学院生活平台系统

简要说明书

班级：　　学号：　　姓名：

1 引言

1.1 编写目的

本概要设计说明书旨在明确 “学院生活平台” 的总体架构、模块划分、接口设计、数据结构等核心设计内容，为后续详细设计、开发、测试及维护工作提供统一依据。预期读者包括项目开发人员、测试人员、运维人员、项目管理人员及相关评审人员。

1.2 背景

a. 待开发软件系统的名称：校园综合服务平台 b. 任务提出者：学校相关管理部门 c. 开发者：项目开发团队 d. 用户：学校师生 e. 运行该软件的计算站（中心）：学校数据中心

1.3 定义

专门术语 / 外文首字母组词	定义 / 原词组
Ruoyi	若依框架，本系统基于若依开发框架构建，提供快速开发支持
Admin	系统默认管理员账号标识，拥有最高操作权限
数据中心	学院专用数据存储与处理中心，负责系统数据的安全存储与运维
接口	系统内部模块间、系统与外部环境间的数据交互通道
OSS	对象存储服务（Object Storage Service），用于存储用户上传的各类文件资源
RBAC	基于角色的访问控制（Role-Based Access Control），系统权限控制模型
JSON	JavaScript 对象表示法（JavaScript Object Notation），接口数据交换格式
JDBC	Java 数据库连接（Java Database Connectivity），系统与数据库交互的接口规范

1.4 参考资料

a. 本项目的经核准的计划任务书、学院与开发团队签订的项目合同；

b. 若依开发框架官方文档；

c. 《软件开发规范》《数据库设计规范》《网络安全等级保护基本要求》；

d. 学院提供的用户需求调研文档、业务流程说明文档。

2 总体设计

2.1 需求规定

本系统需满足校园内师生在课程资源分享、功能房预约、数字图书馆、失物招领、问卷收集、通知公告等方面的需求，具体如下：

输入输出：支持用户上传各类资源（文件、外链、图书、问卷、通知公告等），并能展示、下载、查看相关内容；

功能要求：实现资源上传与审核、功能房预约与管理、图书检索与下载、失物招领信息发布与查找、问卷创建与填写、通知公告发布与阅读等核心功能；

性能要求：系统响应时间≤3 秒，支持并发用户数≥500，数据存储安全可靠，支持数据备份与恢复。2.2 运行环境

2.2.1 硬件环境

服务器：CPU 为 Intel Xeon E3 及以上，内存≥8GB，硬盘≥500GB，支持千兆网络接口；

客户端：普通个人计算机、智能手机等，支持主流浏览器（Chrome、Firefox、Edge、Safari 等），网络带宽≥1Mbps。

2.2.2 支持环境

服务器操作系统：Windows Server 2012 及以上或 Linux（CentOS 7.0 及以上）；

数据库：MySQL 5.7 及以上；

开发框架：若依（Ruoyi）开发框架；

Web 服务器：Nginx 1.10 及以上或 Tomcat 8.0 及以上；

编程语言：Java、JavaScript、HTML/CSS。

2.3 基本设计概念和处理流程

2.3.1 基本设计概念：

基于若依开发框架的分层架构设计，分为表现层、业务逻辑层、数据访问层和数据存储层，各层职责分离、低耦合。采用 “角色 - 权限 - 菜单” 的权限控制模型，管理员账号（admin）拥有全部权限，普通用户仅拥有对应业务操作权限。

2.3.2 核心处理流程

资源上传与审核流程：用户登录系统后，选择对应模块（课程资源分享、数字图书馆等）上传资源，填写相关信息并提交；系统将资源信息存储至数据库，文件资源上传至 OSS；对应审核角色（专业负责人、图书管理员等）登录系统，查看待审核资源，进行审核操作（通过、驳回、标记最佳等），审核结果反馈给用户，通过审核的资源可在系统中展示供其他用户使用。

功能房预约流程：用户登录系统，查询可用功能房，选择预约时间段并提交预约申请；系统验证预约信息的合法性（如时间段是否冲突、用户是否被封禁等），验证通过后将预约信息存储至数据库；审核人员（如有）审核预约申请，审核通过后预约生效，用户可在预约时间段内使用功能房，使用完成后进行签退操作。

问卷填写与提交流程：用户登录系统，查看可用问卷，进入问卷填写页面，按要求填写各题目答案；填写完成后提交答卷，系统验证答卷的完整性（必填题目是否已填），验证通过后将答卷信息存储至数据库，完成问卷提交。

2.4 结构

2.4.1 系统模块划分

系统分为六大核心模块，各模块下包含具体的子模块，如下表所示：

模块标识符	模块名称	子模块	功能描述
CM	课程资源分享模块	资源上传子模块	支持用户上传文件或外链形式的课程资源，填写相关信息

		资源审核子模块	专业负责人审核本专业的资源，可通过、驳回或标记最佳
		资源展示与使用子模块	用户按专业、课程浏览、搜索、下载已通过资源
		积分与排行子模块	计算用户积分，生成积分排行榜
FR	功能房预约模块	预约管理子模块	用户进行功能房预约、取消预约操作
		房间管理子模块	管理员维护楼房、功能房信息（添加、修改、删除、启用 / 禁用）
		审核子模块（如有）	审核人员审核功能房预约申请
		封禁管理子模块	管理员对违规用户进行封禁操作
DL	数字图书馆模块	图书上传子模块	用户上传电子图书，填写图书相关信息
		图书审核子模块	图书管理员审核图书，进行发布、驳回、下架等操作
		图书检索与使用子模块	用户检索、下载、收藏图书
		排行子模块	生成用户上传数量排行榜和书籍下载次数排行榜
LI	失物招领模块	信息发布子模块	用户发布失物或招领信息，可上传图片
		信息管理子模块	用户修改、删除自己发布的失物招领信息
		信息查询子模块	用户按类型、状态等查询失物招领信息
SU	问卷（表单收集）模块	问卷创建子模块（如有）	授权用户创建问卷，设置题目、选项等
		问卷填写子模块	用户查看问卷并填写答案
		答卷管理子模块	系统存储答卷信息，支持查看答卷统计结果
NO	通知公告模块	公告发布子模块	学生工作者发布通知公告，可上传附件

		公告管理子模块	发布者删除自己发布的公告，管理员进行全局公告管理
		公告查看子模块	

2.4.2 元素间控制关系

所有业务模块（RM、FM）均依赖 UAM 模块完成用户身份验证，未登录用户无法访问核心功能；

UAM、RM、FM 模块均依赖 DM 模块进行数据的存储与查询；

SM 模块对 UAM、RM、FM、DM 模块进行权限控制和运维管理，管理员通过 SM 模块配置普通用户的操作权限；

各业务模块间相互独立，通过系统公共接口进行必要的数据交互。

2.5 功能需求与程序的关系

功能需求	课程资源分享模块程序	功能房预约模块程序	数字图书馆模块程序	失物招领模块程序	问卷模块程序	通知公告模块程序
资源上传功能	√	–	√	√	–	√
资源审核功能	√	√（如有）	√	√	–	–
资源展示 / 查看功能	√	√	√	√	√	√
资源下载功能	√	–	√	–	–	√（附件下载）
预约功能	–	√	–	–	–	–
信息发布功能	–	–	–	√	–	√
信息修改 / 删除功能	√（本人资源）	√（本人预约）	–	√（本人发布）	–	√（本人发布）
检索 / 查询功能	√	√	√	√	–	–
积分计算功能	√	–	–	–	–	–

排行榜功能	√	–	√	–	–	–
问卷填写 / 提交功能	–	–				

2.6 人工处理过程

资源审核过程：专业负责人、图书管理员等审核角色需人工查看用户上传的资源，根据相关规则进行审核判断，填写审核意见（如驳回理由）；

功能房预约审核过程（如有）：审核人员需人工查看用户的功能房预约申请，确认是否符合预约要求，进行审核通过或驳回操作；

违规处理过程：管理员发现用户存在违规行为（如上传违规资源、恶意预约等），需人工进行封禁等处理操作；

数据备份与恢复过程：运维人员定期人工执行数据备份操作，当系统数据发生丢失或损坏时，人工执行数据恢复操作。

2.7 尚未解决的问题

系统高并发场景下的性能优化问题，需在详细设计阶段进一步研究并制定优化方案；

不同模块间数据同步的一致性问题，需明确数据同步机制和冲突解决策略；

系统与学校现有其他系统（如教务系统、用户认证系统）的集成方案尚未最终确定，需进一步沟通和协商；

电子图书版权问题，需制定相关规则和措施，确保用户上传的电子图书不侵犯版权。3 接口设计

3.1.1 Web 端用户接口

登录界面：提供用户名、密码输入框，登录按钮，忘记密码链接；

主界面：包含各功能模块入口导航栏、用户信息展示区、通知公告展示区等；

资源上传界面：提供资源类型选择（文件 / 外链）、专业选择、课程选择、资源名称输入框、资源描述输入框、文件上传控件（文件类型）、外链 URL 输入框（外链类型）、提交按钮；

资源浏览 / 搜索界面：提供专业筛选条件、课程筛选条件、搜索输入框、资源列表展示区（包含资源名称、上传者、上传时间、下载次数等信息）、下载按钮；

功能房预约界面：提供楼房选择、楼层选择、房间选择、预约时间段选择、使用目的输入框、提交按钮；

问卷填写界面：按顺序展示问卷题目及选项（单选 / 多选）、文本题输入框，底部提供提交按钮；

通知公告查看界面：展示公告标题、发布者、发布时间、公告内容、附件下载链接。

3.1.2 命令及语法结构

系统主要通过图形化界面供用户操作，无需用户输入命令。部分接口（如搜索功能）支持简单的关键词搜索语法，用户直接输入关键词即可进行搜索。

3.1.3 软件回答信息

操作成功提示：如 “资源上传成功，请等待审核”“预约申请提交成功”“问卷提交成功” 等；

操作失败提示：如 “资源上传失败，请检查文件大小和格式”“该时间段已被预约，请选择其他时间段”“必填题目未填写，请补充完整” 等；

状态提示：如 “您的资源正在审核中”“您的预约已通过审核” 等。

3.2 外部接口

3.2.1 软件与硬件接口

服务器与存储设备接口：通过网络接口连接阿里云 OSS 存储服务，采用 HTTP/HTTPS 协议进行文件上传和下载操作；

客户端与服务器接口：通过网络接口进行数据交互，采用 HTTP/HTTPS 协议。

3.2.2 系统与支持软件接口

与数据库接口：采用 JDBC 接口连接 MySQL 数据库，进行数据的增删改查操作；
与缓存服务接口：采用 Redis 客户端接口连接 Redis 服务，进行数据的缓存和读取操作；
与对象存储服务接口：采用阿里云 OSS 提供的 SDK 接口，进行文件的上传、下载、删除等操作；
与应用服务器接口：Web 应用部署在 Tomcat/Nginx 应用服务器上，通过应用服务器提供的接口对外提供服务。
3.3 内部接口
3.3.1 模块间接口
用户模块与其他模块接口：
输入：用户名、密码；
输出：用户身份验证结果、用户角色信息、用户权限信息。
课程资源分享模块与数据库模块接口：
输入：资源信息（专业 ID、课程 ID、资源名称、资源类型等）、用户操作指令（查询、新增、更新、删除）；
输出：数据库操作结果、查询到的资源数据。
功能房预约模块与数据库模块接口：
输入：预约信息（房间 ID、申请人 ID、预约时间段、使用目的等）、用户操作指令；
输出：数据库操作结果、查询到的预约数据、房间可用状态信息。
各模块间数据交互接口：如通知公告模块向其他模块提供公告列表查询接口，其他模块可调用该接口获取公告信息并展示。
3.3.2 接口数据格式
内部接口数据交换采用 JSON 格式，例如用户登录接口返回数据格式：
json
```
{
  "code": 200,
  "message": "登录成功",
  "data": {
    "userId": "1001",
    "userName": "张三",
    "role": "普通用户",
    "permissions": ["资源上传", "资源下载", "问卷填写"]
  }
}
```
4 运行设计
4.1 运行模块组合
用户登录运行组合：用户模块（身份验证子模块）+ 数据库模块（用户信息查询子模块）；
资源上传运行组合：对应模块的资源上传子模块 + 数据库模块（数据存储子模块）+ OSS 存储服务（文件存储子模块）；
资源审核运行组合：对应模块的资源审核子模块 + 数据库模块（数据更新子模块）+ 通知模块（审核结果通知子模块）；
功能房预约运行组合：功能房预约模块的预约管理子模块 + 数据库模块（数据存储与查询子模块）+ 权限模块（用户权限验证子模块）；
问卷填写与提交运行组合：问卷模块的问卷填写子模块 + 数据库模块（答卷数据存储子模块）；
系统后台管理运行组合：各模块的管理子模块 + 数据库模块（数据管理子模块）+ 权限模块（管理员权限验证子模块）。
4.2 运行控制
4.2.1 用户登录控制

用户打开系统登录页面，输入用户名和密码；
系统调用用户模块的身份验证子模块，向数据库模块发送用户信息查询请求；
数据库模块查询用户信息并返回给身份验证子模块；
身份验证子模块验证用户名和密码的正确性，若正确则返回用户角色和权限信息，跳转至系统主界面；若错误则提示用户 “用户名或密码错误”。
4.2.2 资源上传控制
用户登录系统后，进入对应模块的资源上传页面；
选择资源类型（文件 / 外链），填写相关信息（专业、课程、资源名称、描述等），上传文件（文件类型）或输入外链 URL（外链类型）；
用户点击提交按钮，系统验证输入信息的完整性和合法性（如专业和课程是否选择、资源名称是否填写、文件格式和大小是否符合要求等）；
验证通过后，系统调用数据库模块存储资源基本信息，调用 OSS 存储服务上传文件（文件类型）；
操作完成后，向用户返回 “资源上传成功，请等待审核” 的提示。
4.2.3 资源审核控制
审核人员登录系统，进入对应模块的资源审核页面；
系统查询待审核资源列表并展示，审核人员选择某一资源进行查看；
审核人员根据资源内容和相关规则，选择审核结果（通过、驳回、标记最佳），填写审核意见（如驳回理由）；
审核人员点击提交按钮，系统调用数据库模块更新资源审核状态和审核信息；
若审核通过，系统将资源状态更新为 “已通过”，可在资源展示区展示；若驳回，系统向资源上传者发送驳回通知。
4.2.4 功能房预约控制
用户登录系统，进入功能房预约页面；
选择楼房、楼层、房间，系统查询该房间的可用时间段并展示；
用户选择预约开始时间和结束时间，填写使用目的，点击提交按钮；
系统验证预约信息：检查预约时间段是否与已有的预约冲突、用户是否被封禁、预约时长是否超过最大限制等；
验证通过后，系统调用数据库模块存储预约信息，若需要审核则将预约状态设置为 “待审”，否则设置为 “已批准”；
操作完成后，向用户返回预约结果提示。
4.3 运行时间
用户登录：正常情况下，登录过程耗时≤1 秒；
资源上传：根据文件大小不同，上传时间有所差异，小文件（≤100MB）上传耗时≤5 秒，大文件（100MB~1GB）上传耗时≤30 秒；
资源审核：审核人员操作耗时由人工决定，系统数据更新耗时≤1 秒；
资源查询与下载：资源列表查询耗时≤2 秒，小文件下载耗时≤3 秒，大文件下载耗时根据网络带宽和文件大小而定；
功能房预约：预约操作数据处理耗时≤2 秒；
问卷填写与提交：问卷提交数据处理耗时≤2 秒；
通知公告查看与附件下载：公告查看耗时≤1 秒，附件下载耗时根据文件大小而定。
5 系统数据结构设计
5.1 逻辑结构设计要点
系统数据主要包括用户相关数据、课程资源相关数据、功能房相关数据、数字图书馆相关数据、失物招领相关数据、问卷相关数据、通知公告相关数据等，各核心数据结构的逻辑关系如下：

5.1.1 用户相关数据结构

数据结构名称	标识符	数据项	定义	长度	相互关系
用户表	sys_user	id	用户唯一标识	bigint(20)	主键，与角色表通过角色 ID 关联
		username	登录账号	varchar(50)	唯一，如 admin
		password	加密后的密码	varchar(100)	非空
		role_id	角色 ID	int(11)	关联角色表（sys_role）的 id 字段
		status	账号状态（0 正常 / 1 禁用）	char(1)	默认值为 0
		create_time	创建时间	datetime	自动记录
		update_time	更新时间	Datetime	自动记录

5.1.2 核心业务数据结构

数据结构名称	标识符	核心数据项	定义	数据类型及长度	相互关系
角色表	sys_role	id、role_name、permissions	存储角色信息及权限配置	id:int(11)，role_name:varchar(50)，permissions:varchar(1000)	主键，与用户表通过 role_id 多对一关联
专业表	tb_major	id、major_name、status、del_flag	存储专业信息	id:bigint(20)，major_name:varchar(128)，status:char(1)，del_flag:char(1)	主键，与课程表通过 major_id 一对多关联，与专业负责人映射表通过 major_id 一对多关联

课程表	tb_course	id、major_id、course_name、status、del_flag	存储课程信息	id: bigint(20), major_id: bigint(20), course_name: varchar(255), status: char(1), del_flag: char(1)	主键，与专业表通过 major_id 多对一关联，与课程资源主表通过 course_id 一对多关联
课程资源主表	tb_course_resource	id、major_id、course_id、resource_name、resource_type、status、uploader_id、del_flag	存储课程资源核心信息	id: bigint(20), major_id: bigint(20), course_id: bigint(20), resource_name: varchar(255), resource_type: tinyint(4), status: tinyint(4), uploader_id: bigint(20), del_flag: char(1)	主键，与专业表、课程表、用户表分别通过 major_id、course_id、uploader_id 关联，与最佳标记表、操作审计表、用户积分流水表通过 resource_id 一对多关联
楼房表	tb_facility_building	id、building_name、status、del_flag	存储楼房信息	id: bigint(20), building_name: varchar(128), status: char(1), del_flag: char(1)	主键，与功能房表通过 building_id 一对多关联
功能房表	tb_fa	id、building_id、	存储功能房信	id:	主键，与楼房表

	cility_room	floor_no、room_name、status、del_flag	息	bigint(20), building_id: bigint(20), floor_no: int(11), room_name: varchar(128), status: char(1), del_flag: char(1)	通过 building_id 多对一关联，与功能房预约表通过 room_id 一对多关联
功能房预约表	tb_facility_booking	id、room_id、applicant_id、start_time、end_time、status、del_flag	存储功能房预约信息	id: bigint(20), room_id: bigint(20), applicant_id: bigint(20), start_time: datetime, end_time: datetime, status: char(1), del_flag: char(1)	主键，与功能房表、用户表分别通过 room_id、applicant_id 关联，与预约使用人关联表通过 booking_id 一对多关联
图书主体表	tb_library_book	id、isbn13、title、author、status、uploader_id、del_flag	存储数字图书核心信息	id: bigint(20), isbn13: char(13), title: varchar(255), author: varchar(255), status: char(1), uploader_id: bigint(20), del_flag:	主键，与用户表通过 uploader_id 关联，与图书资产表、收藏关系表、下载日志表、操作日志表通过 book_id 一对多关联

				char(1) id: bigint(20), type: varchar(10), title: varchar(100), content: text, create_by: varchar(64), status: char(1), solved_flag: char(1), del_flag: char(1)	
失物招领主体表	tb_lost_item	id、type、title、content、create_by、status、solved_flag、del_flag	存储失物招领核心信息		主键，与失物招领图片表通过 item_id 一对多关联
问卷主表	tb_survey	id、title、status、deadline、create_by、del_flag	存储问卷核心信息	id: bigint(20), title: varchar(200), status: char(1), deadline: datetime, create_by: varchar(64), del_flag: char(1)	主键，与问卷可见范围表、问卷题目表、答卷表通过 survey_id 一对多关联
通知公告主体表	tb_notice	id、title、content_html、type、status、publisher_id、del_flag	存储通知公告核心信息	id: bigint(20), title: varchar(200), content_html: mediumtext, type: tinyint(4), status: tinyint(4),	主键，与通知公告可见范围表、附件表、阅读回执表通过 notice_id 一对多关联

				publisher_id：bigint(20)，del_flag：char(1)

各数据结构通过主键 - 外键关联，确保数据一致性，例如用户表的 id 关联资源表的 uploader_id、预约表的 applicant_id 等，专业表的 id 关联课程表的 major_id 等。

5.2 物理结构设计要点

5.2.1 存储要求

数据库存储：采用 MySQL 数据库，数据文件存储在学院数据中心服务器的专用磁盘分区，支持磁盘阵列（RAID 5）备份，重要数据定期全量备份 + 增量备份；

账号密码存储：密码采用 MD5 加密后存储，不保留明文，加盐处理增强安全性；

文件存储：用户上传的资源文件存储于阿里云 OSS，按模块 + 日期（年 / 月 / 日）划分存储目录，服务器本地仅暂存上传临时文件；

缓存存储：Redis 缓存数据存储在服务器内存中，重要缓存数据支持持久化（RDB+AOF 混合模式）。

5.2.2 访问方法与索引设计

访问方法：通过 JDBC 接口执行 SQL 语句，实现数据的增删改查操作；采用 MyBatis 框架简化数据库操作，提高开发效率；

索引设计：

主键索引：所有表的 id 字段均创建主键索引（PRIMARY KEY），采用 BTREE 结构；

唯一索引：用户表 username 字段、专业表 major_name 字段、课程表（major_id,course_name）组合字段、图书表 isbn13 字段等创建唯一索引，确保数据唯一性；

普通索引：资源表的 course_id、status、uploader_id 字段，预约表的 room_id、applicant_id、start_time、status 字段，图书表的 title、author、status 字段，问卷表的 status、create_by 字段，公告表的 status、publisher_id、title 字段等创建普通索引，提升查询效率；

组合索引：资源表（course_id,status）、预约表（room_id,start_time,end_time）、公告表（status,pinned,pinned_time,publish_time）等创建组合索引，优化多条件查询；

存取单位：以记录为基本存取单位，单条记录或批量记录均可进行读写操作，批量操作采用分批处理方式，避免占用过多数据库连接。

5.2.3 保密条件

数据库访问需通过专用账号密码验证，仅系统服务账号拥有访问权限，不同模块采用最小权限原则分配数据库账号；

数据传输采用 HTTPS 加密协议，防止传输过程中被窃取或篡改；

管理员操作日志、用户登录日志、敏感数据操作日志全程记录，包含操作人、操作时间、操作内容、IP 地址等信息，便于追溯违规操作；

OSS 文件访问采用签名 URL 方式，设置有效时长，防止文件被非法访问；

敏感数据（如用户联系方式）在展示时进行脱敏处理（如隐藏部分手机号）。

5.3 数据结构与程序的关系

用户模块：依赖 sys_user、sys_role 表，实现用户登录、身份验证、权限控制功能，程序通过查询用户表和角色表获取用户信息及权限，进行登录验证和操作授权；

课程资源分享模块：依赖 tb_major、tb_course、tb_course_resource、tb_course_resource_best 等表，程序通过这些表存储和查询资源相关数据，实现资源上传、审核、展示、下载、积分计算等功能；

功能房预约模块：依赖 tb_facility_building、tb_facility_room、tb_facility_booking、

tb_facility_booking_user 等表，程序通过这些表存储和查询楼房、房间、预约相关数据，实现预约申请、审核、签到签退等功能；
数字图书馆模块：依赖 tb_library_book、tb_library_book_asset、tb_library_book_favorite 等表，程序通过这些表存储和查询图书相关数据，实现图书上传、审核、检索、下载、收藏等功能；
失物招领模块：依赖 tb_lost_item、tb_lost_item_image 表，程序通过这些表存储和查询失物招领相关数据，实现信息发布、管理、查询等功能；
问卷模块：依赖 tb_survey、tb_survey_scope、tb_survey_item、tb_survey_answer 等表，程序通过这些表存储和查询问卷、题目、答卷相关数据，实现问卷创建、填写、提交、统计等功能；
通知公告模块：依赖 tb_notice、tb_notice_scope、tb_notice_attachment、tb_notice_read 表，程序通过这些表存储和查询公告相关数据，实现公告发布、管理、查看、阅读回执等功能；
缓存模块：程序将排行榜数据、热点资源数据、用户会话数据等缓存至 Redis，提高查询效率，减少数据库压力，定期更新缓存数据确保一致性。

6　系统出错处理设计

6．1 出错信息

出错 / 故障情况	系统输出信息形式	含意	处理方法
用户登录时用户名或密码错误	页面弹窗提示	输入的用户名或密码不正确	用户检查用户名和密码后重新输入，可通过 “忘记密码” 功能找回密码
用户登录时验证码错误（如有）	页面弹窗提示	输入的验证码不正确	刷新验证码后重新输入
资源上传时文件大小超过限制	页面弹窗提示	上传的文件大小超过系统限制（如 500MB）	用户压缩文件或选择更小的文件重新上传
资源上传时文件格式不支持	页面弹窗提示	上传的文件格式不在支持的格式列表中	用户更换为支持的文件格式后重新上传
资源上传时网络异常	页面弹窗提示	网络连接中断或不稳定导致上传失败	用户检查网络连接后重新上传，大文件支持断点续传
功能房预约时时间段冲突	页面弹窗提示	选择的预约时间段已被其他用户预约	用户重新选择未被占用的时间段
功能房预约时用户被封禁	页面弹窗提示	用户因违规操作被限制预约功能	用户联系管理员了解封禁原因，封禁到期后自动解除

功能房预约时预约时长超过限制	页面弹窗提示	预约时长超过系统最大限制（默认 72 小时）	用户缩短预约时长后重新提交
问卷提交时必填题目未填写	页面弹窗提示	存在未填写的必填题目	用户返回问卷填写页面，补充填写必填题目后重新提交
问卷提交时网络异常	页面弹窗提示	网络连接中断导致提交失败	用户检查网络连接后重新提交，若已保存草稿可恢复填写
文件下载时网络异常	页面弹窗提示	网络连接中断导致下载失败	用户检查网络连接后重新下载
数据库连接失败	系统页面提示 + 日志记录	无法连接到数据库服务器	运维人员检查数据库服务状态、网络连接及数据库配置，重启数据库服务（必要时）
OSS 服务异常	系统页面提示 + 日志记录	无法连接到 OSS 存储服务	运维人员检查 OSS 服务状态、网络连接及配置，联系阿里云技术支持（必要时）
Redis 缓存服务异常	系统页面提示 + 日志记录	无法连接到 Redis 缓存服务	运维人员检查 Redis 服务状态、网络连接及配置，重启 Redis 服务（必要时），系统降级为直接查询数据库
系统内部错误（如代码异常）	系统页面提示 “系统繁忙，请稍后重试”+ 日志记录	系统运行过程中出现未捕获的异常	开发人员查看系统日志定位错误原因，修复代码漏洞，重启系统服务（必要时）

6.2 补救措施

6.2.1 后备技术

数据库备份：采用定期全量备份（每日凌晨）+ 增量备份（每小时）的方式，备份数据存储在异地服务器和云存储中，保留最近 30 天的备份数据；当原始系统数据丢失时，可通过备份数据恢复至最近的可用状态；

文件备份：阿里云 OSS 存储的文件自动开启版本控制和跨区域备份，防止文件误删或损坏，可通过 OSS 控制台恢复历史版本文件；

缓存备份：Redis 缓存采用 RDB+AOF 混合持久化方式，当 Redis 服务异常重启时，可通过持久化文件恢复缓存数据，减少数据库访问压力。

6.2.2 降效技术

数据库服务异常时：系统暂时关闭非核心功能（如排行榜、统计分析），仅保留核心业务功能（如资源查看、预约申请），降低系统负载；核心功能优先使用本地缓存数据，减少数据库访问；

OSS 服务异常时：系统暂时关闭文件上传功能，仅保留文件下载和查看功能，已上传的文件可通过服务器本地缓存（如有）提供访问；

缓存服务异常时：系统降级为直接查询数据库，关闭非必要的缓存查询功能，优化数据库查询语句，提高查询效率；

高并发场景下：系统采用限流措施，限制同时访问人数，优先保障核心功能可用，非核心功能可暂时关闭或排

队等待。

6.2.3 恢复及再启动技术

程序级恢复：系统采用事务管理机制，数据库操作失败时自动回滚事务，确保数据一致性；关键业务操作（如资源审核、预约提交）记录操作日志，失败时可通过日志手动恢复数据；

服务级恢复：应用服务器采用集群部署，当某一台服务器异常时，负载均衡器自动将请求转发至其他正常服务器，不影响用户使用；服务器异常重启后，自动加载配置文件和上下文，恢复服务功能；

系统级恢复：当整个系统崩溃时，运维人员先通过备份数据恢复数据库和文件存储，然后依次启动数据库服务、缓存服务、应用服务器，最后通过负载均衡器开放系统访问；恢复过程中系统展示维护页面，告知用户恢复进度。

6.3 系统维护设计

检测点设计：在系统核心模块（如用户登录、资源上传、数据存储）设置检测点，定期检查模块运行状态、数据库连接状态、缓存服务状态、OSS 服务状态等，检测结果记录至系统监控日志；

专用维护模块：系统后台提供专用维护模块，支持管理员进行以下操作：

数据备份与恢复：手动触发数据库备份、文件备份，查看备份记录，执行数据恢复操作；

日志管理：查看系统运行日志、操作日志、错误日志，支持日志查询、导出、清理；

缓存管理：查看缓存数据，手动刷新缓存、清除缓存；

系统配置：修改系统核心配置参数（如文件上传大小限制、预约时长限制、并发用户数限制等），无需修改代码即可生效；

服务监控：查看服务器 CPU、内存、磁盘使用率，数据库连接数、查询耗时，接口访问频率等监控指标，支持异常告警；

代码可维护性设计：代码按模块和功能划分目录结构，遵循统一的编码规范，添加详细的注释；核心业务逻辑封装为服务组件，便于修改和扩展；采用面向接口编程，降低模块间耦合度；

版本管理：系统支持版本升级，升级过程中保留用户数据和配置信息，提供升级日志和回滚方案，确保升级失败时可恢复至之前的稳定版本。