-1 所示。

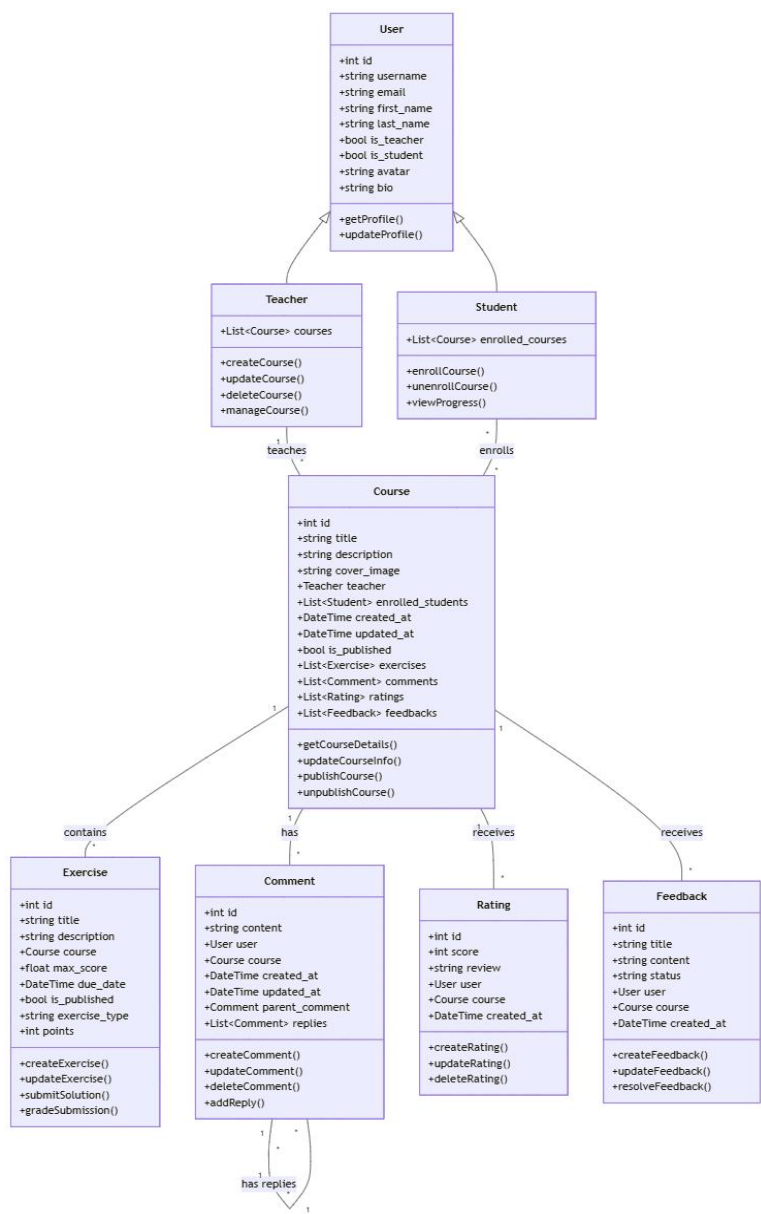


图 4-1 课程模块类图

教师首先点击创建课程，前端界面显示课程创建表单，教师填写课程信息并上传封面。然后，前端发送创建课程请求到 API 服务，API 服务验证请求数据后，将课程封面上传到存储服务并返回文件 URL，接着将课程信息保存到数据库并确认保存。创建结果返回给前端界面，显示创建成功。如果教师需要编辑课程内容，也会通过前端发送更新请求到 API 服务，API 服务更新课程信息并确认更新，最后返回更新结果给前端界面，显示更新成功。如图 4-2 所示。

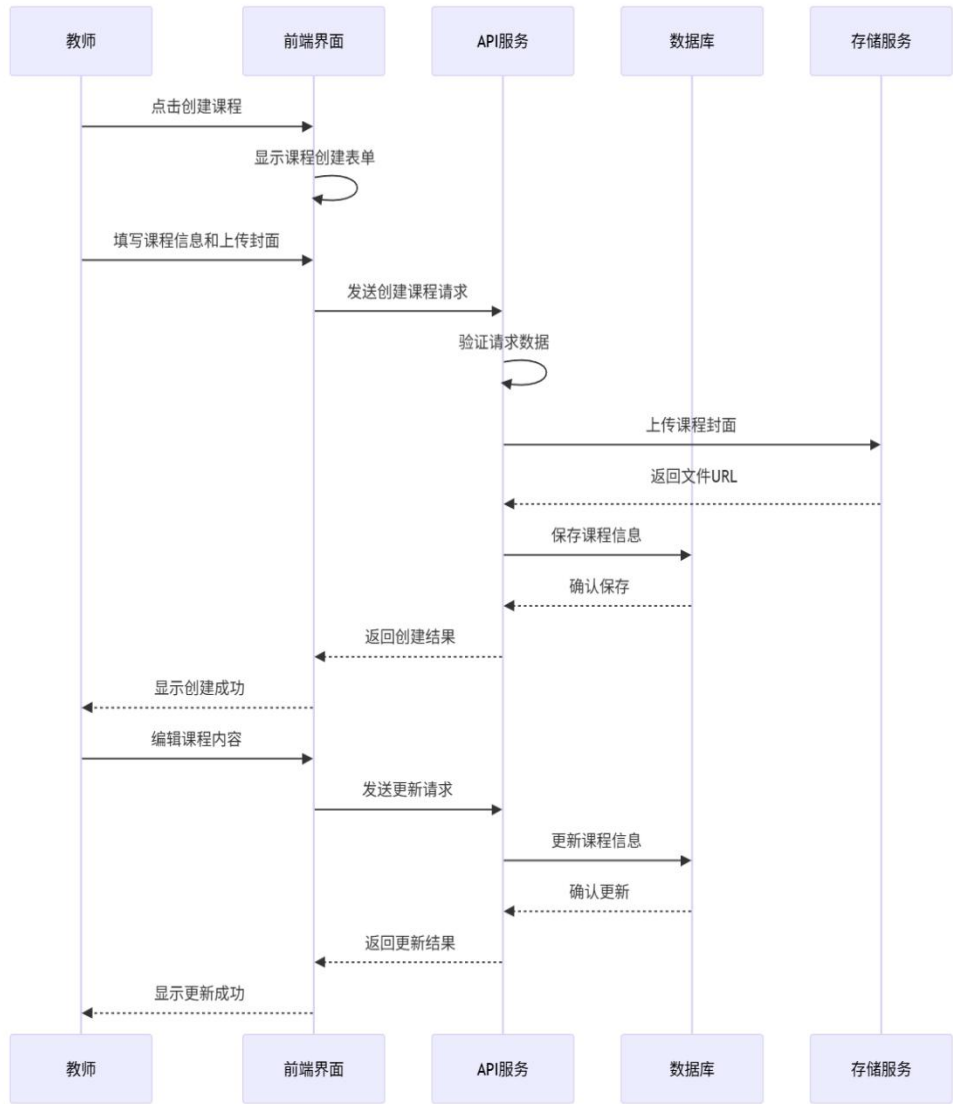


图 4-2 课程模块时序图

4.2 课程评分功能

当学生在课程详情页面点击评分按钮时，系统首先验证用户是否处于登录状态。如果不处于登录状态，系统会自动跳转到主页，并提示用户需要登录才能进行评分；然后系统继

续进行下一步。用户在评分界面选择适当的评分等级，然后用户可以填写评分意见，系统接收评分数据并处理。通过这种设计，评分模块不仅满足了用户表达意见的需求，还为平台的持续优化提供了 **valuable** 的数据支持，形成了一个促进教育质量提升的良性循环。如图 4-3 所示。

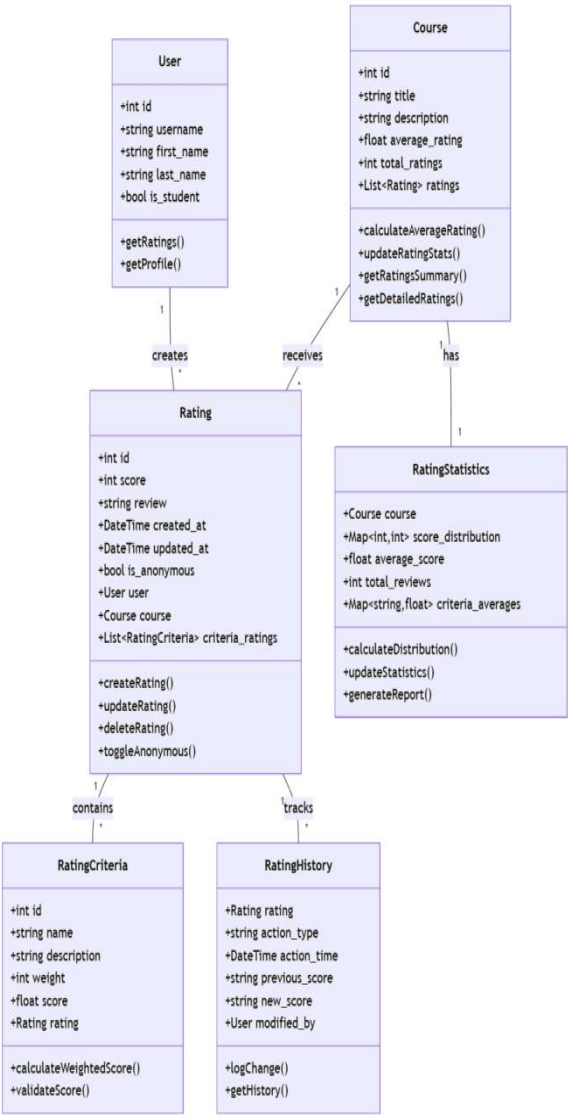


图 4-3 课程评分功能类图

用户在前端界面点击评分按钮，触发评分流程。前端界面显示评分表单，用户可以在表单中输入评分和评论。用户提交评分表单，前端界面将评分数据发送给 API 服务。API 服务将请求发送给验证服务，验证用户是否有权限进行评分。验证服务返回验证结果给 API 服务。API 服务将评分数据保存到数据库。数据库确认评分数据已成功保存。API 服务将评分结果返回给前端界面。前端界面显示评分成功的提示信息给用户。如图 4-4 所示。

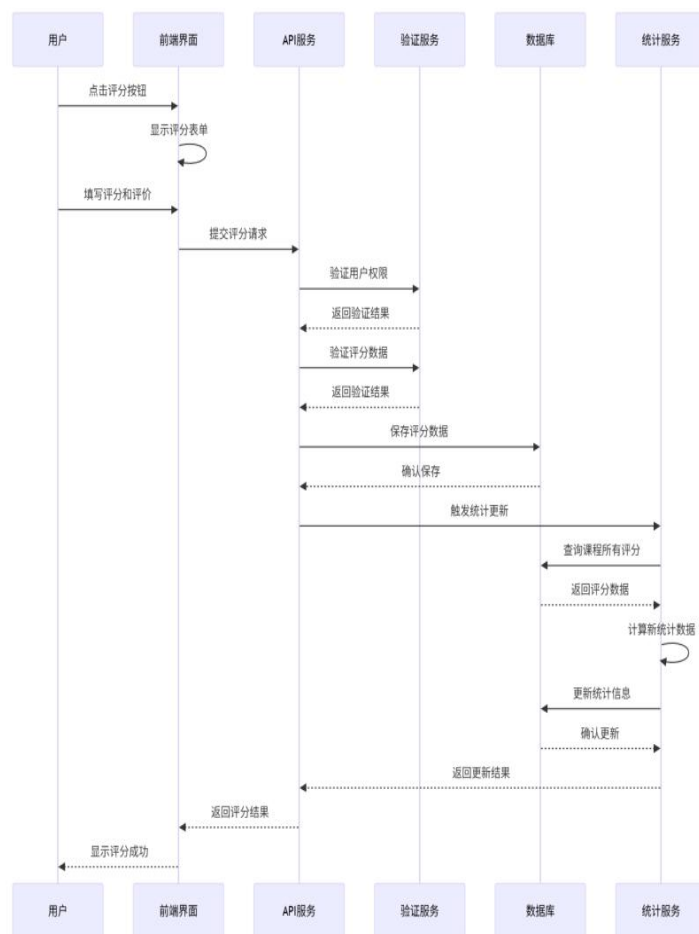


图 4-4 课程评分功能时序图

4.3 评论功能

进入课程的评论区以查看所有评论的列表。评论列表可以按照时间顺序进行排序，方便学生查看最新的评论。学生可以点击评论查看详细的评论内容。可以查看其他用户对某条评论的回复。如果学生想回复某条评论，需要先登录。如果未登录，系统会跳转到登录页面。教师可以查看所有评论，删除违反规定的评论。系统会更新评论列表，确保列表中不包含被删除的评论。有评论展示功能，评论管理功能，用户体验优化，安全防护功能，通知系统集成，数据统计功能，这个评论功能通过完善的设计和实现，为用户提供了良好的互动体验，同时也为教师提供了有效的管理工具。系统的实时性和互动性，促进了用户之间的交流和讨论，增强了学习社区的活跃度。

如图 4-5 所示。

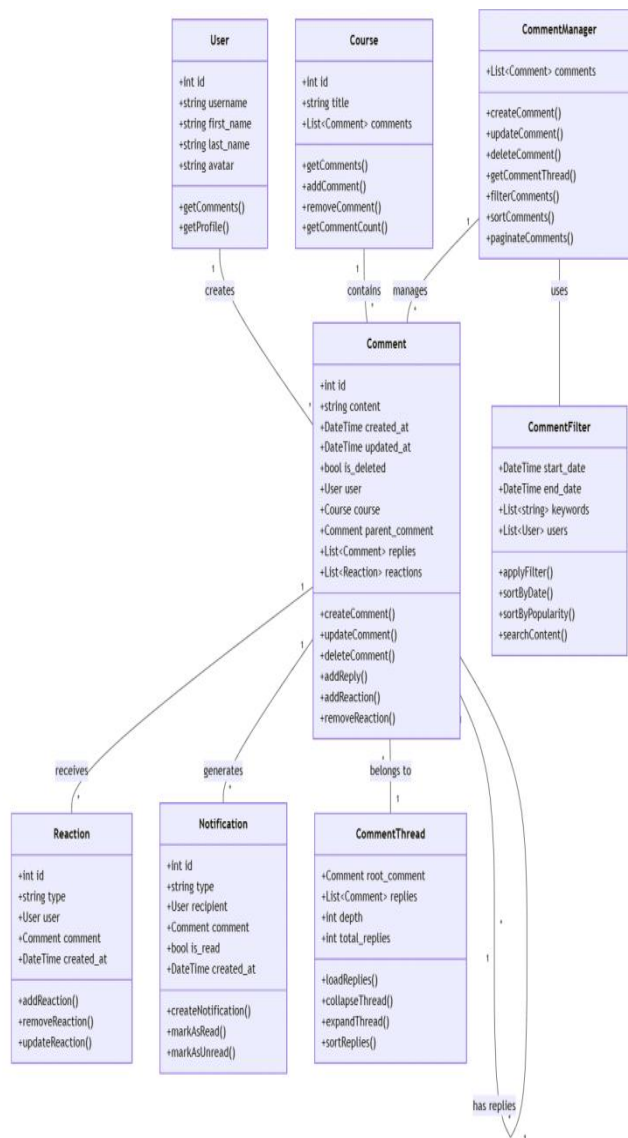


图 4-5 评论功能类图

用户向系统发送登录请求。系统接收到登录请求后，验证用户的身份。系统将验证结果返回给用户。如果验证成功，用户可以进行后续操作。用户请求查看评论。系统从数据库中获取评论列表。数据库将评论列表返回给系统。系统将评论列表展示给用户。用户提交新的评论。系统将新评论存储到数据库。数据库确认评论已存储。系统更新评论列表以包含新评论。用户选择编辑已有评论。系统更新数据库中的评论信息。系统再次更新评论列表以反映编辑后的评论。用户选择删除评论。系统从数据库中删除评论。系统更新评论列表以反映删除操作。如图 4-6 所示。

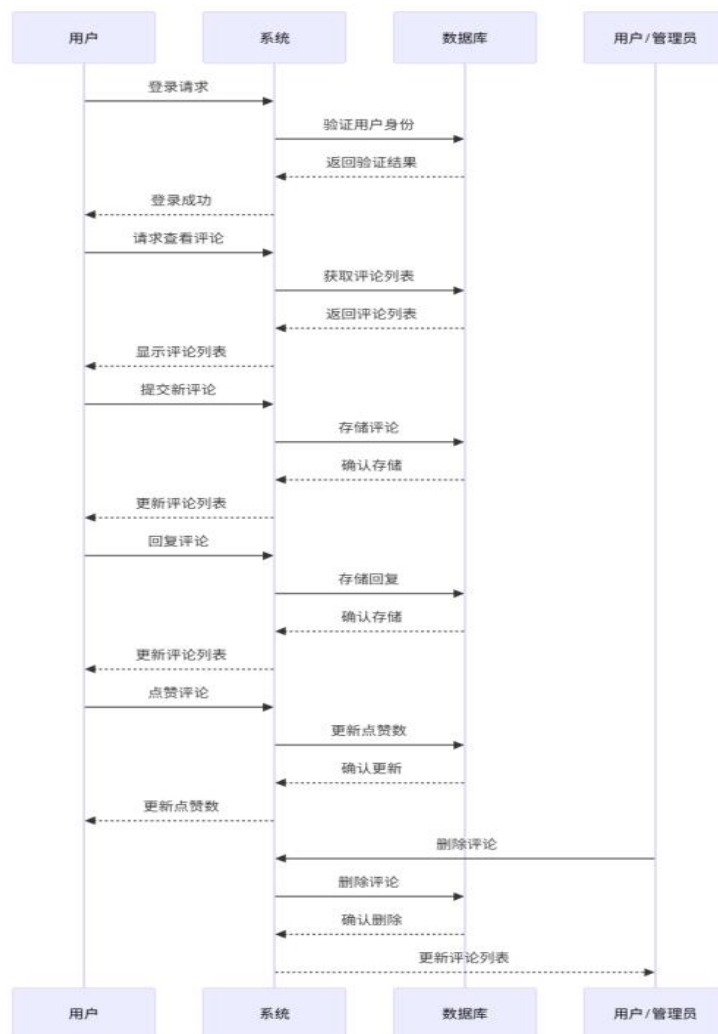


图 4-6 评论功能时序图

4.4 试题功能

试题模块主要服务于两类用户：学生和教师，各自拥有不同的操作流程和权限。学生可以浏览教师发布的各类试题和测试，根据个人需求选择特定的试题或测试进行练习。学生可以进入选定的试题界面，阅读题目并回答相关问题，最后提交答案完成测试。答题结束后可以立即查看个人得分，了解自己的学习成果和表现。教师可以创建各类试题和测试，设计试题内容和结构，设置评分标准和规则。等学生答完题之后教师可以查看学生的答题情况，批改学生提交的试卷，分析学生的成绩数据。这个试题模块通过完善的功能设计和实现，为教师提供了强大的试题管理工具，同时也为学生提供了便捷的答题体验。系统的灵活性和可扩展性，使其能够满足不同场景下的教学需求。如图 4-7 所示

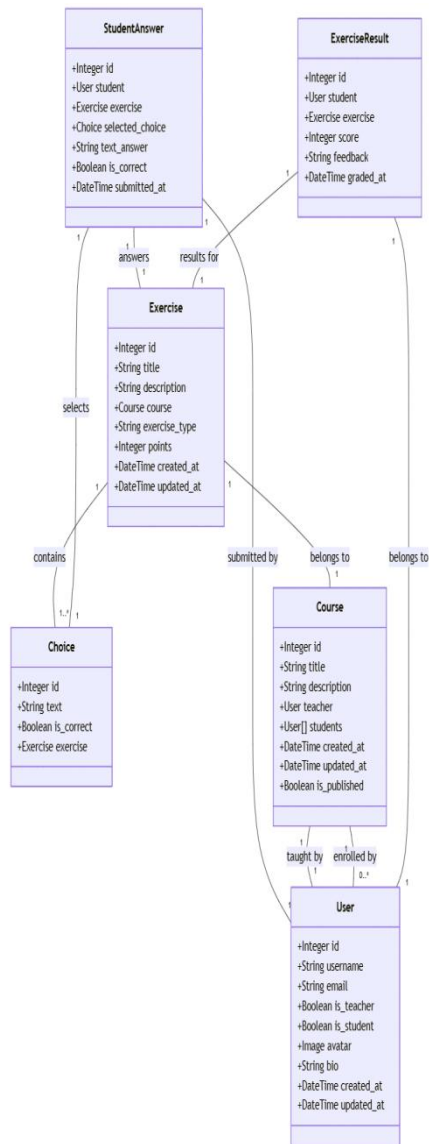


图 4-7 试题功能类图

学生向系统请求可用的测试列表。系统向教师请求试题列表，教师从数据库获取试题数据并返回给系统，系统再将试题列表展示给学生。学生在试题列表选择一个试题。系统向教师请求该试题的详细信息，教师从数据库获取试题详情并返回，系统将试题详情展示给学生。学生开始答题，学生点击开始答题，系统展示题目。学生完成答题后提交答案。系统将答案保存到数据库，并进行自动评分，更新成绩。系统将评分结果返回给学生，学生可以立即看到自己的得分。学生可以选择查看每道题的答案解析。系统向教师请求答案解析，教师从数据库获取解析内容并返回，系统展示解析给学生。如图 4-8 所示

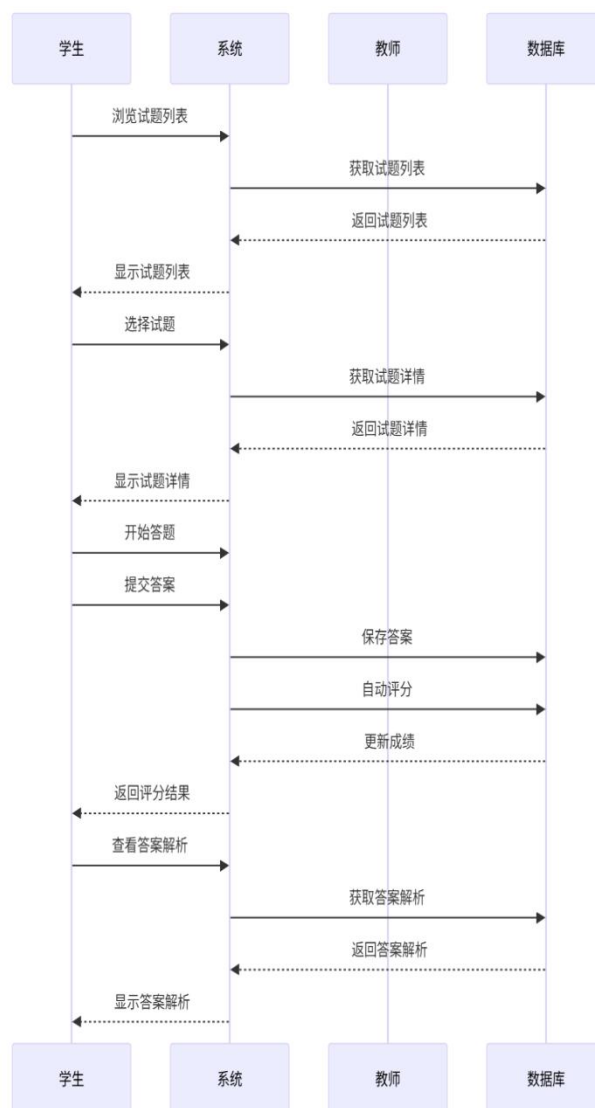


图 4-8 试题功能时序图

4.5 直播功能

直播功能模块是网课系统的核心互动组件，也是分两类用户：教师和学生。教师可以在系统中创建直播课程，设置课程标题、简介和封面图可以灵活配置直播参数，包括开始时间、预计时长、参与人数上限等支持设置直播互动规则，如是否允许学生提问。提供直播准备室功能，教师可以在正式开播前测试设备、调整画面直播过程中可以使用白板、PPT 等教学工具辅助讲解，可以实时查看在线学生数量、互动数据等统计信息。如图 4-9 所示。

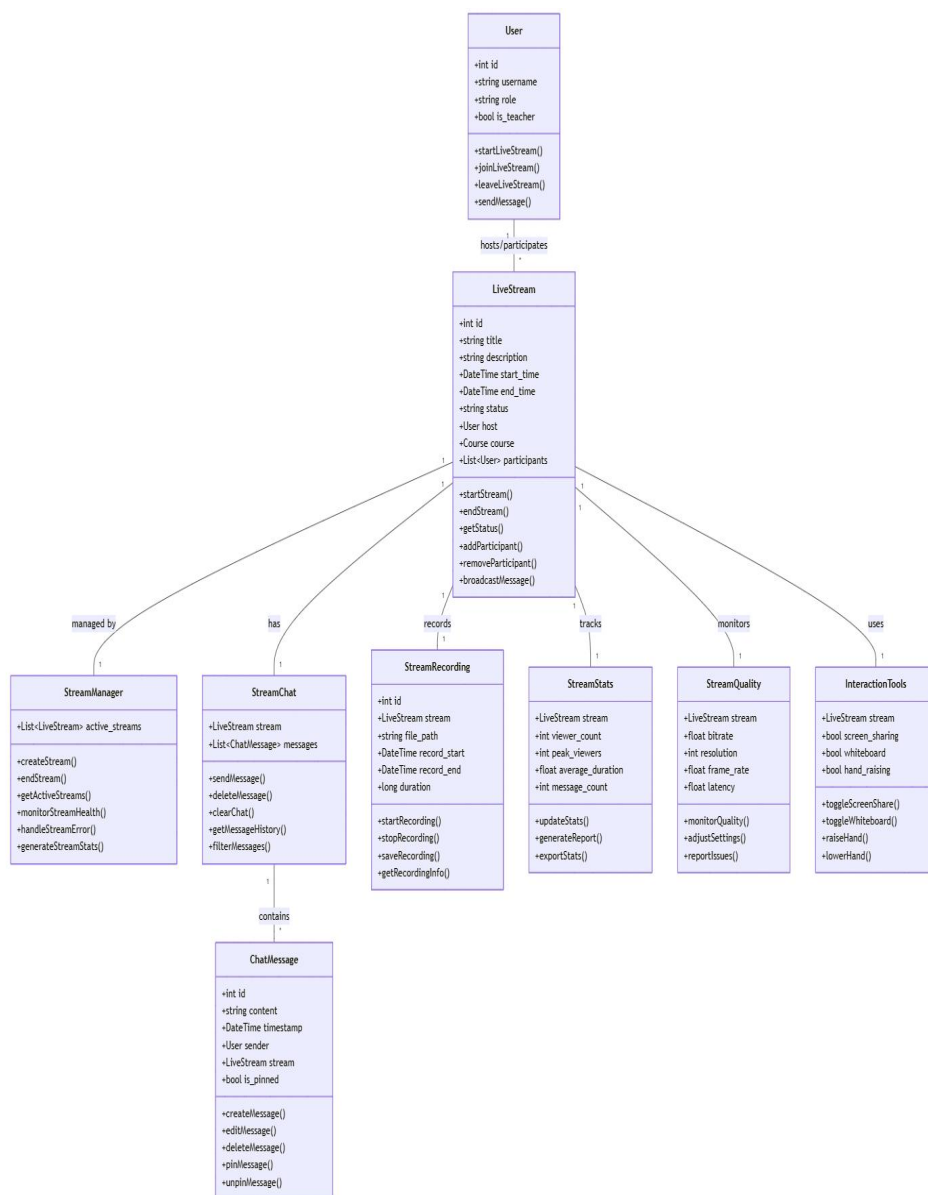


图 4-9 直播功能类图

教师在前端界面点击开始直播的按钮，前端界面向 API 服务发送请求，请求创建一个新的直播。API 服务将请求发送给验证服务，以验证教师是否有权限创建直播。验证服务完成权限验证后，将结果返回给 API 服务。API 服务向流媒体服务发送请求，初始化直播所需的流媒体服务。流媒体服务完成初始化后，返回流媒体配置信息给 API 服务。API 服务在数据库中创建一条新的直播记录。数据库确认直播记录已成功创建。API 服务向通知服务发送请求，通知直播已经开始。通知服务在数据库中保存通知记录。通知服务将直播开始的实时通知推送给相关用户。API 服务将直播配置信息返回给前端界面。前端界面向教师显示直播已成功开始的消息。如图 4-10 所示。

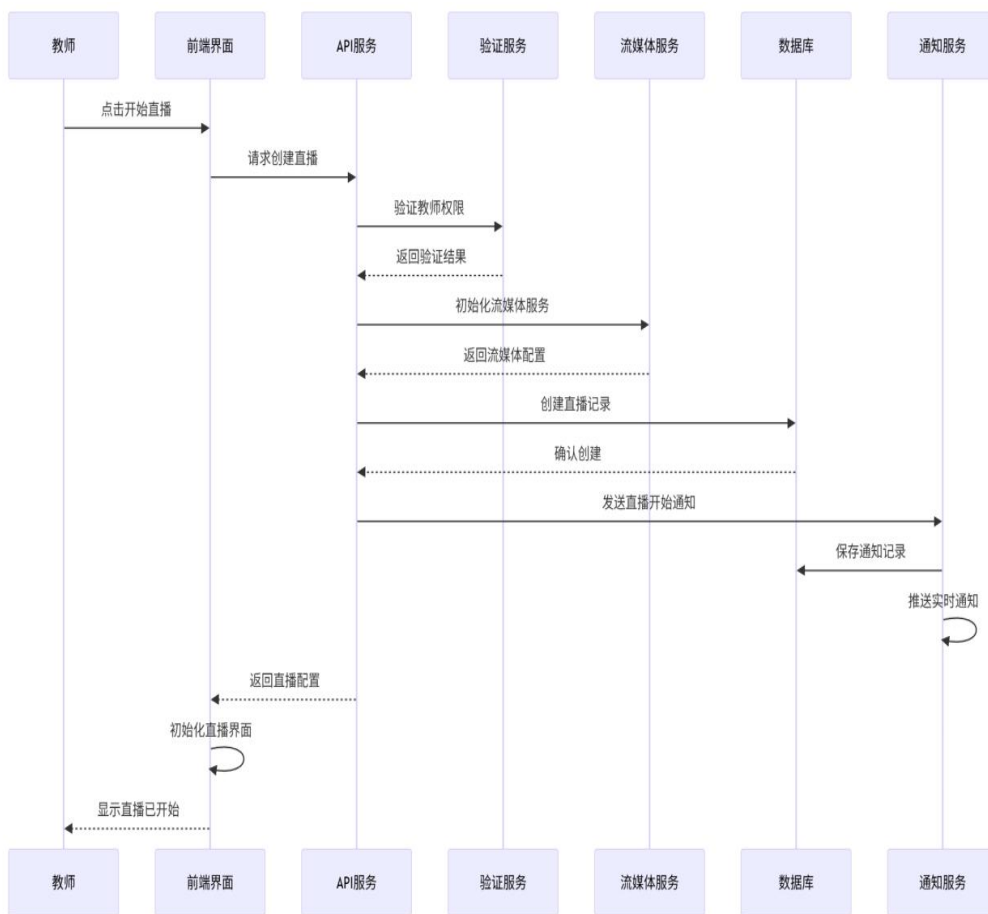


图 4-10 直播功能时序图

4.6 用户管理功能

用户管理模块是网课系统的核心控制中心，为系统管理员提供全面的管理工具和权限控制。管理员通过专属登录界面进行身份验证，可以查看所有注册用户的基本信息和活动状态，审核新用户注册申请，确认身份信息，管理用户权限，如升级为教师；管理课程分类和标签，优化课程结构；审核用户评论，过滤垃圾信息和违规内容。维护系统更新和版本控制。管理员可以对模块中的数据进行增加、查看、修改和删除等操作，允许管理员对系统中的数据进行维护和更新。管理员可以登录到选择管理模块，对数据进行增删改查，最后记录操作并结束管理活动。如图 4-11 所示。

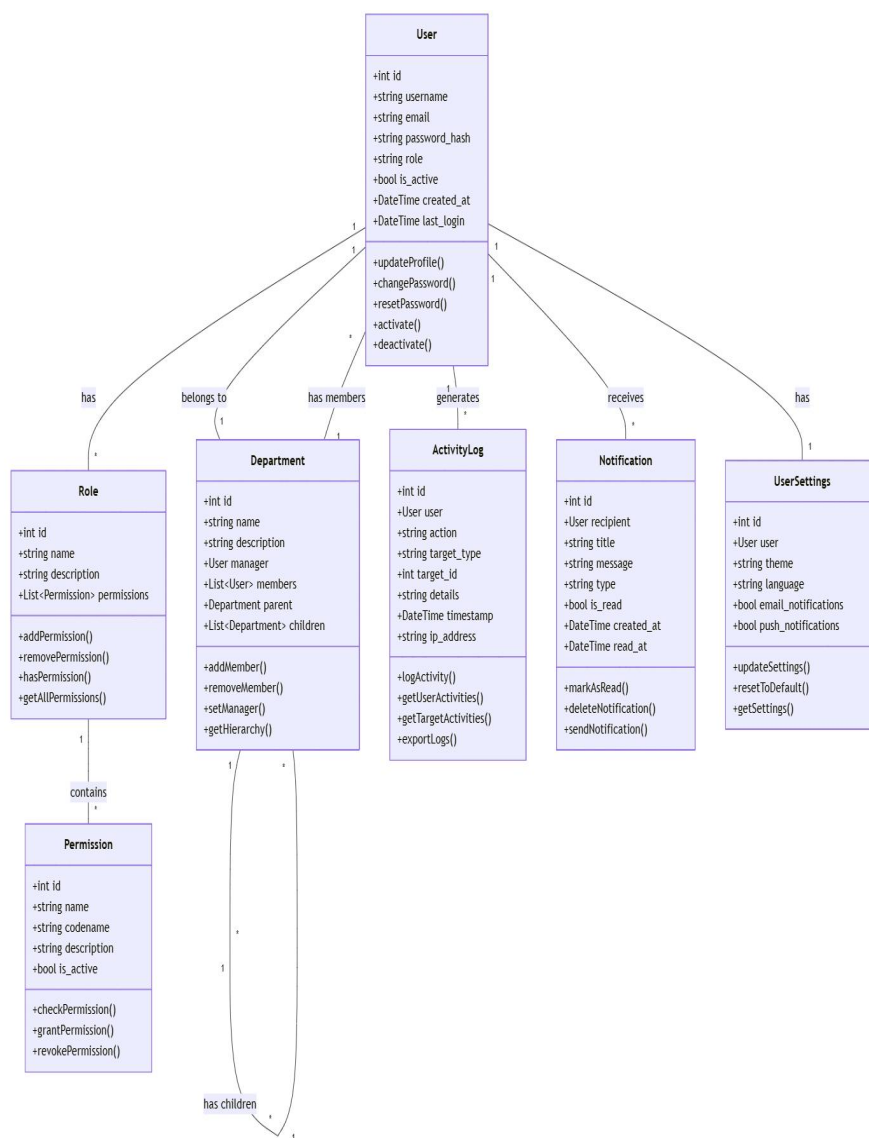


图 4-11 后端人员管理类图

管理员通过后端界面进入用户管理模块，系统向数据库发送请求以获取当前用户列表。数据库根据请求查询用户数据，并将查询到的用户列表返回给系统。系统将用户列表展示给管理员。管理员填写新用户的信息并通过系统界面提交，系统将新用户数据保存到数据库。数据库确认保存并记录操作日志，系统显示添加结果。管理员选择一个用户并修改其信息，系统将更新后的用户数据保存到数据库。数据库确认更新并记录操作日志，系统显示修改结果。管理员选择一个用户进行删除操作，系统从数据库中删除该用户数据。数据库确认删除并记录操作日志，系统显示删除结果。如图 4-12 所示。

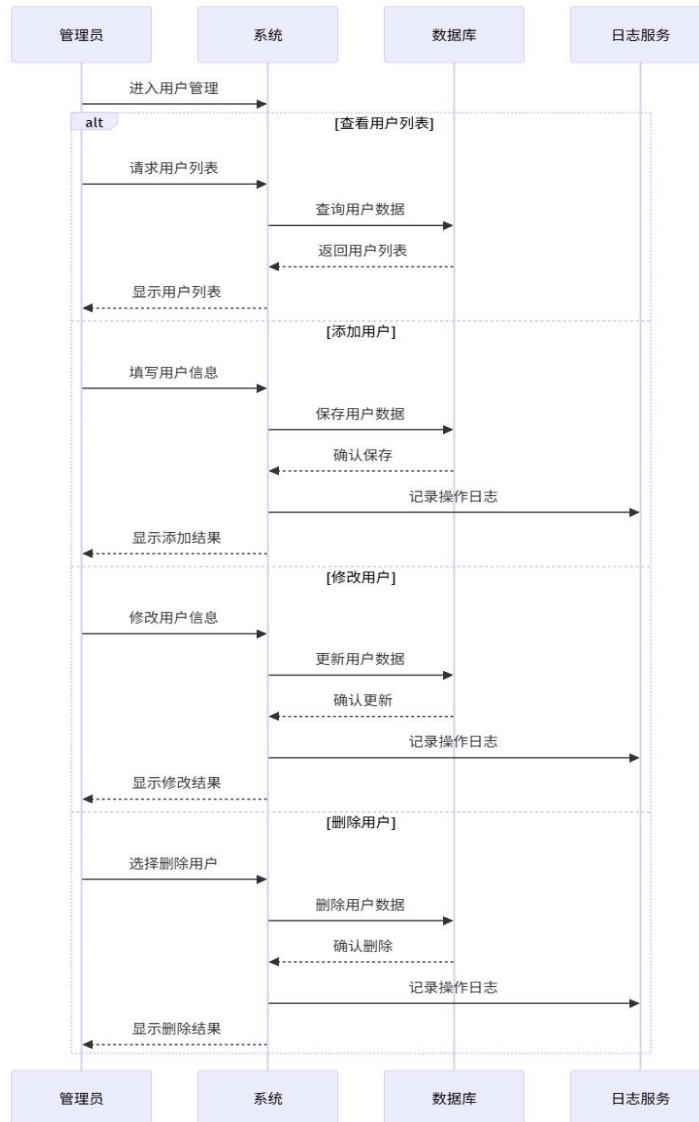


图 4-12 后端人员管理时序图

4.7 数据库设计

4.7.1 数据库概念结构设计

本系统共有 Django 默认表、管理员、用户、课程、视频等 24 个表，表之间的关系如下：管理员（通过 Django admin）对用户表进行管理用户表存储教师和学生信息，包括基本信息和权限设置，教师可以创建和管理课程，课程表存储课程基本信息，包括标题、描述、封面图等，管理员在后台对课程表进行管理，用户可以对课程发表评论，用户可以对课程进行评分，学生可以浏览课程列表，教师可以为课程创建直播

如图 4-13 所示。

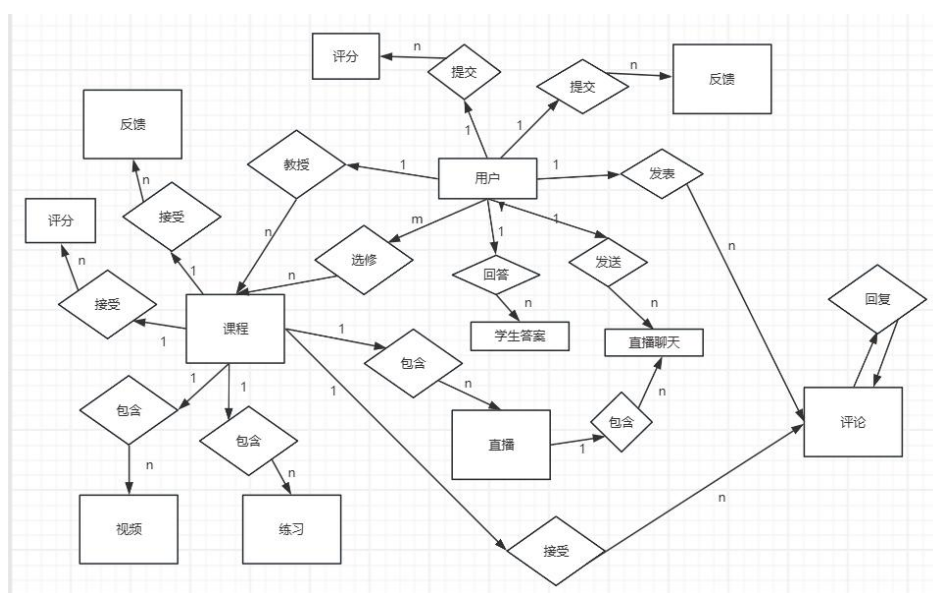


图 4-13 E-R 图

4.7.2 数据库逻辑结构设计

Users user

用户表，存储用户信息，包括用户名、密码、邮箱、头像、个人简介等，并标记用户角色（教师/学生）。

表 4-1 users user 表

序号	字段名	字段描述	数据类型	长度/限制
1	id	序号	Bigint	20
2	username	用户名	varchar	150
3	password	密码	varchar	128
4	email	邮箱	varchar	254
5	is_teacher	是否教师	tinyint	1
6	is_student	是否学生	tinyint	1
7	avatar	头像	varchar	100
8	bio	简介	longtext	0
9	created_at	创建时间	timestamp	0
10	updated_at	更新时间	timestamp	0

`courses_course`

课程主表，存储课程基本信息，包括课程标题、描述、封面图片、创建时间、更新时间、发布状态等，并关联教师和学生。如表 4-2 所示。

表 4-2 `courses_course` 表

序号	数据库字段	备注	类型	长度
1	<code>id</code>	序号	<code>Bigint</code>	20
2	<code>title</code>	课程标题	<code>varchar</code>	200
3	<code>description</code>	课程描述	<code>longtext</code>	0
4	<code>teacher_id</code>	教师 ID	<code>Bigint</code>	20
5	<code>cover_image</code>	封面图片	<code>varchar</code>	100
6	<code>is_published</code>	是否发布	<code>tinyint</code>	1
7	<code>created_at</code>	创建时间	<code>timestamp</code>	0
8	<code>updated_at</code>	更新时间	<code>timestamp</code>	0

`courses_video` 是一个用于存储课程视频信息的数据库表，用于在课程详情页面显示视频列表如表 4-3 所示。

表 4-3 `courses_video` 表

序号	数据库字段	备注	类型	长度
1	<code>id</code>	序号	<code>Bigint</code>	20
2	<code>course_id</code>	课程 ID	<code>Bigint</code>	20
3	<code>title</code>	视频标题	<code>varchar</code>	200
4	<code>description</code>	视频描述	<code>longtext</code>	0
5	<code>video_file</code>	视频文件	<code>varchar</code>	100
6	<code>duration</code>	时长	<code>int</code>	11
7	<code>order</code>	排序	<code>int</code>	11
8	<code>created_at</code>	创建时间	<code>timestamp</code>	0

`courses_comment` 表是评论表，存储用户对课程的评论，包括评论内容、父评论（用于回复功能）、创建和更新时间，关联到课程和用户

表 4-4 `courses_comment` 表

序号	数据库字段	备注	类型	长度
1	<code>id</code>	序号	<code>Bigint</code>	20
2	<code>course_id</code>	课程 ID	<code>Bigint</code>	20

续表 4-4 courses_comment 表

序号	数据库字段	备注	类型	长度
3	user_id	用户 ID	Bigint	20
4	content	评论内容	longtext	0
5	parent_id	父评论 ID	Bigint	20
6	created_at	创建时间	timestamp	0
7	updated_at	更新时间	timestamp	0

courses_rating 表评分表，存储用户对课程的评分和评价，包括评分（1-5 分）、评价内容、创建和更新时间，确保每个用户对同一课程只能评分一次。如表 4-5 所示。

表 4-5 courses_rating 表

序号	数据库字段	备注	类型	长度
1	id	序号	Bigint	20
2	course_id	课程 ID	Bigint	20
3	user_id	用户 ID	Bigint	20
4	score	评分	int	11
5	review	评价内容	longtext	0
6	created_at	创建时间	timestamp	0
7	updated_at	更新时间	timestamp	0
序号	数据库字段	备注	类型	长度

courses_feedback 表反馈表，存储用户对课程的问题反馈，包括标题、内容、状态（待处理/处理中/已解决/已关闭）、创建和更新时间。如表 4-6 所示。

表 4-6 courses_feedback 表

序号	数据库字段	备注	类型	长度
1	id	序号	Bigint	20
2	course_id	课程 ID	Bigint	20
3	user_id	用户 ID	Bigint	20
4	title	标题	varchar	200
5	content	内容	longtext	0
6	status	状态	varchar	20
7	created_at	创建时间	timestamp	0
8	updated_at	更新时间	timestamp	0

`courses_courselivestream` 表课程直播表，存储直播课程信息，包括直播标题、描述、推流密钥、推流地址、播放地址、计划开始时间、实际开始/结束时间、直播状态（计划中/直播中/已结束）等。如表 4-7 所示。

表 4-7 `courses_courselivestream` 表

序号	数据库字段	备注	类型	长度
1	<code>id</code>	序号	<code>Bigint</code>	20
2	<code>course_id</code>	课程 ID	<code>Bigint</code>	20
3	<code>title</code>	直播标题	<code>varchar</code>	200
4	<code>description</code>	直播描述	<code>longtext</code>	0
5	<code>stream_key</code>	推流密钥	<code>varchar</code>	128
6	<code>stream_url</code>	推流地址	<code>varchar</code>	200
7	<code>playback_url</code>	播放地址	<code>varchar</code>	200
8	<code>scheduled_time</code>	计划时间	<code>timestamp</code>	0
9	<code>status</code>	状态	<code>varchar</code>	20
10	<code>created_at</code>	创建时间	<code>timestamp</code>	0
11	<code>updated_at</code>	更新时间	<code>timestamp</code>	0

`exercises_exercise` 表练习题表，存储课程相关的练习题，包括标题、描述、题型（选择题/判断题/简答题）、分值、创建和更新时间。如表 4-8 所示。

表 4-8 `exercises_exercise` 表

序号	数据库字段	备注	类型	长度
1	<code>id</code>	序号	<code>Bigint</code>	20
2	<code>course_id</code>	课程 ID	<code>Bigint</code>	20
3	<code>title</code>	标题	<code>varchar</code>	200
4	<code>description</code>	描述	<code>longtext</code>	0
5	<code>exercise_type</code>	练习类型	<code>varchar</code>	20
6	<code>points</code>	分值	<code>int</code>	11
7	<code>created_at</code>	创建时间	<code>timestamp</code>	0
8	<code>updated_at</code>	更新时间	<code>timestamp</code>	0

exercises_choice 选项表，存储练习题的选择题选项，包括选项文本、是否正确，关联到具体练习题。如表 4-9 所示。

表 4-9 exercises_choice 表

序号	数据库字段	备注	类型	长度
1	id	序号	Bigint	20
2	exercise_id	练习 ID	Bigint	20
3	text	选项内容	varchar	200
4	is_correct	是否正确	tinyint	1

exercises_studentanswer 表学生答案表，记录学生的答题情况，包括选择的选项、文本答案、是否正确、提交时间，确保每个学生对同一练习题只能提交一次答案。如表 4-10 所示。

表 4-10 exercises_studentanswer 表

序号	数据库字段	备注	类型	长度
1	id	序号	Bigint	20
2	student_id	学生 ID	Bigint	20
3	exercise_id	练习 ID	Bigint	20
4	selected_choi ce	选择答案	Bigint	20
5	text_answer	文本答案	longtext	0
6	is_correct	是否正确	tinyint	1
7	submitted_at	提交时间	timestamp	0

courses_course_students 表是链接表，链接 course 表跟 user 表。如表 4-11 所示。

表 4-11 courses_course_students 表

序号	数据库字段	备注	类型	长度
1	id	序号	Bigint	20
2	course_id	课程 ID	Bigint	20
3	user_id	学生 ID	Bigint	20

users_user_groups 表是用户组表，负责用户的分组，分成教师跟学生。如表 4-12 所示。

表 4-12 users_user_groups 表

序号	数据库字段	备注	类型	长度
1	id	序号	Bigint	20
2	user_id	用户 ID	Bigint	20
3	group_id	用户组 ID	Bigint	20

users_user_permissions 表是用户权限表，里面写着用户对课程等模块的编辑、修改、查看、删除权限。如表 4-13 所示。

表 4-13 users_user_permissions 表

序号	数据库字段	备注	类型	长度
1	id	序号	Bigint	20
2	user_id	用户 ID	Bigint	20
3	permission_id	权限 ID	Bigint	20

courses_courselivechat 表直播聊天表，存储直播过程中的聊天消息，包括消息内容、是否提问、是否已回答、发送时间等，关联到直播和用户。如表 4-14 所示。

表 4-14 courses_courselivechat 表

序号	数据库字段	备注	类型	长度
1	id	序号	Bigint	20
2	live_stream_id	直播 ID	Bigint	20
3	user_id	用户 ID	Bigint	20
4	message	消息内容	longtext	0
5	is_question	是否提问	tinyint	1
6	is_answered	是否回答	tinyint	1
7	created_at	创建时间	timestamp	0

auth_group_permissions 表是 Django 后端默认的权限表，里面有管理员的权限。如表 4-15 所示。

表 4-15 auth_group_permissions 表

序号	数据库字段	备注	类型	长度
1	id	序号	Bigint	20
2	group_id	用户组 ID	Bigint	20
3	permission_id	权限 ID	Bigint	20

网课教学领养系统分为前端与后端，前端教师和学生使用，完成课程学习，课程评分，评论，试题模块，直播模块等；后台管理者使用，完成用户管理、数据管理等任务。系统用于满足当前远程教学与在线编程学习的现实需求。

本模块具体介绍了课程模块、课程评分模块、课程评论模块、直播模块以及后端管理模块的具体的流程实现以及代码逻辑的实现，4.7 章具体介绍了数据库的 E-R 图表结构以及每一个表的字段和相应的含义。

第 5 章 系统实现

5.1 课程模块系统

首先，想进入课程详情页需要进行登录
如图 5-1 所示。



图 5-1 登陆实现

本系统的课程模块提供了全面的课程管理功能，包括课程浏览、课程管理、课程学习、课程评价、课程数据统计、课程权限管理、课程推荐、课程直播、课程练习和课程通知等功能。在课程浏览方面，用户可以查看分页显示的课程列表，通过分类筛选和搜索功能快速找到所需课程，查看课程详情页中的基本信息、课程简介、教师信息等。教师可以进行课程管理，包括创建新课程、编辑课程信息、上传课程资料、发布课程公告、管理课程视频和设置课程权限。在课程内容管理方面，支持课程章节管理、视频上传与管理、课程资料上传、作业发布与管理以及课程讨论区管理。如图 5-2 所示。

课程中心

探索各种学科领域的优质课程



图 5-2 课程模块展示实现

5.2 课程评分功能展示

本系统的课程评分功能提供了全面的课程评价体系,用户可以对已学习的课程进行星级评分(1-5星)并提交文字评价,系统会自动记录评分时间和评分用户信息,同时支持用户修改自己的评分和评价内容。评分系统会对每个课程计算平均评分、评分人数和评分分布情况,并在课程详情页面展示这些统计信息。教师可以查看自己课程的所有评分和评价,了解学生的学习反馈。用户可以在课程学习完成后直接进行评分,也可以随时返回修改自己的评分,系统会自动更新评分统计信息,为课程质量提供客观的参考依据。如图 5-3 所示。

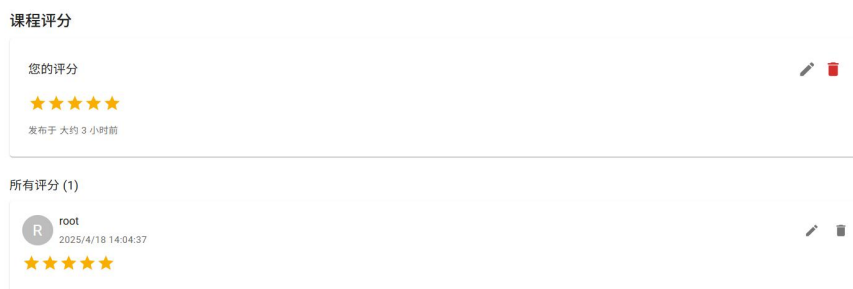


图 5-3 评分功能展示实现

5.3 评论功能展示

进入课程的评论区后，学生可以查看所有评论的列表。评论列表支持按时间顺序排序，方便学生查看最新的评论。学生可以点击评论以查看详细内容，并查看其他用户对该评论的回复。如果学生想要回复某条评论，系统会要求其先登录；若未登录，系统将跳转到登录页面。教师有权限查看所有评论，并可以删除违反规定的评论。系统会自动更新评论列表，确保其中不包含已删除的评论。如图 5-4 所示。



图 5-4 评论功能展示实现

5.4 试题功能展示

学生可以浏览教师发布的各类试题和测试，根据个人需求选择特定的试题或测试进行练习。进入选定的试题界面后，学生可以阅读题目并回答相关问题，最后提交答案完成测试。答题结束后，学生可以立即查看个人得分，了解自己的学习成果和表现。教师则可以创建各类试题和测试，设计试题内容和结构，并设置评分标准和规则。如图 5-5 所示。

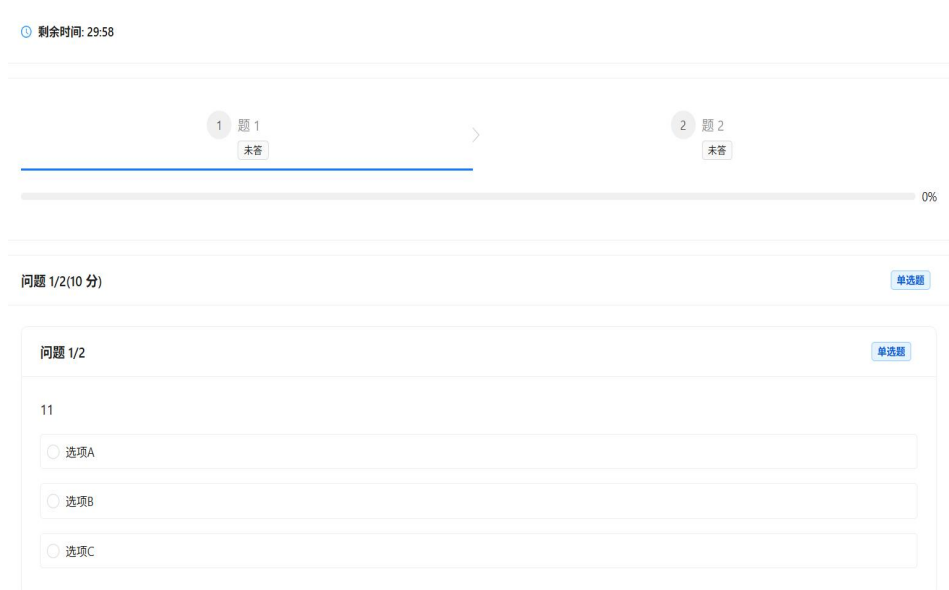


图 5-5 试题功能展示实现

5.5 直播功能展示

系统的直播功能提供了完整的在线教学直播解决方案,教师可以创建直播课程并设置直播时间、直播标题和直播简介,系统支持教师上传 obs 链接以进行直播。在直播过程中,教师可以使用白板工具进行实时板书,支持文字、图形、公式等多种教学内容的展示。学生可以通过课程页面进入直播间,观看高清直播画面,系统支持自动适应网络状况,确保直播画面的流畅性。整个直播过程支持多端同步观看,包括电脑、手机、平板等设备,确保学生可以随时随地参与直播课程学习。如图 5-6 所示。

参与实时直播课程,与讲师和其他学生进行互动学习,提升学习效果。



图 5-6 直播功能展示实现

5.6 用户管理功能

管理员通过专属登录界面进行身份验证后,可以查看所有注册用户的基本信息和活动状态,审核新用户的注册申请并确认其身份信息,管理用户权限,例如将用户升级为教师。管理员还可以管理课程分类和标签,以优化课程结构,并审核用户评论,过滤垃圾信息和违规内容。此外,管理员负责维护系统更新和版本控制,确保系统的稳定性和安全性。如图 5-7 所示。

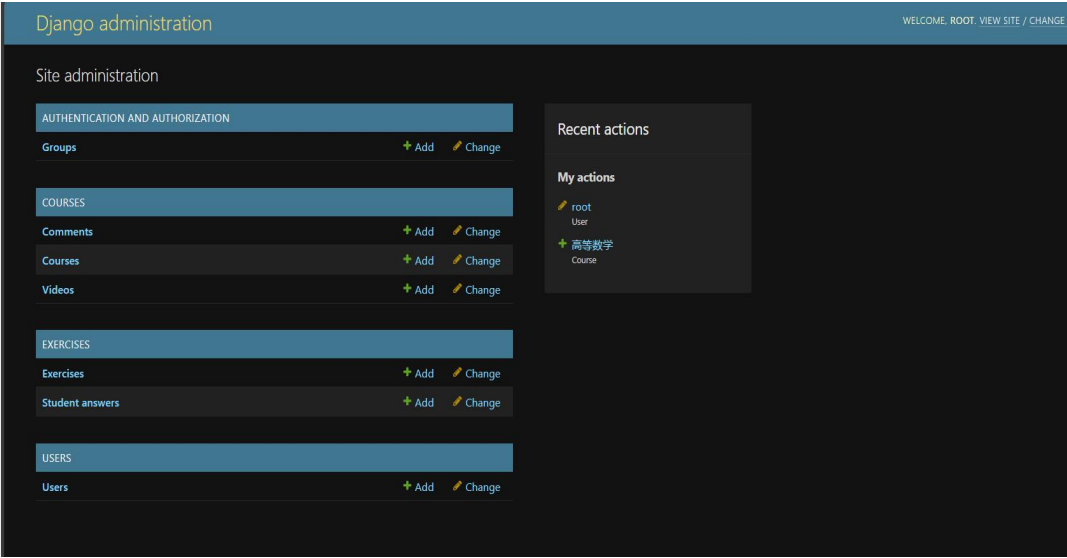


图 5-7 用户管理系统实现

本模块主要展示项目的模块的系统实现,展示了课程模块,课程评分模块,课程评论模块,课程直播模块,课程试题模块,用户管理模块功能。

第 6 章 系统测试

6.1 网课教学系统详细测试

网课系统的教学模块详细测试通常包括以下几个方面，以确保系统的功能性、性能和用户体验。

6.2 课程模块测试

测试登录、注册功能，确保用户身份验证的安全性。如表 6-1 所示。

表 6-1 登录模块测试表

序号	测试用例	预期结果	测试结果
1	用户正常注册	注册成功，返回 用户信息和 token	注册成功，返回 用户信息和 token
2	密码错误登录	提示账号或密码 错误	提示账号或密码 错误
3	用户不存在登录	提示用户不存在	提示用户不存在
4	账 号 输 入 “root”，密码空缺	“密码不能为 空”	“密码不能为 空”
5	账 号 输 入 “root”，密码输入 “980314”	跳转界面	跳转界面

此外，课程中心还为不同角色的用户提供了个性化的操作体验和功能入口：

对于教师，课程中心不仅支持课程的创建和编辑，还可以对已发布的课程进行管理，包括添加或调整课程章节、课时内容，上传或替换教学视频、课件资料，设置课程的可见范围和价格等。教师还可以查看自己课程的学生报名情况、学习进度和课程评价，及时与学生进行互动答疑，提升教学质量。对于学生，课程中心则聚合了所有可选课程，学生可以通过分类、搜索等方式快速查找感兴趣的课程。进入课程后，学生可以按照课程大纲自主学习各章节内容，观看教学视频，参与课程讨论，提交作业或测验，并实时查看自己的学习进度和成绩。系统还会根据学生的学习情况，推荐相关课程，帮助学生持续提升。

整个课程中心界面简洁直观，操作便捷，确保教师和学生都能高效地完成各自的教学与学习任务，促进了教学资源的共享和师生之间的良好互动。

如表 6-2 所示。

表 6-2 课程模块测试表

序号	测试用例	预期结果	测试结果
1	创建新课程	课程保存成功	成功
2	修改标题、描述等基本信息	课程信息更新成功	成功
3	浏览并选择课程	成功加入课程学习	成功
4	浏览多页课程	正确显示分页内容	成功
5	上传课程视频	视频上传成功并显示在课程中	成功

6.3 评分模块测试

本系统的课程评分功能提供了一个全面的评价体系，用户可以对已学习的课程进行 1 到 5 星的评分，并提交文字评价。系统会自动记录评分的时间和用户信息，同时允许用户修改自己的评分和评价内容。评分系统会为每个课程计算平均评分、评分人数和评分分布，并在课程详情页面展示这些统计信息。教师可以查看自己课程的所有评分和评价，以了解学生的学习反馈。用户可以在完成课程学习后立即进行评分，也可以随时返回修改自己的评分。如图 6-3 所示。

表 6-3 评分模块测试表

序号	测试用例	预期结果	测试结果
1	课程评分	评分和评价保存成功	成功
2	修改评分	评分和评价更新成功	成功
3	查看课程评分统计	显示评分统计信息	成功
4	发表课程反馈	反馈显示在详情页	成功
5	反馈删除	删除成功	成功

6.4 评论模块测试

进入课程的评论区后，学生可以查看所有评论的列表，并按时间顺序排序，以便轻松查看最新评论。学生可以点击评论查看详细内容，以及其他用户对该评论的回复。如果学生想要回复某条评论，系统会要求其先登录；若未登录，系统将自动跳转到登录页面。教师则拥有查看所有评论的权限，并可以删除任何违反规定的评论。如表 6-4 所示。

表 6-4 评论模块测试表

序号	测试用例	预期结果	测试结果
1	发表课程评论	评论显示在课程 评论区	成功
2	回复评论	回复显示在对应评论 下	成功
3	教师删除评论	删除成功	成功
4	管理员在后端进 行评论的删除跟修改	评论内容删除修 改成功	成功
6	管理员进入后台 系统，点击“查看评 论”	在管理员端显示 全部评论	成功

6.5 试题模块测试

此外，教师还可以对已发布的试题和测试进行管理和维护，包括修改题目内容、调整分值、更新答案解析等。教师能够实时查看学生的答题情况和成绩分布，针对学生的薄弱环节进行针对性讲解和辅导。系统还支持教师对主观题进行人工批改，完善自动评分的不足，确保评分的公平与准确。对于学生而言，系统不仅提供了丰富的练习资源，还支持多次尝试同一测试，帮助学生查漏补缺、巩固知识。学生可以在答题后查看详细的答案解析，了解每道题的解题思路和知识点归纳，提升自主学习能力。系统还会记录学生的答题历史和成绩变化，生成学习报告，帮助学生全面评估自己的学习进展。

通过试题模块的完善设计，教师与学生之间实现了高效的教学互动，极大地提升了在线学习的效果和体验。如表 6-5 所示。

表 6-5 试题模块测试表

序号	测试用例	预期结果	测试结果
1	学生提交答案	系统成功接收答案并显示提交成功的消息	成功
2	教师创建试题	新试题成功保存并显示在试题列表中	成功
3	教师删除评论	删除成功	成功
4	验证教师是否能够批改学生提交的试卷	批改结果成功保存，学生能够查看到更新后的得分	成功

6.6 直播模块测试

学生在学习完课程之后，可以对已学习的课程进行评分和反馈。学生和教师都可以看到课程的评分与反馈。教师可以对反馈的不良言论进行删除，管理员可以在后台编辑课程评分与删除。如表 6-6 所示。

表 6-6 直播模块测试表

序号	测试用例	预期结果	测试结果
1	教师创建直播课程	直播课程成功创建，并显示在课程列表中	成功
2	验证教师是否能够成功启动直播。	直播成功启动，学生可以进入直播间观看	成功
3	验证学生是否能够成功观看直播课程	学生成功进入直播间，观看直播内容	成功
4	验证教师是否能够成功结束直播	直播成功结束，学生无法再进入直播间	成功

6.7 用户管理测试

用户注册后用户的信息增加到后台，由管理员进行管理。管理员在此处可以进行用户的添加、删除、编辑修改某个属性等等。管理员还可以编辑学生和教师的权限如表 6-7 所示。

表 6-7 用户管理模块测试表

序号	测试用例	预期结果	测试结果
1	验证管理员是否能够成功查看所有注册用户列表	管理员能够看到所有注册用户的基本信息和活动状态。	成功
2	验证管理员是否能够成功添加新用户	新用户成功添加，显示在用户列表中	成功
3	验证管理员是否能够成功修改用户信息	用户信息成功更新，系统显示更新成功的消息	成功
4	验证管理员是否能够成功删除用户	用户成功删除，用户列表中不再显示该用户	成功

本模块依据项目的需求分析、概要设计、详析设计进行编程后测试，发现问题后进行修改从而完成本项目。本模块对登陆测试模块、课程模块、评论模块、评价与反馈模块、用户管理模块进行测试，测试结果所有模块的所有用例均可以符合预期并成功的实现。

第 7 章 总结与展望

本次毕业设计我开发了一个基于 Django 和 React 的网课教学系统。该系统采用前后端分离的架构设计，前端使用 React 框架开发，后端采用 Django 框架实现，为用户提供了流畅的学习体验和便捷的教学管理功能。系统的主要功能包括课程管理、视频直播、在线练习等，通过信息化手段提高教学效率，优化学习体验。

在项目开发过程中，首先进行了需求分析和设计，明确了系统的功能和性能要求项目采用现代化的前后端分离架构，这种设计模式不仅提高了开发效率，也使得系统更加灵活和可维护。在前端开发中，我选择了 React 框架，它强大的组件化特性让界面开发变得更加高效和直观。后端则采用 Python 语言配合 Django 框架，这个组合不仅开发效率高，而且拥有丰富的生态系统支持。React 的组件化开发方式带来了显著的优势：代码复用性大大提高维护成本，使用户体验更加流畅。

这个项目不仅让我掌握了具体的技术知识，更重要的是培养了我解决问题的思维方式和持续学习的态度。在计算机技术这片广阔的海洋中，我现在所掌握的只是其中的一滴水。我会保持专心的学习态度，继续钻研下去。

通过这个项目，我深刻体会到技术学习是一个永无止境的过程。只有保持谦虚学习的心态，不断充实自己，才能在技术领域走得更远。在未来的工作中，我将继续保持这种学习态度，不断提升自己的专业能力。

参考文献

- [1] Dai C. Design of Mixed Teaching Scheme of Python Language Programming Course. Computer Education, 2022(1):162 – 166.
- [2] Guo F, Tuo M, Li C. Exploration on the Mixed Teaching of “Four Dimensions and Six Stages” in Python Language Programming Course. Computer Education, 2024(06): 141 – 145.
- [3] Jinli Li, Xuebo Zhang, Lin Wang. Research on the Strategy of Promoting Python Teaching by Online Practice Platform. In: Proceedings of the 2022 3rd International Conference on Modern Education and Information Management (ICMEIM 2022). Atlantis Press, 2022.
- [4] Shi Yabin. Research and analysis of Python in economics courses in the background of big data. The Frontiers of Society, Science and Technology, 2024, 6(9): 59-64.
- [5] Cai T, Hong X, Lin Z. An Analysis on the Hybrid Teaching Mode of Higher Vocational Programming Course Supported by Supercomputing – A Case Study of “Python Data Collection and Analysis.” Southern Journal of Vocational Education, 2023, 13(1): 47 – 57.
- [6] Liu Y, Yu L, Su X. Research on Blended Teaching of Python Programming under “Integrating Virtue into Teaching.” Foreign Trade, 2023(6): 127 – 130.
- [7] Yan Ting. Construction of University Cloud Classroom Platform Based on Python. In: Proceedings of the 2022 3rd International Conference on Artificial Intelligence and Education (IC-ICAIE 2022). Atlantis Press, 2022
- [8] Bowang Liu, Tingting Xu. Python Learning Platform on the Web Service Side. Journal of Big Data and Cloud Computing, 2023.
- [9] Jinli Li, Xuebo Zhang, Lin Wang. Research on the Strategy of Promoting Python Teaching by Online Practice Platform. ResearchGate, 2022
- [10] Ting Yan. Construction of University Cloud Classroom Platform Based on Python. In: Proceedings of the 2022 3rd International Conference on Artificial Intelligence and Education (IC-ICAIE 2022). Atlantis Press, 2022
- [11] Huang T, Song T. Software Engineering Basic Training MOOC/SPOC Hybrid Teaching Model for Comprehensive Ability Cultivation. Computer Education, 2020(1): 123 – 126.
- [12] 蒋卫祥, 王雨萱, 朱利华. 高职 Java Web 应用开发课程模块化教学设计与实践[J].

山西青年, 2022, (09): 61-63.

- [13] 王梦晓, 翟玉庆, 朱林, 弭娜. 基于 Web 的 Java 融合课程教学平台的构建[J]. 信息与电脑(理论版), 2022, 34(15): 176-178
- [14] 郑海燕. 基于 Java Web 的高校英语线上教学平台设计[J]. 自动化与仪器仪表, 2023, (02): 204-208.
- [15] 孙自立. Python 语言视域下网络爬虫系统开发研究[J]. 软件, 2022, 43(03): 109-111.
- [16] 沈杰. 基于 Python 的数据分析可视化研究与实现[J]. 科技资讯, 2023, 21 (02): 14-17+54.
- [17] 陈佳佳, 邱晓荣, 熊宇昊, 段莉华. 基于 Python 的人脸识别技术研究[J]. 电脑知识与技术, 2023, 19 (08): 34-36+39.
- [18] 程俊英. 基于 Python 语言的数据分析处理研究[J]. 电子技术与软件工程, 2022, No.233(15): 236-239.
- [19] 朱向阳. 高中信息技术 python 项目式教学路径分析[J]. 高考, 2023, (24): 126-128.
- [20] 王雄伟, 侯海珍. 大数据专业 Python 程序设计课程建设探究[J]. 知识窗(教师版), 2023, (10): 117-119.

致 谢

时光荏苒，岁月如梭，四年的大学生活即将画上圆满的句号，而我的本科毕业论文也已接近尾声。此刻，站在人生新的起点上，回首过往这段充满挑战与收获的旅程，内心满是感慨与感恩。

我要向我的导师石晔琼教授致以最诚挚的感谢。在论文写作的整个过程中，导师给予了我无微不至的关怀与悉心指导。从选题的迷茫到研究方向的明确，从资料收集的无从下手到论文框架的逐步搭建，再到最终成稿的反复修改，导师始终耐心地为我的答疑解惑。他严谨的治学态度、渊博的专业知识以及对学术的执着追求，深深感染着我，成为我不断前行的动力源泉。在与导师的交流中，我不仅学到了专业知识，更学会了如何思考、如何研究，这些宝贵的经验将让我受益终生。

感谢哈尔滨师范大学软件工程专业的所有老师们。你们用丰富的教学经验和幽默风趣的授课方式，为我打开了软件工程知识的大门。在课堂上，你们深入浅出地讲解专业知识，激发了我对软件工程领域的浓厚兴趣；在实践教学中，你们耐心地指导我们完成一个个项目，培养了我们的动手能力和团队协作精神。你们的辛勤付出，让我在大学期间打下了坚实的专业基础，为我今后的发展奠定了基石。

感谢我的同学们，感谢你们与我一同度过了这段难忘的大学时光。在学习上，我们相互鼓励、相互帮助，共同攻克了一个又一个难题；在生活中，我们分享快乐、分担忧愁，彼此陪伴着成长。在论文写作过程中，我们还经常聚在一起讨论问题，交流思路，正是这种思想的碰撞，让我不断拓宽视野，完善自己的研究。感谢你们给予我的支持与陪伴，让我的大学生活充满了温暖与欢乐。

感谢我的家人，感谢你们一直以来对我的支持与鼓励。在我遇到困难和挫折时，是你们给予我力量，让我重新振作；在我取得成绩时，是你们为我骄傲，让我更加自信。你们是我永远的避风港，无论我走到哪里，你们的爱与支持都将是我最坚实的后盾。

最后，感谢哈尔滨师范大学，感谢这片充满知识与梦想的土地。在这里，我度过了人生中最美好的四年时光，收获了知识、友谊和成长。学校优美的校园环境、丰富的教学资源以及良好的学术氛围，为我提供了良好的学习和成长环境。我将永远铭记在哈师大的这段美好时光，无论未来身在何处，我都会以哈师大人自居，努力拼搏，为母校争光。

在今后的学习和工作中，我将带着这份感恩与收获，用所学知识回报社会，回报那些曾经帮助过我的人。再次感谢所有关心、支持和帮助我的人，你们是我人生中最重要的一部分，谢谢你们！