存档资料

成绩：

华 东 交 通 大 学

实 训 报 告

所属课程名称 计算机综合实训

题　　目 学院生活平台

院 （系） 信息与软件工程学院

班　　级 _____ - _____

学　　生 _____ _____

指导教师 罗志强、孙洪波、喻佳、黄晓辉

辅导教师 _____

实训(论文)任务书

天佑 学 院 **计算机科学与技术** 专 业 **2023-1** 班

一、实习实训课程设计(论文)题目 学院生活平台

二、实习实训课程设计(论文)工作自 2026 年 5 月 18 日起至 2026 年 6 月 18 日止。

三、实习实训课程设计(论文) 地点: 26-612，图书馆

四、实习实训课程设计(论文)内容要求：

1．本课程的目的

1）熟悉当前主流开发技术，综合应用基本概念、基本原理、新技术与理念开展基于应用场景的软件开发实践。

2）能够依据项目需求完成项目的需求分析、原型设计、模块划分、组件设计、接口联调与项目集成，形成较完整的开发方案。

3）理解 AI 辅助编程在需求分析、代码生成、测试调试、文档整理中的应用边界，具备结果审查、风险识别、信息安全与工程伦理意识。

4）通过团队协作与自主学习方法，完成综合项目开发、测试优化、实训报告以及 PPT 撰写、项目展示与答辩，提升沟通表达与项目管理能力。

2．实习实训课程的任务及要求

1）基本要求：

（1）能够整理、分析和解释实验数据，并通过对比实验结果，运用学科理论进行解释，说明解决方案的可行性与有效性；

（2）能合理评估计算机及相关领域的复杂工程实践对社会、健康、安全、法律、文化的影响，以及这些制约因素对项目实施的影响，并理解应承担的责任；

（3）理解诚实公正、诚信守则的工程职业道德和规范，并能在工程实践中自觉遵守；理解计算机工程师对公众的安全、健康和福祉，以及环境保护的社会责任，能够在工程实践中自觉履行责任；

（4）能够在多学科环境下，在计算机及相关领域的复杂工程实践中，正确运用工程管理与经济决策方法。

2）创新要求：

项目增加了较完整的审核反馈闭环，支持预约、资源投稿、图书上传、失物招领等业务的状态跟踪、审核意见记录、驳回修改与日志追溯。技术实现上结合阿里云 OSS 文件存储、Docker 容器化部署、PostgreSQL 数据持久化、角色权限控制和功能房预约冲突检测，提高了系统的工程化程度、部署一致性和业务处理规范性。

3）实习实训课程设计论文编写要求

（1）要按照书稿的规格打印誊写课程设计论文；

（2）论文包括任务书、目录、绪论、正文、小结、参考文献等；

（3）课程设计论文装订按学校的统一要求完成。

4）评分标准：

（1）设计过程（20 分）

（2）调试运行（20 分）

（3）项目答辩（30 分）

（4）论文写作（30 分）

5）参考文献：

[1] Martin R C. Clean code: a handbook of agile software craftsmanship[M]. Upper Saddle River: Prentice Hall, 2008.

[2] Sommerville I. Software engineering[M]. 10th ed. Boston: Pearson, 2015.

[3] Pressman R S, Maxim B R. Software engineering: a practitioner’s approach[M]. 9th ed. New York: McGraw-Hill Education, 2019.

[4] Spring. Spring Boot Reference Documentation[EB/OL]. [2026-06-15]. https://docs.spring.io/spring-boot/

[5] MyBatis. MyBatis 3 User Guide[EB/OL]. [2026-06-15]. https://mybatis.org/mybatis-3/

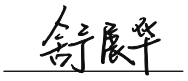
[6] Vue.js Team. Vue.js Guide[EB/OL]. [2026-06-15]. https://vuejs.org/guide/

6）综合实训进度进度安排

内容	天数	地点
构思及收集资料	2	图书馆
熟悉软件与仿真设计	21	26-612
撰写论文	2	图书馆

7）诚信申明

本人郑重声明：所呈交的课程设计报告是我个人独立完成的，如在文中涉及抄袭或剽窃行为，本人愿承担由此而造成的一切后果及责任。

学生签名：

2026 年 6 月 30 日

课程设计(论文)评价标准

序号	考核内容	得分
1	设计过程（20 分）	
2	调试运行（20 分）	
3	项目答辩（30 分）	
4	论文写作（30 分）	
总　得　分		

评阅人：_____

2026 年 7 月 1 日

目 录

第一章 设计目的 1
1.1 实训背景与选题来源 1
1.2 项目建设目标 1
1.3 技术路线与开发环境 2
1.4 课程目标对应关系 2
1.5 报告组织结构 3
第二章 需求分析 4
2.1 业务现状与问题分析 4
2.2 用户角色分析 5
2.3 功能需求分析 5
2.3.1 通知公告模块 5
2.3.2 问卷表单模块 6
2.3.3 失物招领模块 6
2.3.4 功能房预约模块 6
2.3.5 课程资源分享模块 6
2.3.6 数字图书馆模块 7
2.4 功能需求汇总 7
2.5 非功能需求分析 8
2.5.1 安全性需求 8
2.5.2 可靠性与一致性需求 8
2.5.3 性能与可扩展性需求 8
2.5.4 易用性需求 8
2.5.5 社会、健康、安全与法律约束需求 9

2.6 数据需求分析 9
2.7 可行性分析 10
2.7.1 技术可行性 10
2.7.2 经济可行性 11
2.7.3 操作可行性 11
2.7.4 管理可行性 11
第三章 概要设计 12
3.1 总体架构设计 12
3.2 功能模块划分 14
3.3 前后端交互设计 15
3.4 权限与数据范围设计 15
3.5 数据库概念结构设计 16
3.6 主要业务流程设计 18
3.7 异常处理与审计设计 19
3.8 系统部署与运行结构 19
第四章 详细设计与关键实现 20
4.1 后端分层设计 20
4.2 前端页面与接口封装设计 21
4.3 数据库逻辑结构设计 21
4.4 核心业务模块实现 24
4.4.1 通知公告与问卷表单实现 24
4.4.2 功能房预约实现 25
4.4.3 课程资源分享实现 25
4.4.4 数字图书馆实现 25
4.4.5 失物招领与 OSS 上传实现 25

4.5 权限认证与日志审计设计 26
4.6 社会、健康、安全、法律因素考虑与功能实现 27
4.7 关键算法与事务控制 27
第五章 系统测试与运行结果 29
5.1 测试环境与方法 29
5.2 功能测试结果 30
5.3 权限与安全测试 32
5.4 社会、健康、安全、法律功能验证 33
5.5 运行结果分析 34
第六章 项目管理与实训总结 35
6.1 开发过程与任务分解 35
6.2 项目成果总结 36
6.3 不足与改进方向 36
6.4 实训体会 36
参考文献 37

第一章　设计目的

1.1　实训背景与选题来源

《计算机综合实训》要求学生围绕一个完整度较高的软件项目，完成需求分析、总体设计、详细设计、编码实现、测试验证和文档撰写。课程更看重真实业务抽象、工程实现和规范意识。

本项目选题为“学院生活平台”。通知公告、活动报名、课程资料、图书资源、功能房预约和失物招领都很常见，但如果长期依赖群聊、线下登记或临时表格，信息查找和过程追踪都不方便。本实训将这些事务放到同一套认证、权限、审核和数据管理体系中处理。

项目以 Spring Boot3 和 Vue3 为主要技术基础，使用 PostgreSQL 保存业务数据，使用阿里云 OSS 保存文件资产。报告在已有学院生活平台项目基础上整理业务逻辑，重点说明需求、设计、实现、测试和责任约束。

1.2　项目建设目标

本项目的目标是把校园信息发布、资源共享和公共空间使用等事务搬到线上，并保留审核、权限、统计和日志记录。普通用户要能方便办理事务，管理人员也要能看清状态和责任记录。

项目建设目标如表 1-1 所示。

表 1-1　项目建设目标

目标类别	目标说明
门户入口	普通用户登录后可以查看公告、填写问卷、发布或检索失物招领信息、提交功能房预约、浏览课程资源和访问数字图书馆。
后台管理	管理端提供公告发布、问卷管理、资源审核、图书审核、预约审核、封禁管理、回收站处理、角色授权和日志查询等功能。
权限边界	系统通过角色、菜单、按钮和接口权限控制不同用户的操作范围，使专业负责人、图书管理员、学生工作者和管理员承担不同职责。
过程追踪	对资源投稿、图书上传、下载、预约审核、公告阅读等操作保留状态和日志，为后续统计、追责和优化提供依据。
工程责任	按课程要求检查社会、健康、安全、法律因素，并将项目中已经实现的社会、安全、法律约束落实到校园互助、认证授权、隐私保护、内容审核和下架机制中。

1.3 技术路线与开发环境

系统采用前后端分离架构。后端以 Spring Boot3 为主体，按 Controller、Service、Mapper 和 Domain/VO 分层组织；前端以 Vue3 实现路由页面、表单、列表、审核弹窗和接口封装；数据层使用 PostgreSQL，文件资源保存到阿里云 OSS，并在数据库中记录地址、哈希、大小和审计信息。

本项目采用的主要技术与工具如表 1-2 所示。

表 1-2 项目主要技术与工具

技术或工具	用途说明
Spring Boot3	构建后端 Web 服务、业务接口、认证授权和事务处理能力。
Vue3	构建前端单页应用，实现门户端和管理端用户界面。
PostgreSQL	保存用户、角色、公告、问卷、预约、资源、图书、失物信息等业务数据。
阿里云 OSS	保存课程资源、图书资产、公告附件和失物图片等文件。
Maven	管理后端多模块工程依赖、编译和打包流程。
Vite	管理前端开发服务、构建与静态资源打包流程。
Apifox/浏览器	用于接口联调、功能测试和运行结果验证。

1.4 课程目标对应关系

本项目对应课程教学大纲中的四类课程目标，关系如表 1-3 所示。

表 1-3 课程目标与项目工作的对应关系

课程目标	项目对应工作
目标 1	采用 Spring Boot3、Vue3、PostgreSQL 和阿里云 OSS 完成 Web 应用开发，综合使用后端接口、前端页面、数据库和文件存储技术。
目标 2	按照需求分析、总体设计、数据库设计、详细设计和功能实现的流程推进项目，完成软件设计说明。
目标 3	在认证授权、日志审计、隐私保护、内容审核和公共空间安全使用等方面引入约束机制，回应信息安全、工程伦理和社会责任要求。
目标 4	通过模块拆分、接口联调、功能测试、报告撰写和答辩准备完成实训交付。

从课程目标看，本项目不能只停留在页面可运行，还要把调研、设计、编码、测试和成果汇报连起来。Web 技术应用对应目标 1，需求与设计过程对应目标 2，认证授权、

日志记录、隐私控制和资源审核对应目标 3，联调测试、报告撰写和答辩准备对应目标 4。

1.5 报告组织结构

后续章节依次说明需求分析、概要设计、详细设计与关键实现、测试结果和实训总结，并在设计与测试章节中单独说明社会、安全、法律相关约束的落实情况。

第二章　需求分析

需求分析是后续设计和实现的依据。本项目面向学院日常服务场景，用户包括普通学生、教师和承担管理职责的学生工作者、专业负责人、图书管理员、系统管理员。这里需要说明做哪些功能、谁可以操作、数据怎样流转，以及系统运行时需要遵守哪些边界。

2.1　业务现状与问题分析

学院日常事务多而分散，通知公告、问卷、课程资料、数字图书、功能房预约和失物招领都和学生生活直接相关。若长期依赖群聊、线下登记或临时表格，信息查找、权限控制和过程追踪都会变得麻烦。用户端常见的问题是入口分散，管理端常见的问题是审核口径不统一、统计不方便。

结合项目场景，现有业务痛点与平台应对方式如表 2-1 所示。

表 2-1　业务痛点与系统应对方式

业务痛点	系统应对方式
信息入口分散	将公告、问卷、资源、图书、预约和失物招领纳入统一门户，用户登录后可以在同一平台完成查询、提交和查看处理结果。
权限边界不清	通过角色、菜单、按钮和接口权限限定操作范围，使普通用户、专业负责人、图书管理员和管理员承担不同职责。
审核缺少留痕	对公告发布、资源审核、图书上下架、预约审批、失物下架等操作保存状态、处理人、处理时间和原因说明。
公共资源易冲突	功能房预约在提交和审核阶段检查时间区间、房间状态、申请人封禁状态和使用时长，减少重复占用。
统计分析不足	对公告阅读、问卷答卷、资源下载、图书下载、功能房使用时长和用户贡献情况保留统计字段或日志记录。

因此，系统需要把分散事务整理成可提交、可审核、可回溯、可统计的流程。

2.2 用户角色分析

系统采用多角色协作方式，不同角色拥有不同的门户端和后台权限。角色划分如表 2-2 所示。

表 2-2 系统用户角色与职责

角色	主要职责与权限
普通用户	使用门户功能，查看公告，填写问卷，发布失物招领信息，提交功能房预约，上传或下载课程资源，上传、收藏或下载数字图书。普通用户只能管理本人提交的资源、图书、预约和失物信息。
学生工作者	在普通用户能力基础上，负责通知公告发布、撤回、置顶和阅读情况查看；在功能房预约模块中可参与预约审核和违规申请人封禁处理。
图书管理员	负责数字图书馆的图书审核、上下架、删除、下载统计查看和图书管理员映射管理，使电子图书资源按平台规则发布。
专业负责人	负责本专业课程与课程资源管理，审核本专业用户提交的课程资料，对优质资料标记为最佳推荐，并维护本专业资源质量。
管理员	对各业务模块拥有管理能力，包括公告、问卷、功能房、课程资源、数字图书、失物招领等模块的数据维护、审核和统计。
超级管理员	拥有系统级配置和用户权限管理能力，可维护用户、角色、菜单、模块开关和高权限操作，是平台最高管理角色。

角色权限遵循最小必要授权。普通用户只操作本人数据，专业负责人处理本专业资源，图书管理员处理数字图书馆数据，管理员和超级管理员负责跨模块维护。

2.3 功能需求分析

系统功能分为门户端和管理端。门户端面向普通用户，处理查询、提交和分享；管理端面向授权角色，处理审核、维护和统计。两端共享同一套身份、权限和业务数据。

2.3.1 通知公告模块

通知公告模块用于发布学院通知、公告和附件材料。发布者可以设置标题、内容、附件、有效期和可见范围，范围可以面向全员，也可以按角色、部门或岗位定向发布。用户查看详情后，系统记录阅读回执，同一用户对同一公告只保留一条记录。

管理端需要支持草稿、发布、撤回、置顶、取消置顶和删除。附件通过阿里云 OSS 保存，数据库记录文件名、地址、类型、大小和排序信息。门户端支持关键词、状态和已读未读筛选。

2.3.2 问卷表单模块

问卷表单模块用于收集信息和意见。管理人员创建问卷时，需要设置标题、题目、题型、必填项、选项、截止时间和可见范围。题型以文本题、单选题和多选题为主。问卷发布后，用户可以填写并提交；截止前允许修改本人答卷。

系统保证同一用户对同一问卷只有一份有效答卷。选择题可以汇总票数，文本题可以查看明细。可见范围沿用角色、部门和岗位三类维度。

2.3.3 失物招领模块

失物招领模块统一管理“丢失”和“捡到”两类信息。用户发布时填写标题、描述、地点、时间、联系方式，并可上传图片。新增或编辑后的条目先进入待审核状态，审核通过后再公开。条目解决后，作者可以标记为已解决。

该模块需要控制图片数量、格式和大小，图片同样保存到阿里云 OSS。管理员驳回或下架条目时必须填写原因，用户可以据此修改后重新提交。解决状态确认后不再反复回退。

2.3.4 功能房预约模块

功能房预约模块负责会议室、研讨室、活动室等公共空间的线上申请。用户可以按楼房和楼层浏览功能房，查看容量、设备标签、启用状态和预约时间轴。提交预约时，需要填写申请人、使用人、用途、开始时间和结束时间。系统会检查房间是否启用、申请人是否被封禁、使用人数是否符合要求、预约时长是否超限，以及时间区间是否冲突。

预约支持审核开关。审核开启时，新预约进入待审核状态；审核关闭时，规则通过后直接批准。被驳回的预约可以修改后重新提交，待审核或未开始的预约允许取消。管理端还需要提供封禁和时长统计。

2.3.5 课程资源分享模块

课程资源分享模块围绕“专业—课程—资源”组织学习资料。用户可以上传压缩包或外链资源，并填写名称、课程、专业和简介。资源提交后进入待审核状态，审核通过后才可见并可下载。被驳回或下架时，系统保存原因。

文件型资源记录文件哈希，外链资源做地址规范化并校验唯一性。系统还通过下载计数、最佳推荐、用户积分和排行榜鼓励资料共享。资源创建、编辑、审核、下架、下载和删除都会写入操作日志。

2.3.6 数字图书馆模块

数字图书馆模块用于电子图书和学习资料的上传、审核、发布、收藏和下载。用户上传图书时需要填写书名、作者、ISBN、出版社、出版年份、简介、封面和文件资产。图书采用 ISBN-13 做唯一性校验，可关联多个文件或外链资产。图书管理员和管理员负责审核、驳回、下架、重新上架和删除。

门户端提供列表、详情、检索、收藏和下载入口。下载时系统更新次数并写入下载日志，记录用户、图书、资产、处理结果和访问来源。

2.4 功能需求汇总

为便于后续设计与测试，主要功能需求汇总如表 2-3 所示。

表 2-3 主要功能需求汇总

模块	功能需求摘要
通知公告	公告发布、撤回、置顶、附件、有效期、定向可见、阅读回执、已读未读筛选。
问卷表单	问卷创建、题目配置、可见范围、截止时间、用户提交、重复提交控制、提交结果汇总。
失物招领	丢失/捡到信息发布、图片上传、审核、驳回、下架、恢复、已解决标记。
功能房预约	楼房房间管理、时间轴、预约申请、冲突检测、审核开关、封禁管理、排行榜。
课程资源分享	专业课程管理、资源上传、审核发布、最佳推荐、下载计数、积分与排行榜、审计日志。
数字图书馆	图书上传、ISBN 去重、多格式资产、审核上下架、收藏、下载统计、图书管理员管理。
系统基础能力	登录认证、角色权限、菜单权限、按钮权限、用户管理、操作日志、文件上传网关。

2.5 非功能需求分析

2.5.1 安全性需求

系统所有涉及写入、修改、删除、审核和下载计数的接口均需要登录认证，并根据角色权限进行访问控制。后台按钮权限要与后端接口权限标识保持一致，不能只在前端隐藏按钮而放开接口访问。用户密码应加密存储，阿里云 OSS AccessKey 等敏感凭据不能写入代码仓库，应通过环境变量或安全配置注入。

文件上传需要前后端共同校验。课程资源限制为压缩包或外链，数字图书限制常见电子书格式，失物招领限制图片格式和单张图片大小，公告附件限制常见文档、图片和压缩包。前端负责提示，后端负责拦截。

2.5.2 可靠性与一致性需求

业务状态必须前后一致。功能房预约提交时，房间状态、封禁状态、时间冲突检查和预约写入应放在同一事务内；课程资源和数字图书的审核、下架、重新上架需要同步更新状态和日志；问卷提交需要保证一个用户对一份问卷只保留一份有效答卷。

对容易重复或冲突的数据，系统同时使用业务校验和数据库约束。课程资源通过文件哈希或外链地址去重，数字图书通过 ISBN-13 去重，功能房预约通过时间区间冲突检测减少重复占用。

2.5.3 性能与可扩展性需求

系统列表查询都应支持分页，避免一次性加载大量数据。公告、问卷、图书、课程资源和失物招领需要提供关键词、状态和时间范围筛选。功能房时间轴查询应围绕房间和时间窗口建立索引；下载榜、积分榜可以先按统计字段或日志聚合实现。

各业务模块需要保持相对独立。通知公告、问卷、失物招领、功能房预约、课程资源和数字图书馆分别拥有自己的业务表和服务接口，同时复用认证、权限、文件上传和审计字段等公共能力。

2.5.4 易用性需求

门户端应让普通用户少走步骤。课程资源可以按专业和课程浏览，功能房详情页可以直接发起预约，数字图书馆支持关键词搜索，失物招领支持按状态筛选。管理端则需要提供状态标签、审核按钮、驳回理由输入框和批量操作入口。

2.5.5 社会、健康、安全与法律约束需求

本节按课程要求检查外部约束，但只列出已有功能能承接的内容。本项目能明确落到需求中的主要是社会、安全和法律三类，见表 2-4。

表 2-4 社会、安全与法律约束需求

因素	需求落实方式
社会因素	失物招领减少物品找回成本；课程资源分享和数字图书馆支持学习资料共建；通知公告和问卷表单让学院事务处理更集中。
安全因素	登录认证、角色权限、接口权限、文件校验、封禁机制、预约冲突检测、操作日志和下载日志共同保障系统运行安全。
法律因素	密码加密、最小化联系方式采集、可见范围控制、资源审核、违规内容下架、下载留痕和 OSS 凭据保护用于降低隐私、版权和违规传播风险。

后续设计只围绕这些可验证功能展开。未单独实现的因素不在表中扩展。

2.6 数据需求分析

系统数据包括用户权限、内容发布、问卷收集、空间预约、资源共享、图书资产和日志统计。主要数据对象如表 2-5 所示。

表 2-5　主要数据对象

数据对象	数据内容
用户与权限	用户、角色、菜单、部门、岗位、用户角色关系、角色菜单关系等，用于认证和授权。
通知公告	公告主体、可见范围、附件、阅读回执、发布状态、置顶状态和有效期。
问卷表单	问卷主体、可见范围、题目、选项、答卷和答案明细。
失物招领	失物/拾物主体、图片、联系方式、审核状态、解决状态、驳回和下架原因。
功能房预约	楼房、功能房、预约记录、使用人关联、审核设置、申请人封禁和排行榜统计。
课程资源	专业、课程、专业负责人、资源主体、最佳推荐、资源操作日志、用户积分和积分流水。
数字图书	图书主体、图书资产、收藏关系、下载日志、操作日志和图书管理员映射。
文件资产	OSS 文件地址、对象键、文件大小、文件哈希、文件格式和关联业务记录。
审计日志	创建人、创建时间、更新人、更新时间、操作人、操作动作、处理结果和原因说明。

核心业务表需要保留主键、必要索引、审计字段和软删除字段。审核类业务保存状态、审核人、审核时间和原因说明；文件型业务记录文件地址、大小、哈希和对象键；资源、图书、预约和问卷等容易重复的场景，通过唯一约束、事务校验或幂等提交控制数据质量。

2.7　可行性分析

2.7.1　技术可行性

本系统采用 Spring Boot3 与 Vue3 构建前后端分离应用，可以满足接口开发、页面交互、状态管理和权限控制需求。PostgreSQL 适合保存公告、问卷、预约、资源、图书和日志等结构化数据；阿里云 OSS 负责文件资产存储，避免把大文件放入应用服务器或数据库。结合已有项目基础，技术上可完成实训演示。

2.7.2 经济可行性

本项目主要用于课程实训和本地演示，开发过程以开源技术、已有项目代码和个人测试环境为基础。数据库和后端服务可以部署在本地或云服务器，阿里云 OSS 只保存少量测试文件，成本可控。

2.7.3 操作可行性

系统采用 Web 方式访问，普通用户可以通过浏览器完成查看、提交、下载和预约。管理端使用表格、表单、弹窗和状态标签组织业务流程，符合常见后台系统的操作习惯。不同角色登录后看到的菜单和按钮不同，可以减少误操作。

2.7.4 管理可行性

系统将多类校园事务纳入统一认证、权限和审计框架。学生工作者负责公告与预约审核，专业负责人负责课程资源质量控制，图书管理员负责数字图书审核，管理员负责跨模块维护。该分工符合学院日常管理方式，也能保留必要的责任记录。

第三章　概要设计

本章在需求分析基础上说明系统结构、模块边界、权限模型、数据概念结构和主要业务流程。概要设计不展开代码细节，重点说明各部分如何协作。

3.1　总体架构设计

系统采用前后端分离的 Web 架构。前端由 Vue3 实现门户端和管理端页面，负责路由、表单、列表筛选、状态展示和接口调用；后端由 Spring Boot3 提供业务接口，负责认证授权、参数校验、事务处理、文件上传和日志记录；PostgreSQL 保存结构化数据，阿里云 OSS 保存附件、图片、课程资源和图书资产。浏览器只与后端接口交互，文件访问地址由后端统一生成和记录。

系统架构层次如表 3-1 所示。

表 3-1　系统总体架构层次

架构层次	设计说明
用户访问层	用户通过浏览器访问门户端或管理端。普通用户主要使用门户功能，授权角色进入管理端处理审核和维护事务。
前端表现层	Vue3 页面负责菜单路由、列表查询、详情展示、表单提交、弹窗审核、状态标签和接口封装。前端只做必要的交互校验，最终业务规则以后端为准。
接口控制层	Controller 接收前端请求，完成登录态检查、权限注解校验、参数绑定和统一返回。门户接口与管理接口分开组织，便于控制使用场景。
业务服务层	Service 承担主要业务规则，包括公告发布、问卷提交、预约冲突检查、资源审核、图书下载记录、失物状态流转等。涉及多表写入的操作放入事务中处理。
数据访问层	Mapper 负责访问 PostgreSQL，完成分页查询、条件筛选、状态更新、统计聚合和日志写入。高频查询字段设置索引，减少列表和时间轴查询压力。
文件存储层	阿里云 OSS 保存公告附件、失物图片、课程资源压缩包、图书封面和图书文件。数据库只保存文件地址、对象键、大小、格式和哈希等元数据。

系统总体架构如图 3-1 所示。浏览器侧负责页面交互，业务规则、权限边界和文件访问由后端处理，结构化数据和文件资产分别保存到 PostgreSQL 与阿里云 OSS。

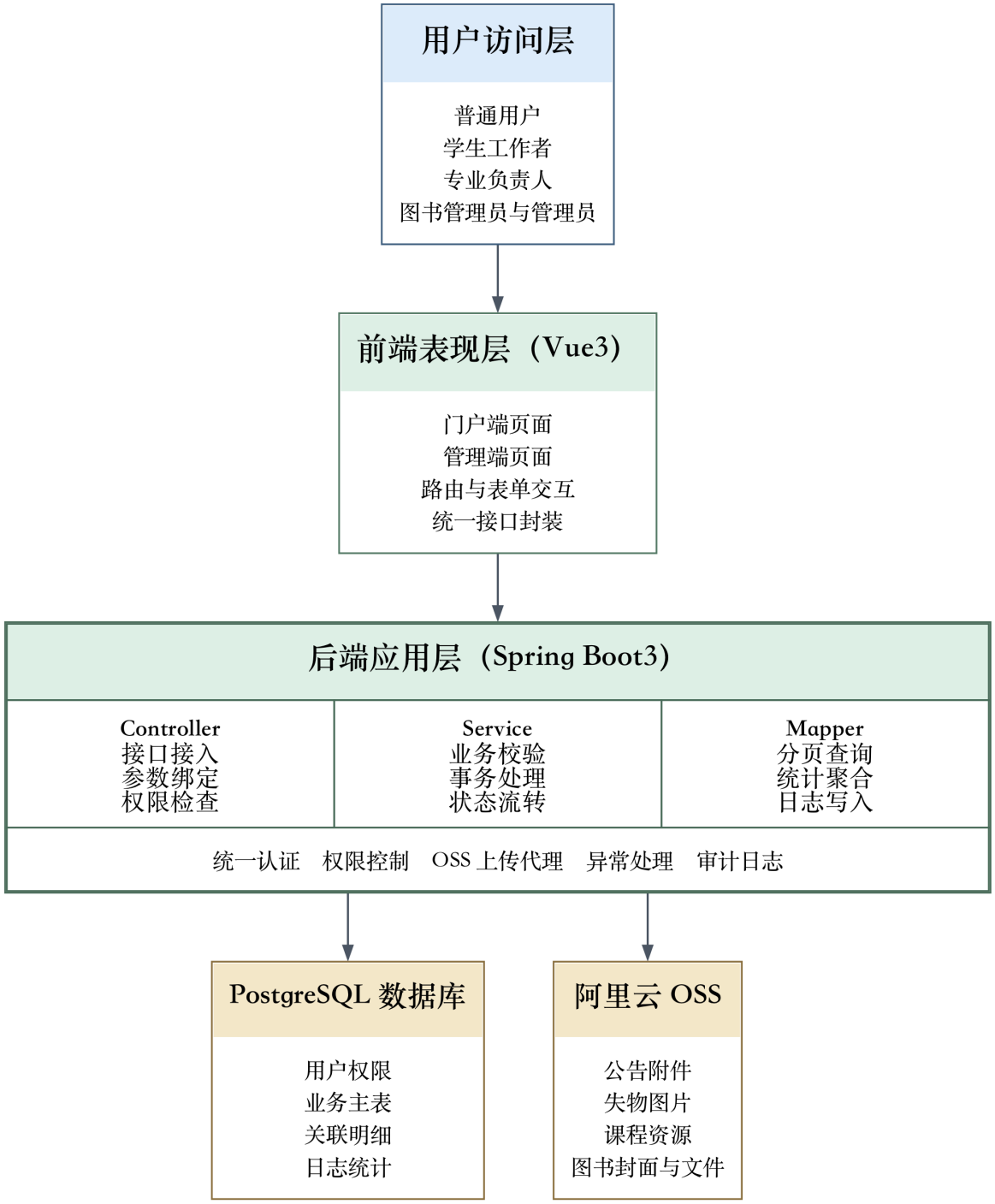


图 3-1 系统总体架构图

这种分层让职责更容易划分。前端不直接接触数据库和 OSS 凭据，后端处理业务规则和安全边界，数据库保存结构化事实，文件存储从应用服务器中独立出来。

3.2 功能模块划分

按照使用对象和业务性质，系统分为门户服务、后台管理和公共支撑三类模块。门户服务处理用户日常操作，后台管理处理审核和维护，公共支撑提供认证、权限、上传和日志能力。模块划分如表 3-2 所示。

表 3-2 系统功能模块划分

模块类别	模块设计
门户服务模块	包括通知公告浏览、问卷填写、失物招领发布与检索、功能房预约、课程资源浏览与下载、数字图书检索与收藏。门户页面面向普通用户，重点是入口清楚、反馈及时。
后台管理模块	包括公告管理、问卷管理、失物审核、楼房房间管理、预约审核、封禁管理、专业课程管理、课程资源审核、图书审核、图书管理员维护和排行榜统计。后台页面面向授权角色，重点是状态处理和责任留痕。
公共支撑模块	包括登录认证、用户角色、菜单权限、按钮权限、文件上传网关、操作日志、登录日志和基础配置。业务模块不重复实现这些能力，而是通过统一机制复用。

图 3-2 展示模块归属关系。门户端承接用户日常办理，后台管理端处理审核和维护，公共支撑模块为两端提供认证、权限、文件和日志能力。

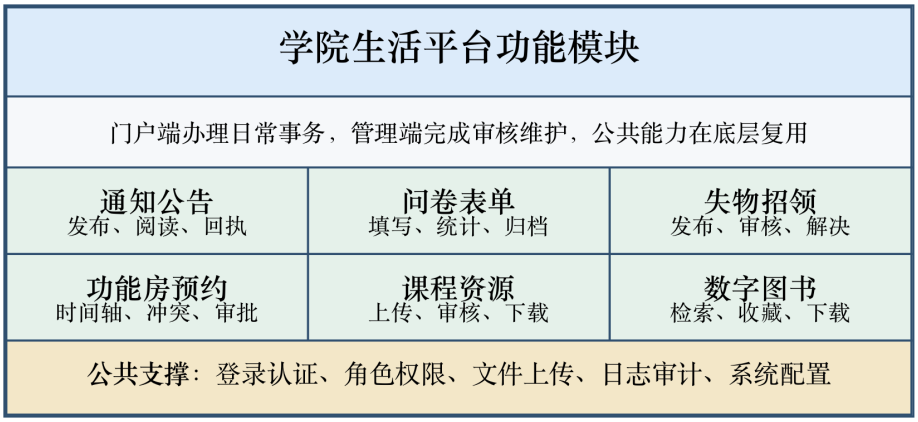


图 3-2 系统功能模块结构图

各业务模块相对独立，但共用一些规则。通知公告和问卷都使用可见范围控制；课程资源和数字图书馆都依赖文件上传、审核发布和下载日志；功能房预约和失物招领都需要状态流转、驳回原因和回收处理。公共支撑模块负责承接这些共性能力。

3.3 前后端交互设计

前端按“门户端”和“管理端”组织页面。门户端展示已发布、已审核或本人相关的数据；管理端面向授权角色，展示待审核、已发布、回收站和统计榜单等视图。页面通过统一请求工具调用后端接口，前端根据返回结果展示提示信息。

接口设计上，门户接口与管理接口采用不同路径前缀。门户接口主要用于查看公开数据、提交本人申请或管理本人数据；管理接口检查更细的权限标识。以课程资源为例，门户侧提供专业、课程、资源列表和下载接口，管理侧提供新增、编辑、审核、驳回、下架和删除接口。

请求处理过程如表 3-3 所示。

表 3-3 前后端请求处理过程

处理阶段	设计说明
页面发起请求	用户在列表、表单或详情页面触发操作，前端整理查询条件或表单数据，并携带登录令牌调用接口。
接口权限检查	后端先检查登录状态，再根据接口权限标识判断当前用户是否可执行该操作。无权限请求直接返回错误结果。
业务规则校验	Service 检查状态、时间、归属关系、文件类型、重复提交、预约冲突等业务条件。校验失败时返回明确原因。
数据写入与日志	对需要落库的操作，后端在事务内写入主体数据、关联数据和日志记录。下载、审核、下架等操作保留处理痕迹。
前端结果反馈	前端根据返回结果刷新列表、关闭弹窗或显示错误提示，使用户知道操作是否成功以及下一步应如何处理。

3.4 权限与数据范围设计

系统权限采用“用户—角色—菜单/按钮—接口”的方式组织。用户通过角色获得菜单和按钮权限，前端控制入口和按钮显示，后端通过权限注解控制接口调用。前端隐藏按钮只能改善体验，权限判断必须由后端完成。

数据范围主要按“本人数据”和“授权模块数据”处理。普通用户管理自己提交的资源、图书、预约和失物信息；专业负责人审核本专业课程资源；图书管理员处理数字图书馆数据；学生工作者处理公告和功能房预约。管理员和超级管理员可以跨模块维护，但操作仍需留痕。

权限设计与业务数据的关系如表 3-4 所示。

表 3-4　权限与数据范围设计

场景	控制方式
门户浏览	公告、问卷根据全员、角色、部门、岗位范围判断是否可见；资源、图书和失物只展示审核通过且未下架的数据。
本人数据	普通用户只能编辑或删除本人提交且状态允许的数据，例如待审核资源、被驳回图书、本人预约和本人失物条目。
审核操作	预约审核、资源审核、图书审核和失物审核均要求管理端权限。驳回和下架需要保存原因，避免只留下状态变化。
文件操作	上传接口需要登录和权限检查，下载接口需要记录业务日志。OSS 凭据不暴露给浏览器，文件元数据由后端保存。
系统配置	用户、角色、菜单、模块开关等高权限操作只开放给管理员或超级管理员，并写入系统操作日志。

3.5　数据库概念结构设计

数据库概念结构围绕“用户身份、业务主体、关联明细、行为日志”四类对象组织。用户与权限数据提供身份基础；公告、问卷、失物、房间、资源和图书是业务主体；附件、可见范围、答卷明细、预约使用人、图书资产等是关联明细；阅读记录、下载记录、操作日志和积分流水记录行为事实。

主要实体关系如表 3-5 所示。

表 3-5　主要实体关系设计

业务域	实体关系说明
通知公告	公告主体与附件、可见范围、阅读回执是一对多关系。公告通过可见范围控制读者，通过阅读回执统计已读情况。
问卷表单	问卷主体包含多个题目，题目可以包含多个选项。用户答卷与答案明细分表保存，以适配文本题、单选题和多选题。
失物招领	失物条目与图片是一对多关系，条目状态记录待审、已发布、驳回和下架，解决状态与审核状态分开保存。
功能房预约	楼房包含多个房间，房间包含多个预约记录，预约记录再关联多个使用人。封禁信息独立保存，只限制用户作为申请人提交预约。
课程资源	专业包含课程，课程包含资源，资源关联操作日志、最佳推荐、下载统计和用户积分流水。专业负责人用于限定资源审核范围。
数字图书	图书主体关联多个图书资产、收藏记录、下载日志和操作日志。图书管理员映射表用于识别拥有审核职责的用户。

图 3-3 展示系统基础数据的核心关系。用户、部门、角色和菜单权限是各业务模块按身份访问、按范围管理的前提。更细的业务表关系放在第 4 章说明。

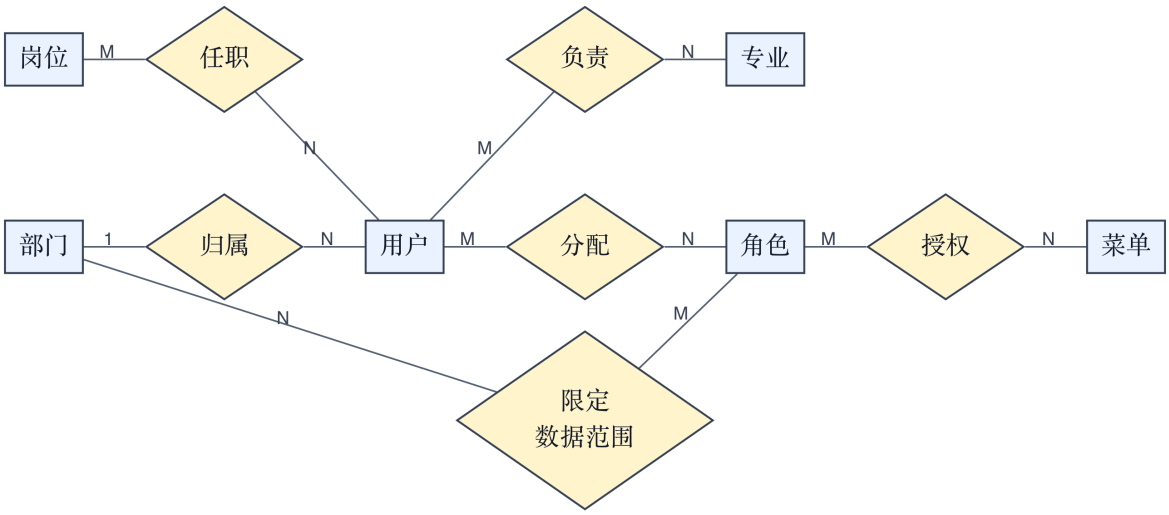


图 3-3　系统核心实体关系图

转换为关系表时，一对多关系通过外键字段或逻辑关联表达，多对多关系拆成关联表，审核和下载等事件以日志表保存。系统统一使用审计字段记录创建人、创建时间、更新人和更新时间，并通过软删除字段处理大部分业务删除。

3.6 主要业务流程设计

系统中不少模块都包含审核流。公告发布、失物审核、功能房预约、课程资源审核和数字图书审核采用相近的处理思路：先保存为草稿或待审核状态，审核通过后进入可见或可用状态，驳回、下架或撤回时保存原因。不同模块的差异主要在于可见范围、文件约束和是否允许重新提交。

典型状态流转如表 3-6 所示。

表 3-6 主要业务状态流转

业务模块	状态流转概要
通知公告	草稿经发布后对可见范围内用户展示；发布后可撤回，撤回时同步取消置顶；超过有效期后按过期数据处理。
问卷表单	草稿发布后允许可见用户填写；截止前用户可修改本人答卷；问卷归档或过期后停止提交，管理端查看汇总结果。
失物招领	新增或编辑后进入待审核；审核通过后公开展示；驳回或下架进入管理端回收视图；作者可将已发布条目标记为已解决。
功能房预约	提交后根据审核开关进入待审核或已批准；待审核可通过或驳回；未开始的待审核或已批准预约可取消；时间到达后转为进行中或已完成。
课程资源	用户上传后进入待审核；专业负责人或管理员审核通过后发布；被驳回或下架后保存原因，修改后可重新提交。
数字图书	用户上传图书后进入待审核；图书管理员或管理员审核通过后上架；驳回、下架和重新上架均记录处理人和原因。

功能房预约的状态和规则较多，图 3-4 单独给出审核路径。提交和审核通过前都要检查冲突，驳回原因和处理结果会进入日志。

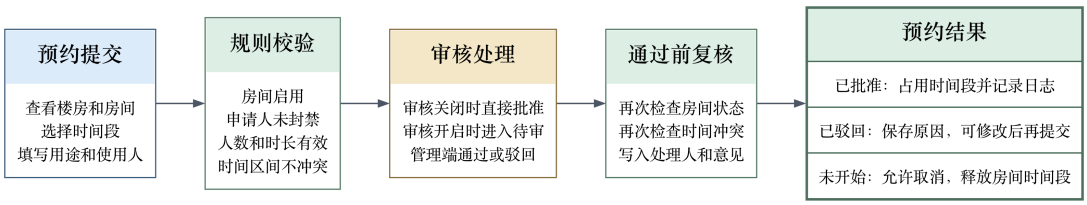


图 3-4 功能房预约审核流程图

功能房预约涉及公共空间占用和时间冲突。用户提交时，后端先检查房间状态、申请人封禁状态、使用人数和预约时长，再检查同一房间在目标时间段内是否已有占用。审核通过前还要再次检查冲突，防止相近时间提交的申请同时通过。

文件型业务也采用统一流程。用户上传附件、图片、资源压缩包或图书文件时，前端先限制格式和大小，后端再次校验并写入阿里云 OSS。业务表只保存 OSS 地址、对

象键、大小、格式和哈希。下载时，系统先判断权限，再记录下载日志并返回访问地址。

3.7 异常处理与审计设计

系统异常处理遵循“前端提示，后端裁决”的原则。表单必填项、时间格式、文件大小等可在前端先提示；状态不允许、权限不足、预约冲突、重复资源、ISBN 重复、问卷过期等业务错误由后端返回明确原因。

审计设计贯穿各业务模块。普通创建和修改通过审计字段记录，审核、驳回、下架、下载、封禁等动作写入业务日志或系统操作日志。日志保存操作人、时间、结果和原因说明，便于统计和复核。

3.8 系统部署与运行结构

系统部署采用前后端分离方式。前端构建后生成静态资源；后端 Spring Boot3 服务独立运行，对外提供门户端和管理端接口；PostgreSQL 保存结构化数据；阿里云 OSS 保存公告附件、失物图片、课程资源和图书资产。运行结构如表 3-7 所示。

表 3-7 系统部署与运行结构

运行组成	设计说明
前端静态资源	Vue3 工程构建后输出静态文件，用户通过浏览器访问门户端和管理端页面，页面通过接口地址调用后端服务。
后端应用服务	Spring Boot3 服务负责认证授权、业务接口、事务处理、日志记录和 OSS 上传代理，是系统运行的核心服务。
PostgreSQL 数据库	保存用户权限、公告、问卷、预约、资源、图书、失物招领和日志统计等结构化数据，通过索引支撑分页与条件查询。
阿里云 OSS	保存文件型资产，后端负责生成对象键、校验文件、写入文件元数据，并在下载时根据权限返回访问地址。
配置与密钥	数据库连接、OSS AccessKey、Bucket、Endpoint 等配置通过环境变量或安全配置文件注入，不进入前端代码和公开仓库。

在本地开发和实训演示场景中，前端开发服务、后端应用和 PostgreSQL 可以部署在同一台机器上，OSS 使用云端测试目录。若后续部署到服务器，前端静态资源、后端服务、数据库和文件存储可以分别扩展。

第四章　详细设计与关键实现

本章说明系统在编码层面的组织方式和主要业务规则，包括后端分层、前端页面、数据库表、文件上传、权限审计以及责任约束的实现。

4.1　后端分层设计

后端采用 Controller、Service、Mapper、Domain/VO 分层。Controller 接收请求、绑定参数、检查权限并返回统一结果；Service 处理业务校验、状态流转、事务和日志；Mapper 访问 PostgreSQL，完成查询、更新、统计和冲突检测；Domain 与 VO 分别承载数据库实体和前后端交互数据。

后端主要分层职责如表 4-1 所示。

表 4-1　后端分层职责

层次	实现职责
Controller	暴露门户接口和管理接口，使用权限注解限制管理操作，完成参数接收、分页启动和统一响应封装。
Service	集中处理业务规则，例如问卷提交校验、预约冲突判断、资源审核、图书 ISBN 校验、失物状态流转和 OSS 上传结果整理。
Mapper	通过 XML 或接口方法执行数据库访问，支持列表筛选、关联查询、下载计数、日志写入、时间轴查询和排行榜统计。
Domain/VO	Domain 对应主要业务表字段，VO 用于提交预约、创建图书、时间轴片段和排行榜等复杂交互数据。
配置与公共服务	OSS 配置、上传服务、登录态工具、权限工具、统一异常和操作日志为各业务模块提供基础能力。

涉及多表写入或状态变更的操作放在事务中处理。例如预约创建要同时写入预约主体和使用人关联，问卷提交要写入答卷主体和答案明细，课程资源审核通过后要更新状态、写入日志并发放积分。事务边界放在 Service 层，可以避免部分写入成功、部分失败。

4.2 前端页面与接口封装设计

前端页面按门户端和管理端组织。门户端负责浏览、提交和查看本人记录，管理端负责筛选、审核、状态切换和统计查看。接口封装文件与页面目录保持对应关系，公告、问卷、功能房、课程资源、数字图书和失物招领分别封装接口。

主要页面组织如表 4-2 所示。

表 4-2 前端页面组织

页面类别	页面设计
门户公告与问卷	公告列表和详情负责展示可见公告并记录阅读；问卷列表、填写页和我的答卷页负责提交、修改和回看答卷。
门户功能房	功能房列表展示楼房、楼层和房间信息，详情页展示预约时间轴并提供申请入口，我的预约页展示个人申请结果。
门户资源与图书	课程资源按专业、课程浏览并支持下载排行；数字图书馆提供列表、详情、上传、收藏、贡献记录和下载榜。
门户失物招领	列表页支持丢失和捡到信息检索，编辑页用于发布或修改条目，我的发布页展示审核和解决状态。
后台审核管理	管理端按公告、问卷、失物、功能房、课程资源和数字图书拆分页面，审核、驳回、下架等动作通过弹窗或按钮完成。

前端只做必要的交互校验，例如必填项、文件大小、时间格式和表单完整性。权限、状态、时间冲突、文件类型和重复提交等规则仍以后端校验为准。

4.3 数据库逻辑结构设计

系统数据库表按业务域划分。通知公告包含公告主体、附件、可见范围和阅读回执；问卷包含问卷主体、题目、选项、答卷和答案明细；功能房包含楼房、房间、预约、使用人、设置和封禁；课程资源包含专业、课程、资源、操作日志、积分和最佳推荐；数字图书包含图书主体、图书资产、收藏、下载日志、操作日志和图书管理员映射；失物招领包含条目主体和图片。

主要业务表关系如图 4-1 至图 4-4 所示。各图按业务域拆开绘制，避免把二十多张表挤进一张总图。

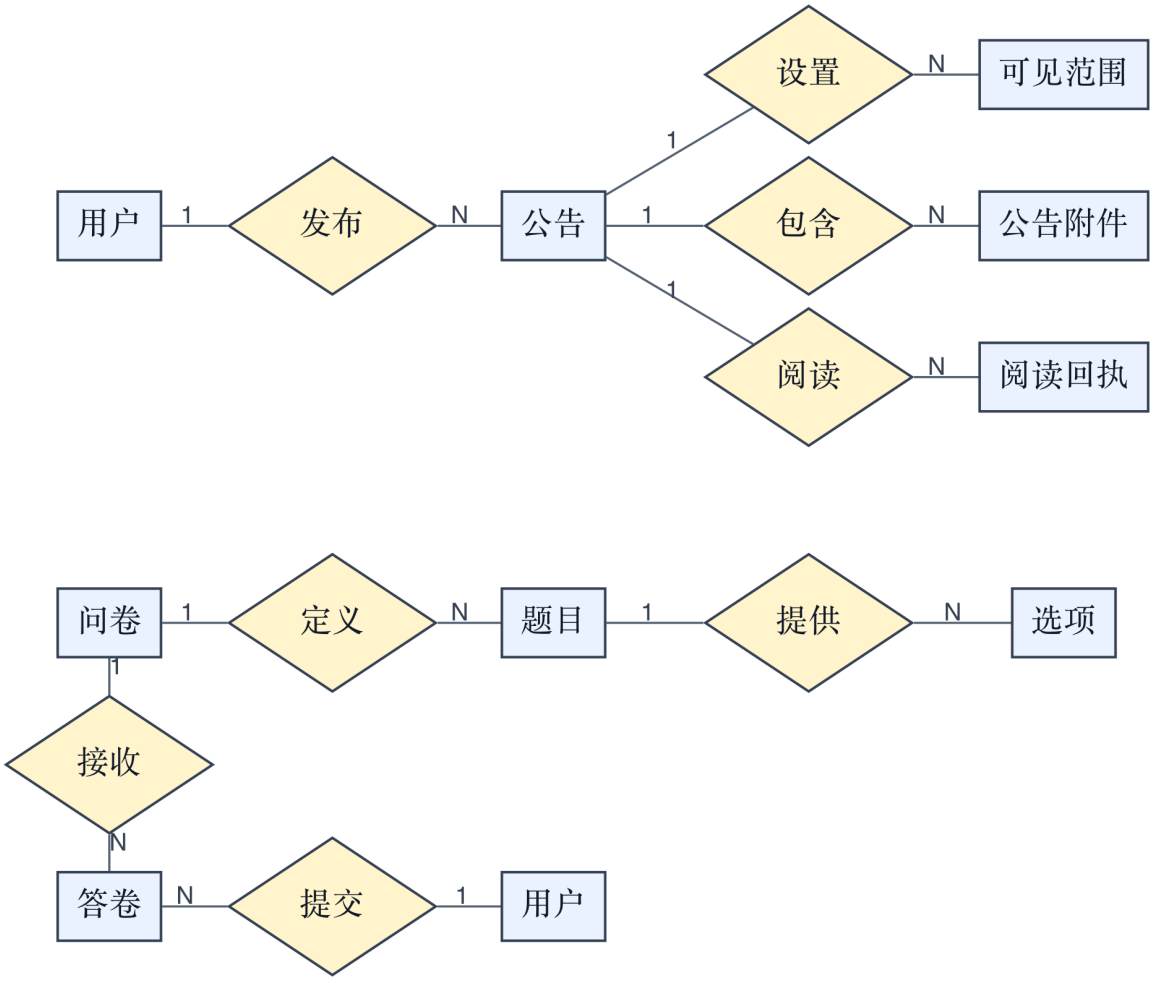


图 4-1　公告与问卷业务表 E-R 图

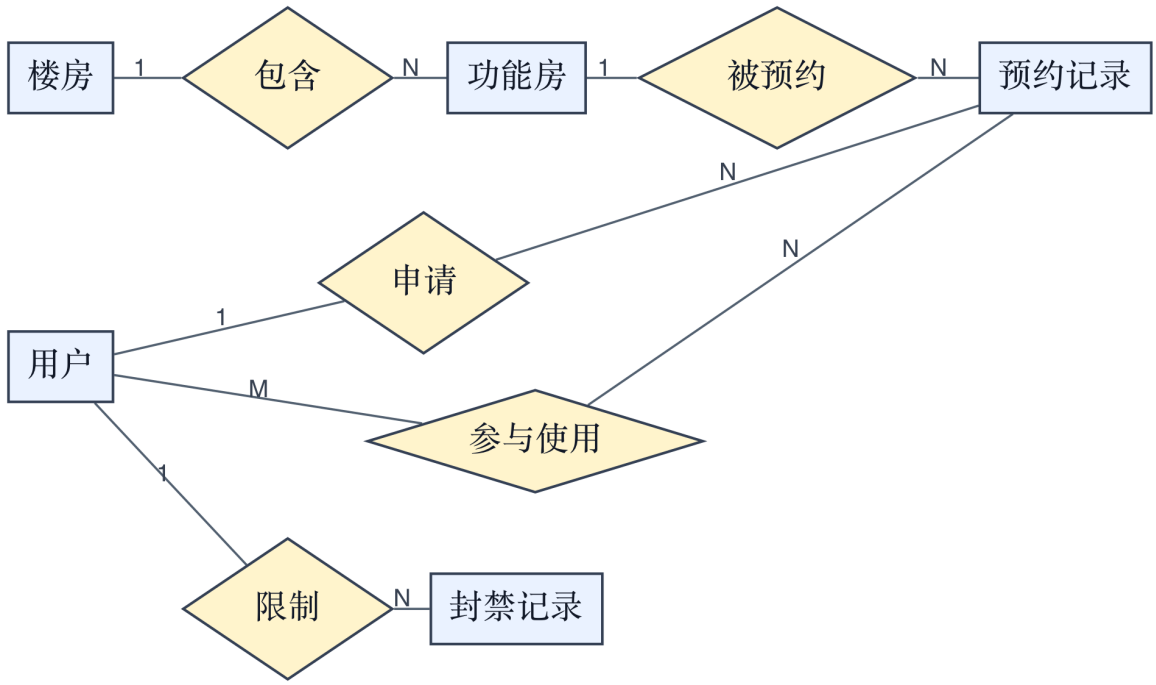


图 4-2　功能房预约业务表 E-R 图

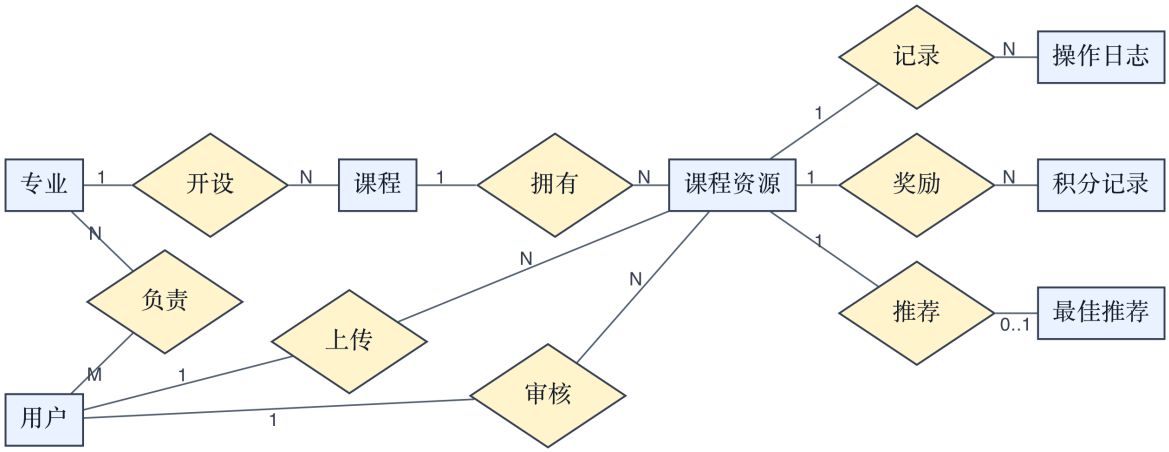


图 4-3　课程资源业务表 E-R 图

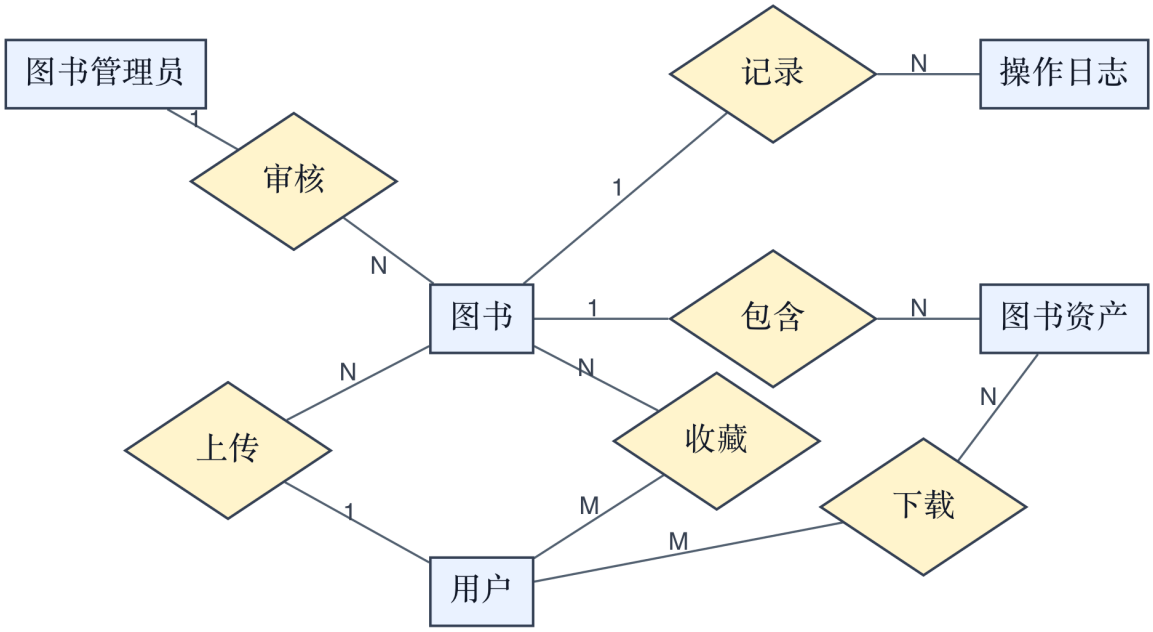


图 4-4　数字图书业务表 E-R 图

各业务表统一保留创建人、创建时间、更新人、更新时间和删除标志。审核类表保存审核人、审核时间和原因说明；下载类表保存访问用户、访问结果和请求来源；文件类表保存文件地址、对象键、格式、大小和哈希。

4.4　核心业务模块实现

4.4.1　通知公告与问卷表单实现

通知公告围绕可见范围和阅读回执实现。公告可以全员可见，也可以按角色、部门或岗位选择范围。用户查看详情时，后端读取公告、附件和范围信息，并尝试写入阅读回执。同一公告、同一用户只记录一次，只有首次写入成功时才增加阅读次数。

问卷表单围绕题目结构和唯一答卷实现。问卷保存时，题目和选项随问卷一并写入；编辑草稿时，系统重建题目与选项。用户提交答卷时，后端先检查问卷是否发布、是否过期，再按题目定义校验必填题和选择题。若用户已提交过答卷，系统更新原答卷并替换答案明细。

4.4.2 功能房预约实现

功能房预约围绕时间冲突和公共空间秩序实现。用户提交预约时，Service 依次检查房间是否存在且启用、申请人是否被封禁、使用人数是否符合要求、时间是否有效、预约时长是否超限。随后系统按房间编号和时间区间查询冲突记录，待审核、已批准和进行中的预约都视为占用。

审核通过时，后端再次检查房间状态和时间冲突。只有二次校验通过，预约才进入已批准状态。驳回必须填写原因，申请人可以修改后重新提交；被封禁用户不能作为申请人提交新预约，但仍可作为他人预约的使用人。

4.4.3 课程资源分享实现

课程资源分享模块围绕专业、课程和资源三层结构实现。用户上传资源时选择专业与课程，文件型资源保存 OSS 地址、文件大小和 SHA-256 哈希，外链型资源保存规范化后的链接地址。新增和编辑资源都会进入待审核状态，审核通过后才在门户端展示并允许下载。

专业负责人只能处理本专业资源，管理员可以跨专业处理。审核通过时，系统写入操作日志，并通过积分流水为上传者加分；最佳推荐也通过独立记录和积分规则处理。积分发放依赖唯一流水，重复审核或重复标记不会反复加分。资源下架时，系统保存原因，并自动取消最佳推荐。

4.4.4 数字图书馆实现

数字图书馆将图书主体和图书资产分开保存。图书主体保存 ISBN-13、书名、作者、出版社、出版年份、简介、封面和状态；图书资产保存文件或外链的格式、地址、大小、哈希和排序。用户上传图书时，后端会标准化并校验 ISBN-13，检查是否已有相同 ISBN 的图书。图书审核通过后才对门户用户可见。

下载时，后端先判断图书是否可访问，再增加下载次数并写入下载日志。日志记录图书、资产、用户、处理结果、IP、User-Agent 和来源页面。收藏功能使用图书和用户的关系表保存，并对重复收藏和取消收藏做幂等处理。

4.4.5 失物招领与 OSS 上传实现

失物招领将条目内容和图片分开保存。用户新增或编辑条目后进入待审核状态，审核通过后公开展示；驳回或下架时必须保存原因。图片集合采用先删后插的更新方式，保证编辑后的图片顺序和数量与数据库一致。作者标记已解决后，默认列表不再优先展示该条目。

文件上传统一走阿里云 OSS。后端根据业务目录、日期和随机 UUID 生成对象键，对扩展名和文件大小进行校验，上传时设置 Content-Type 和 Content-Disposition，并计算 SHA-256 哈希。OSS AccessKey 等敏感配置只在后端环境中读取，不写入前端代码。

4.5 权限认证与日志审计设计

系统权限认证围绕登录令牌、角色权限和接口权限展开。用户登录后，前端在后续请求中携带令牌访问接口。普通页面根据菜单和按钮权限控制入口展示，管理接口通过后端权限标识判断访问资格。

权限认证与日志审计设计如表 4-3 所示。

表 4-3 权限认证与日志审计设计

设计项	实现说明
登录认证	用户登录后由后端维护登录态和访问令牌，未登录用户不能访问需要认证的门户接口和管理接口。
密码保护	用户密码不以明文保存，系统认证过程基于加密后的密码摘要进行校验，降低账号信息泄露风险。
接口权限	管理端接口使用 @PreAuthorize 等权限校验方式限制访问，资源审核、图书审核、预约审批、系统配置等操作必须拥有对应权限标识。
按钮权限	前端根据用户权限控制审核、驳回、下架、删除、封禁等按钮显示，减少误操作入口。
数据归属	普通用户只能操作本人提交的数据，专业负责人只处理本专业资源，图书管理员只处理数字图书数据。
操作日志	审核、驳回、下架、封禁、下载等动作记录操作人、操作时间、处理结果和原因说明，用于统计与追溯。

日志审计按业务性质分层处理。系统级操作写入统一操作日志，例如用户、角色、菜单和配置维护；业务级操作写入各模块日志，例如课程资源操作日志、数字图书下载日志、公告阅读回执和功能房审核记录。不同业务保留不同字段，避免把所有日志都塞进同一张表。

资源驳回、图书下架、失物下架、预约驳回和申请人封禁都要求填写原因。原因字段可以把单纯的状态变化补充为有依据的处理记录。

4.6 社会、健康、安全、法律因素考虑与功能实现

本项目只写已经做进系统且能直接说明的内容。对应关系如表 4-4 所示。

表 4-4 社会、安全、法律因素的功能实现

因素	具体实现
社会因素	失物招领支持发布、审核、图片展示和已解决标记；课程资源和数字图书支持上传、审核、下载和贡献统计。
安全因素	登录认证、角色权限、接口权限、文件扩展名和大小校验、预约冲突检测、申请人封禁和操作日志已经实现。
法律因素	密码加密存储，OSS 凭据只在后端读取；失物招领只采集必要联系方式；课程资源和数字图书有审核、下架和下载记录。

其中，法律因素主要通过必要信息采集、审核、下架和日志记录处理。

4.7 关键算法与事务控制

本项目的业务规则不算复杂，但有几类规则必须放到后端和数据库层面处理，不能只靠页面提示。功能房预约、课程资源审核、图书下载、问卷提交和公告阅读都涉及状态一致性或幂等处理。系统将这些规则集中在 Service 层，并对多表写入开启事务。

关键算法与事务控制如表 4-5 所示。

表 4-5　关键算法与事务控制

关键规则	实现方式
预约冲突检测	功能房预约使用半开时间区间进行冲突判断，待审核、已批准和进行中的记录都视为占用；创建和审核通过时均执行冲突检查。
预约事务写入	创建预约时在同一事务中写入预约主体和使用人关联，若房间状态、封禁状态、人数或时间校验失败，则整体回滚。
阅读回执去重	公告详情访问时尝试插入阅读回执，依赖公告和用户的唯一约束保证一人一条；只有首次插入成功时才增加阅读数。
问卷答卷幂等	同一用户对同一问卷只保留一份答卷。再次提交时更新答卷主体并替换答案明细，而不是新增重复答卷。
文件哈希去重	课程资源文件上传后保存 SHA-256 哈希，同课程下相同文件会被识别为重复资源；外链资源通过规范化链接去重。
积分幂等发放	课程资源审核通过和设置最佳推荐时写入积分流水，流水唯一约束避免重复审核或重复标记造成重复加分。
ISBN 校验与唯一性	数字图书上传时先标准化并校验 ISBN-13，再检查是否存在相同 ISBN 图书，减少重复书目。
下载日志事务	图书或资源下载时同步更新下载次数并写入下载日志，保证统计数据和访问事实能够对应。

以功能房预约为例，时间冲突检测不是比较开始时间是否相同，而是判断两个时间区间是否重叠。系统在提交时检查一次，审核通过前再检查一次，避免同一房间同一时间段被重复批准。

资源与图书模块更关注去重和留痕。课程资源通过文件哈希或外链地址减少重复提交，数字图书通过 ISBN-13 减少重复书目。审核、驳回、下架、下载和积分发放都写入日志或流水表，保证状态变化有记录可查。

第五章　系统测试与运行结果

系统测试以浏览器页面操作和接口联调为主，覆盖门户端常用流程、管理端审核流程、权限边界、文件上传、预约冲突和专项责任功能。由于项目用于课程实训和演示，测试数据以本地与测试环境中的典型数据为主，没有进行大规模并发压测。

5.1　测试环境与方法

测试环境与开发演示环境保持一致。后端服务以 Spring Boot3 方式启动，前端以 Vue3 项目运行，数据库使用 PostgreSQL，文件上传接入阿里云 OSS 测试目录。浏览器用于验证页面流程，Apifox 用于验证接口参数、返回结果和异常提示。测试环境如表 5-1 所示。

表 5-1　测试环境

环境项	测试配置
后端服务	Spring Boot3 应用，本地启动管理端与门户端接口，使用统一登录认证和权限校验。
前端页面	Vue3 前端工程，通过浏览器访问门户页面和后台管理页面。
数据库	PostgreSQL 测试库，导入用户、角色、公告、问卷、功能房、课程资源、图书和失物招领等演示数据。
文件存储	阿里云 OSS 测试 Bucket，用于公告附件、失物图片、课程资源和图书文件上传验证。
测试工具	浏览器开发者工具用于观察页面请求和响应；Apifox 用于接口参数、边界条件和错误返回验证。

测试方法分为功能流程、边界异常和专项责任验证三类。功能流程从普通用户和管理角色角度测试提交、审核、查看和统计；边界异常重点检查重复提交、时间冲突、非法文件、ISBN 重复和权限不足；专项责任验证检查社会、安全、法律要求是否对应到规则限制或日志记录。

5.2 功能测试结果

功能测试覆盖六个主要业务模块和公共上传能力。若页面状态、数据库记录和接口返回结果一致，判定为通过；非法操作能返回明确提示，也视为边界测试通过。主要结果如表 5-2 所示。

表 5-2 主要功能测试结果

测试场景	验证内容与结果	结论
通知公告	管理端新建公告并发布，门户端可按可见范围查看；用户打开详情后生成阅读回执，再次打开不会重复增加阅读次数。	通过
问卷表单	管理端创建文本题、单选题和多选题问卷并发布；门户端提交答卷后可在截止前修改，数据库只保留同一用户的一份有效答卷。	通过
失物招领	用户发布丢失或捡到信息后进入待审核；管理员通过后门户展示，驳回和下架均能保存原因，作者可标记已解决。	通过
功能房预约	用户按房间和时间提交预约，合法数据进入待审核；同一房间重叠时间段会被拒绝，审核通过前系统再次检查冲突。	通过
课程资源分享	用户上传文件型资源或提交外链后进入待审核；专业负责人审核通过后门户可见，重复哈希或重复外链会被拦截。	通过
数字图书馆	用户上传图书并附带资产，ISBN-13 校验生效；审核通过后门户可下载和收藏，下载次数和下载日志同步更新。	通过
OSS 上传	合法文件可上传并返回 URL、对象键、大小、类型和 SHA-256；不允许的扩展名或超过大小限制时返回错误提示。	通过

测试结果显示，主要业务流程可以完成提交、审核、发布、查看、下载和统计。各模块都能通过状态字段、审核原因和日志记录保留处理过程。

图 5-1 至图 5-4 给出部分实机运行结果，展示预约、冲突拦截、资源审核发布和统计榜单等流程。

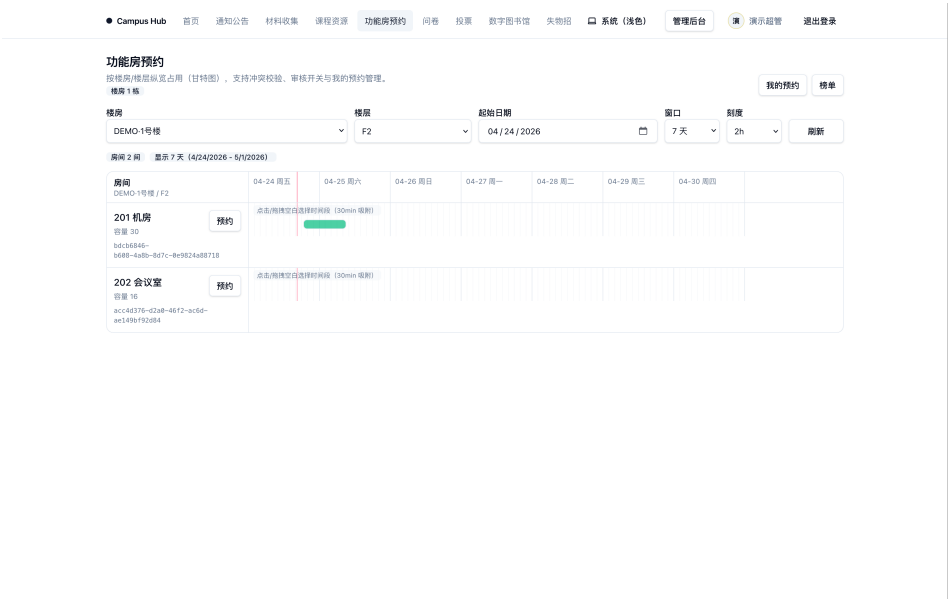


图 5-1　合法预约提交结果

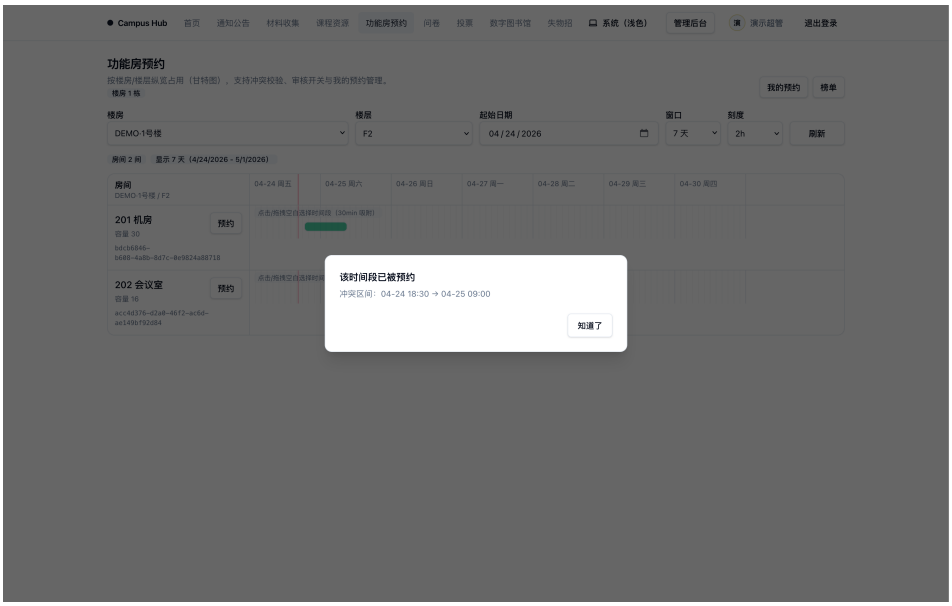


图 5-2　重叠预约被拒绝的运行结果



图 5-3　资源投稿与审核发布结果

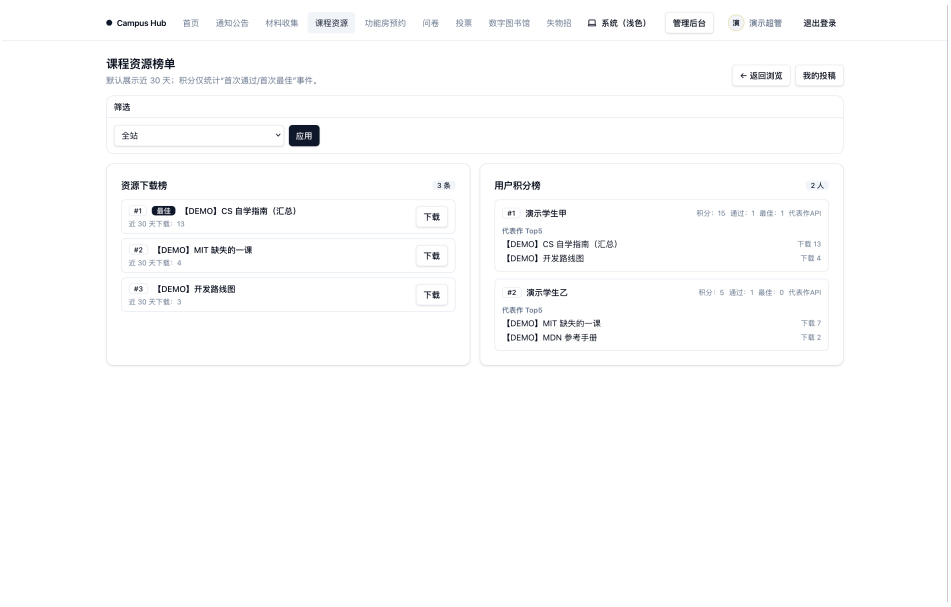


图 5-4　课程资源积分榜查询结果

5.3　权限与安全测试

权限测试重点验证前端按钮控制和后端接口拦截是否同时生效。测试时分别使用普通用户、学生工作者、专业负责人、图书管理员和管理员账号访问同类页面或接口，观察菜单、按钮和接口返回结果。测试结果如表 5-3 所示。

表 5-3　权限与安全测试结果

测试场景	验证内容与结果	结论
普通用户访问管理接口	普通用户直接调用资源审核、图书审核、预约审批等管理接口时，后端返回无权限结果，页面也不展示对应按钮。	通过
本人数据控制	普通用户只能编辑或删除本人提交且状态允许的数据，尝试修改他人资源、图书、预约或失物信息时被拒绝。	通过
专业负责人范围控制	专业负责人只能审核所负责专业下的课程资源，跨专业资源不会进入其审核范围，直接调用接口也会被拒绝。	通过
图书管理员权限控制	图书管理员可以处理数字图书审核和上下架，无法越权处理课程资源、功能房设置等无关模块。	通过
文件上传安全	上传接口需要登录和上传权限，非法扩展名、空文件和超出限制的文件不会写入 OSS。	通过
敏感配置保护	OSS AccessKey 不出现在前端代码和报告提交文件中，上传由后端服务统一代理完成。	通过

权限测试说明，系统没有只依赖前端菜单隐藏来限制操作。即使绕过页面直接调用接口，后端仍会根据登录态和权限标识做判断。

5.4　社会、健康、安全、法律功能验证

本节只检查已经落地且可以直接验证的功能。项目中能明确对应的是社会、安全和法律三类：社会因素对应失物招领和资源共享，安全因素对应权限和上传控制，法律因素对应审核、日志和联系方式控制。验证结果见表 5-4。

表 5-4　社会、安全、法律功能验证

因素	验证内容	结论
社会因素	失物招领审核后发布，找回后可标记已解决；课程资源和图书审核通过后可下载并统计贡献。	通过
安全因素	普通用户直接调用管理接口会被拒绝；非法文件不会写入 OSS；封禁申请人不能新建预约。	通过
法律因素	未审核内容不公开，违规资源可驳回或下架，下载日志与必要联系方式保留。	通过

这些功能对应课程要求中的相关条目。

5.5 运行结果分析

从页面运行和接口联调结果看，系统主要功能可以满足实训演示要求。普通用户可以查看公告、填写问卷、发布失物信息、预约功能房、下载课程资源和收藏图书；管理人员可以维护公告、管理问卷、审核预约、审核资源、审核图书和处理失物。状态标签、驳回原因、下载次数、阅读回执和日志记录都能在页面或接口结果中体现。

性能方面，当前测试数据量较小，分页列表、功能房时间轴和排行榜查询均能正常返回。本次测试不能证明系统在高并发和海量文件场景下的表现，后续可增加批量测试数据、接口响应时间统计和数据库执行计划分析。

第六章　项目管理与实训总结

本章总结本次综合实训的开发过程、项目成果、不足和改进方向。项目来自已有学院生活平台的业务基础，但报告重新梳理了需求、设计、实现、测试和责任约束，使内容对应课程教学大纲。

6.1　开发过程与任务分解

本项目按“调研与需求分析、概要设计、详细设计、编码与联调、测试验证”的顺序推进。前期先梳理校园常见事务，确定通知公告、问卷表单、失物招领、功能房预约、课程资源分享和数字图书馆六个核心模块；随后划分门户端与管理端边界；设计阶段整理数据库表结构、业务状态和权限模型；编码与联调阶段围绕接口、页面、文件上传和审核流程逐步验证。

项目任务分解如表 6-1 所示。

表 6-1　项目任务分解

阶段	主要工作
需求整理	分析学院事务线上化需求，明确普通用户、学生工作者、专业负责人、图书管理员、管理员等角色的使用目标。
概要设计	设计前后端分离架构、门户端与管理端边界、权限模型、数据对象和主要业务状态流转。
详细设计	细化后端分层、前端页面、数据库表结构、OSS 上传、预约冲突检测、资源审核、下载日志等实现方式。
编码联调	对公告、问卷、失物、功能房、课程资源和数字图书等模块进行接口与页面联调，处理权限和状态边界。
测试验证	通过浏览器和接口工具验证主要流程、异常场景、权限边界、文件上传和专项责任功能。

项目推进中，难点不在单个页面，而在多模块规则统一。例如多个模块都需要审核、驳回、下架和日志记录，也都涉及本人数据与管理权限边界；文件上传既服务公告附件，也服务失物图片、课程资源和图书资产。统一这些规则后，后续实现会顺很多。

6.2 项目成果总结

本次实训最终完成了一个以 Spring Boot3、Vue3、PostgreSQL 和阿里云 OSS 为基础的学院生活平台。系统覆盖信息发布、问卷收集、公共空间预约、资料共享、数字图书管理和失物招领等场景，可支撑普通用户使用，也可支撑管理人员审核维护。

功能上，平台已经覆盖门户端和管理端主要流程。通知公告支持定向可见和阅读回执；问卷表单支持题目配置、提交和修改；失物招领支持审核公开和已解决标记；功能房预约支持时间轴、冲突检测、审核和封禁；课程资源支持专业课程层级、审核、下载计数和积分榜；数字图书馆支持 ISBN 校验、多资产、收藏、下载和日志记录。

工程上，后端把权限、业务规则和事务处理集中在 Service 层，前端按门户与管理端组织页面，数据库按业务域划分表结构，文件资产交由阿里云 OSS 保存。社会、安全、法律相关约束也对应到了具体功能。

6.3 不足与改进方向

项目仍有不足。测试数据规模较小，当前验证更偏向功能正确性和边界规则，没有覆盖高并发预约、大量文件上传和大规模排行榜统计。部分统计功能仍基于实时查询或简单聚合，后续数据量增长后，可以增加定时汇总表或物化统计视图。功能房预约的状态推进也需要补充定时任务。

文件治理也可以继续完善。课程资源和数字图书已经记录文件哈希、OSS 地址和下载日志，但 OSS 对象清理、文件预览安全、重复文件治理和版权声明流程还不够细。问卷模块目前支持文本、单选和多选，后续可以扩展文件题、日期题、分段题和结果导出。

后续可继续完善统计面板、导出报表、定时任务和文件清理；安全与性能方面，补充接口测试、日志审计查询、敏感信息脱敏和大规模数据下的响应时间分析。

6.4 实训体会

通过本次综合实训，我对 Web 项目的完整开发过程有了更具体的认识。单独完成某个增删改查页面并不困难，需要反复推敲的是业务状态、权限边界、异常处理和数据留痕。比如功能房预约，如果只做提交表单，页面很快就能完成；但要投入使用，还要处理房间状态、时间冲突、申请人封禁、审核开关、取消规则和统计口径。课程资源和数字图书也是如此，上传与下载只是入口，审核、下架、去重和日志才决定它能否被规范使用。

报告写作也让我重新梳理了责任约束。课程要求检查社会、健康、安全和法律因素，但写到具体项目时，还是要回到已有功能本身。失物招领是否审核，联系方式是否最小化采集，图书资源是否能下架，预约是否能阻止冲突，下载是否有日志，这些细节会直接影响系统是否规范。

本项目基本达到了综合实训对需求分析、总体设计、详细设计、编码测试和成果汇报的要求。后续如果继续迭代，我会优先补充自动化测试、性能测试和运维监控。

参考文献

[1] Martin R C. Clean code: a handbook of agile software craftsmanship[M]. Upper Saddle River: Prentice Hall, 2008.

[2] Sommerville I. Software engineering[M]. 10th ed. Boston: Pearson, 2015.

[3] Pressman R S, Maxim B R. Software engineering: a practitioner's approach[M]. 9th ed. New York: McGraw-Hill Education, 2019.

[4] Walls C. Spring in action[M]. 6th ed. Shelter Island: Manning Publications, 2022.

[5] Spring. Spring Boot Reference Documentation[EB/OL]. [2026-06-15]. https://docs.spring.io/spring-boot/

[6] Spring. Spring Security Reference[EB/OL]. [2026-06-15]. https://docs.spring.io/spring-security/reference/

[7] MyBatis. MyBatis 3 User Guide[EB/OL]. [2026-06-15]. https://mybatis.org/mybatis-3/

[8] Vue.js Team. Vue.js Guide[EB/OL]. [2026-06-15]. https://vuejs.org/guide/

[9] Mozilla Developer Network. JavaScript Guide[EB/OL]. [2026-06-15]. https://developer.mozilla.org/docs/Web/JavaScript/Guide

[10] PostgreSQL Global Development Group. PostgreSQL Documentation[EB/OL]. [2026-06-15]. https://www.postgresql.org/docs/

[11] Silberschatz A, Korth H F, Sudarshan S. Database system concepts[M]. 7th ed. New York: McGraw-Hill Education, 2019.

[12] OWASP Foundation. OWASP Top 10: Web Application Security Risks[EB/OL]. [2026-06-15]. https://owasp.org/www-project-top-ten/

[13] OWASP Foundation. Application Security Verification Standard 4.0[EB/OL]. [2026-06-15]. https://owasp.org/www-project-application-security-verification-standard/

[14] 王珊, 萨师煊. 数据库系统概论 [M]. 5 版. 北京: 高等教育出版社, 2014.

[15] 明日科技. Java 从入门到精通 [M]. 5 版. 北京: 清华大学出版社, 2019.

[16] 阿里云计算有限公司. 对象存储 OSS 产品文档 [EB/OL]. [2026-06-15]. https://help.aliyun.com/zh/oss/

[17] 全国人民代表大会常务委员会. 中华人民共和国网络安全法 [Z]. 北京: 中国法制出版社, 2017.

[18] 全国人民代表大会常务委员会. 中华人民共和国数据安全法 [Z]. 北京: 中国法制出版社, 2021.

[19] 全国人民代表大会常务委员会. 中华人民共和国个人信息保护法 [Z]. 北京: 中国法制出版社, 2021.

[20] 国家质量监督检验检疫总局, 中国国家标准化管理委员会. GB/T 25000.10–2016 系统与软件工程 系统与软件质量要求和评价（SQuaRE）第 10 部分：系统与软件质量模型 [S]. 北京: 中国标准出版社, 2016.