



## 知识共享平台

# 架构设计说明书

文件编号: GOSP-R&D-T05-01

版本: V2.1

发布日期: 2025/1/10

安徽科大国创云网科技有限公司

AnhuiGuoChuangCloudTechnologyCo.,Ltd.

# 版本记录

版本号	时间	记录人	变更描述
V1.0	2025/1/5	徐力行	撰写架构设计说明书初稿
V2.0	2025/1/10	徐力行	进一步完善
V2.1	2025/1/10	李健	协助撰写

# 目录

版本记录 .....	2
目录 .....	2
1. 引言 .....	4
1.1 编写目的 .....	4
1.2 背景 .....	4
1.3 定义 .....	4
1.4 参考资料 .....	5
2. 系统设计原则 .....	5
2.1 系统功能 .....	5
2.2 系统性能 .....	6
2.2.1 精度 .....	6
2.2.2 时间特性要求 .....	7
2.2.3 可靠性 .....	7
2.2.4 灵活性 .....	7
2.3 输入输出要求 .....	8
2.4 数据管理能力要求 .....	8
2.5 故障处理要求 .....	8
2.6 其他专门要求 .....	8
3. 产品配置方案 .....	8
4. 业务流程（待补充） .....	9

---

5.	系统架构 .....	10
5.1	系统软件架构图 .....	10
5.2	系统技术架构图 .....	13
5.3	系统硬件架构 .....	15
6.	需求分配表 .....	16
7.	运行环境 .....	19
7.1	设备 .....	19
7.2	支持软件 .....	20
7.3	接口 .....	20
7.4	控制 .....	21
8.	人工处理过程 .....	21
9.	尚未解决的问题 .....	22

## 1. 引言

### 1.1 编写目的

本文档旨在对“知识共享平台”的系统架构进行设计说明，用于指导开发、测试和后续运维工作的开展。通过本说明书，读者可以清晰了解系统整体结构、业务流程、技术选型以及关键功能设计，确保项目各阶段的工作有章可循。

### 1.2 背景

在当今数字化时代，知识的获取和传播已成为人们学习、工作与日常生活中不可或缺的一部分。然而，传统的知识管理与共享方式存在诸多不足，如内容操作复杂、检索效率低、用户互动体验欠佳等，难以满足现代用户对便捷性、高效性及互动性的期待。因此，构建一个创新型知识共享平台，为用户提供智能化、一体化的服务，成为信息技术发展的重要方向之一。

本平台的核心使命是打造一个集知识采集、管理、共享和互动为一体的综合系统，致力于实现知识资源的高效流转与价值提升。通过功能强大的富文本编辑器，用户能够轻松录入并编辑包括文本、图片、视频在内的多种内容形式，使知识展现更加直观生动。为了满足海量数据存储与快速检索的需求，平台将结合先进的数据库技术与ElasticSearch的强大搜索功能，优化数据存储与检索性能，提供精准、高效的搜索服务。同时，平台在设计上注重用户需求，支持用户对内容进行评论、收藏、点赞和分享，构建一个开放且充满活力的知识型社区。

用户隐私保护与个性化服务是平台设计的重要原则。在个人中心模块，用户可管理自己的发布内容、评论记录、收藏列表及点赞历史，同时通过自定义设置提升使用体验。平台将通过严格的权限管理机制，确保用户数据的安全性与隐私性，进一步增强用户对平台的信赖感与依赖度。

随着知识经济的持续发展，社会对知识共享的需求日益提升。本平台的建设不仅是为了提供高效便捷的知识管理工具，更希望通过技术创新推动知识的广泛传播与深度交流，提升个人与组织的学习效率、创新能力及协作水平，为知识型社会的发展注入新的动力。

### 1.3 定义

术语	定义
知识共享平台	一个支持用户录入、管理、分享、搜索和互动知识的系统，旨在推动知识高效传播与利用。
富文本编辑器	一种支持文本格式化、图片插入、视频嵌入等功能的工具，用于录入和编辑多媒体内容。

ElasticSearch	一种分布式搜索引擎，能够提供高效的数据存储与检索功能，适用于大规模数据查询场景。
权限管理	用于控制用户对平台资源的访问权限的机制，确保数据安全和隐私保护。
关键字搜索	基于用户输入的关键词，快速检索平台知识库中的相关内容的功能。
用户中心	集中显示用户个人信息、发布记录、评论历史、收藏内容和设置页面的模块。
点赞	用户对平台内容表示喜欢的一种互动方式，通常用于提升内容的受欢迎程度。
评论	用户针对特定知识内容发表观点、提问或讨论的一种互动行为。
收藏	用户将特定内容保存到个人空间以便日后查看的一种操作。
知识发布	用户通过编辑器创建并上传知识内容供其他用户查看的操作。
数据库索引	一种优化数据查询效率的技术，通过构建索引加速检索过程。
隐私保护	保障用户个人信息和数据不被未授权访问或泄露的技术和策略。
用户体验	用户在使用平台时的总体感受，包括界面设计、操作便捷性和响应速度等。
交互设计	围绕用户需求设计的系统与用户之间的交互方式，旨在提升用户满意度。
知识管理	对知识内容进行收集、分类、存储、检索和传播的系统化过程。
高效搜索	在最短时间内根据用户查询需求返回相关结果的功能。
多媒体内容	包括文本、图片、视频、音频等多种信息表现形式的内容。
评论历史	用户发表的所有评论的记录，存储于用户中心供用户查看和管理。
收藏列表	用户保存的所有收藏内容的记录，供用户随时查看和使用。
个性化服务	根据用户偏好和行为提供定制化内容或功能的服务。

## 1.4 参考资料

- 《需求规格说明书》
- 《ElasticSearch 官方文档》
- 《数据结构说明书》
- 《知识共享平台项目计划书》

## 2. 系统设计原则

### 2.1 系统功能

#### 1. 用户管理

支持管理员通过接口对用户进行管理，包括用户的新增、编辑、删除及查询操作。

提供用户权限管理功能，支持分级分组，确保系统数据安全。

#### 2. 帖子管理

支持管理员对帖子内容的审核、修改、删除及分类管理。

支持按时间、分类、关键字检索帖子，提高管理效率。

### 3. 评论与反馈管理

提供评论管理接口，支持对用户评论的审核、隐藏、删除及分类统计。

反馈管理功能支持用户提交问题或建议，后台可对其进行分类处理并跟踪状态。

### 4. 搜索与检索

支持用户通过关键字、分类标签对知识内容及历史记录进行快速检索。

结合数据库索引和 ElasticSearch 技术，提升搜索性能与精度。

### 5. 个人中心管理

提供用户查看与管理自己发布的帖子、评论、收藏及历史记录的功能。

支持个性化设置，包括通知管理、偏好设置及账户信息更新。

### 6. 数据统计与分析

提供统计接口，支持对系统内各类数据的统计，包括用户活跃度、内容发布量及用户反馈处理情况等。

支持管理员通过可视化界面查看数据报表，以便于决策与优化。

### 7. 系统权限与安全

实现多级权限管理，不同用户角色具备不同的操作权限，确保系统安全性。

通过严格的身份认证机制保护用户隐私及系统敏感数据。

### 8. 推荐与个性化服务

通过推荐算法为用户推送相关内容，包括推荐帖子、收藏与关注内容。

支持用户订阅感兴趣的主题，平台根据兴趣标签生成个性化推荐。

### 9. 接口与模块集成

提供完善的 API 接口支持，确保系统模块之间的高效协作。

开放接口支持第三方工具集成，为扩展功能提供灵活性。

### 10. AI 辅助功能

支持 AI 对话与智能问答，为用户提供实时知识解答与学习建议。

结合用户数据，优化搜索、推荐与用户互动体验。

### 11. 排行榜功能

支持系统生成内容排行榜（如热度、点赞、评论等）。

## 2.2 系统性能

### 2.2.1 精度

系统的搜索功能在利用数据库索引或 ElasticSearch 进行查询时，应确保检索结果的准确性与相关度。无论是基于关键字的简单查询，还是基于全文索引的复杂检索，系统都需对用户的查询意图进行合理解析，确保返回的结果与用户需求高度匹

配。为进一步提高精度，可结合用户行为（如历史搜索、浏览记录）优化排序算法，提供个性化的检索体验。

在用户点赞、评论、收藏等交互操作中，系统的统计功能必须保证数据的实时性或准实时性更新。无论是点赞数的实时刷新，还是评论数量的准确统计，系统都需确保不同模块之间的同步性，避免出现计数偏差或用户反馈延迟。对于数据的统计结果，还需支持在多设备、多会话中的一致性展示，确保用户体验连贯。

### 2.2.2 时间特性要求

用户操作的响应速度直接影响体验质量，因此系统在执行用户核心操作时需具备极高的响应能力。例如，发布内容、评论、点赞、收藏等操作应能在秒级甚至毫秒级内完成，确保用户交互过程流畅无阻。对于高并发场景下的操作响应，系统需具备负载均衡能力，以保持性能的稳定性。

搜索查询的性能同样至关重要，系统需在毫秒至秒级内返回检索结果，避免因响应延迟导致用户流失。若使用 ElasticSearch 作为搜索服务的核心，需持续优化其集群的健康状态，定期调整索引结构与查询参数，确保搜索请求的平均响应时间在可控范围内。此外，为应对复杂查询和高并发需求，可结合缓存策略减少对主存储的直接访问，提高查询性能。

### 2.2.3 可靠性

系统的可靠性是保障用户数据安全与服务持续性的基础。数据库服务和 ElasticSearch 服务需具备高可用能力，通过主从复制、集群部署等方式规避单点故障的风险。在服务运行过程中，应实时监控系统健康状态，预防可能的异常事件发生。同时，ElasticSearch 集群需支持节点故障自动恢复，确保服务不中断。

系统需设计完善的自动备份与灾备机制，对关键数据（如用户内容、操作记录等）定期备份并存储至安全区域。一旦发生数据丢失或硬件故障，系统可快速启动恢复流程，尽可能降低业务中断时间和用户数据损失的风险。此外，系统需通过异常检测与报警机制，确保运维团队能够及时发现和响应潜在问题。

### 2.2.4 灵活性

系统设计需保持高度的灵活性，以适应不断变化的业务需求与用户场景。通过模块化设计，系统可根据实际需求动态调整功能组件的规模。例如，当业务增长导致搜索请求激增时，可快速扩容 ElasticSearch 集群；若业务中对消息队列需求不明显，也可将其作为可选插件而非必备组件，从而提高资源利用率。

系统在对接第三方服务（如支付网关、用户身份认证、AI 推荐引擎等）时，需提供灵活的接口支持，通过标准化的 API 或插件形式实现快速集成。同时，系统模块的

解耦设计还需保障功能的独立性，避免某一功能的调整对其他模块产生过多影响。在系统功能扩展过程中，开发团队应能快速上手并投入开发，从而提升业务响应速度。

## 2.3 输入输出要求

**输入：**主要包括用户通过富文本编辑器提交的文本、图片或富媒体内容，以及用户的检索关键词、评论等操作。

**输出：**主要以各种格式的知识展示为主，包括文本、图片、视频链接；搜索结果列表；个人中心信息等。

## 2.4 数据管理能力要求

支持对知识内容的数据分类存储与权限控制。

对大量知识内容可分级管理、分区存储，提高查询效率。

数据可定期归档，历史数据仍可检索和查看。

## 2.5 故障处理要求

当网络或服务器发生故障时，系统应提供降级方案，保证核心功能（如搜索和查看知识）正常。

提供实时监控和报警机制，出现异常时可快速定位、处理。

关键操作（如发布、编辑、删除）需做好操作日志，以便事后审计和追溯。

## 2.6 其他专门要求

前端界面需简洁美观、易于操作，并兼顾响应式页面的使用体验。

系统在不同浏览器环境下需有良好的访问速度和稳定性。

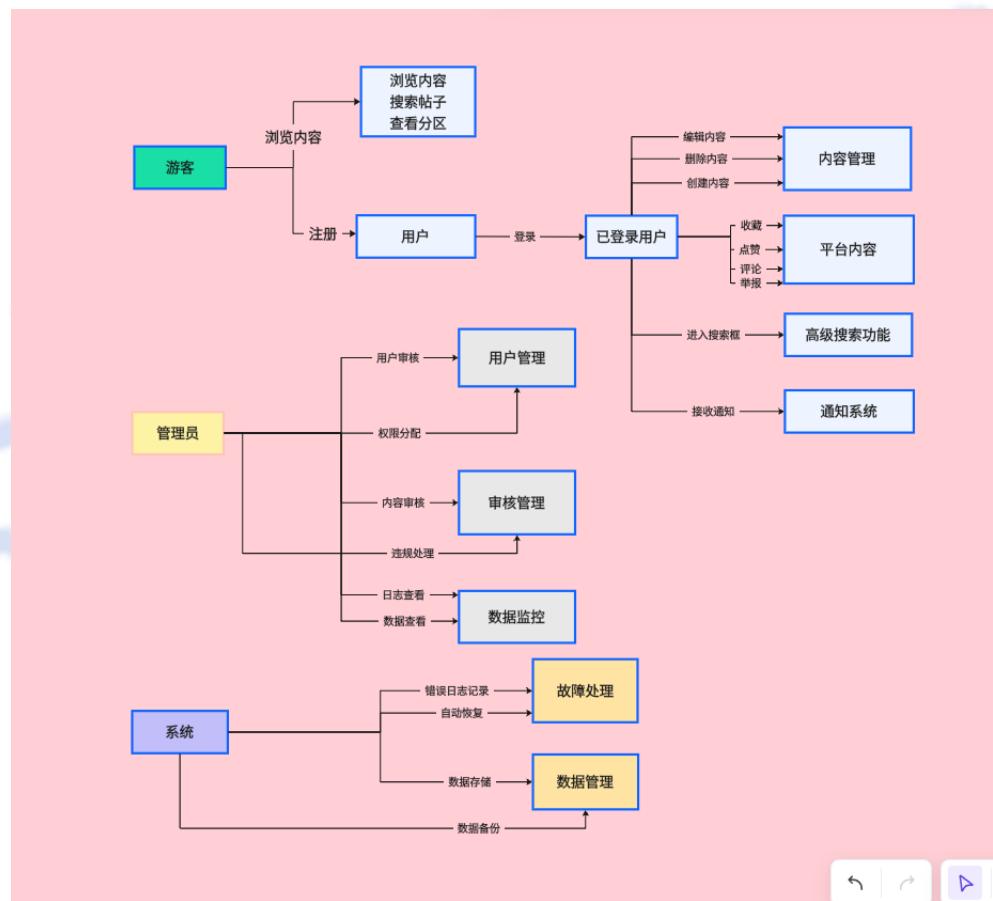
考虑到用户体验，建议加入相应的搜索联想、搜索推荐等功能。

## 3. 产品配置方案

序号	功能	实现形式 (开发、重用、外购)	重用或外购的对象	类型
1	接入安全认证	开发		
2	数据存储和分析	开发		
3	服务注册	开发		
4	服务目录管理	开发		
5	服务查询	开发		
6	服务生命周期管理	开发		
7	服务在线测试	开发		
8	接入管理和鉴权	开发		

9	流量控制	开发		
10	负载均衡	开发		
11	协议适配	开发		
12	分布式事务集成	开发		
13	缓存支持	开发		
14	用户管理	开发		
15	日志管理	开发		
16	系统参数管理	开发		

## 4. 业务流程



知识共享平台的业务流程以用户角色和功能模块为核心展开，主要涉及游客、已登录用户、管理员以及系统本身的功能。

首先，游客作为系统的初始用户角色，能够浏览内容、搜索帖子以及查看分区，但他们只能以有限权限的方式使用平台的基本功能。如果游客想进一步参与互动，例

如评论、点赞或发布内容，则需要通过注册成为用户。一旦注册完成，游客便会转化为“用户”角色，获得更广泛的权限。

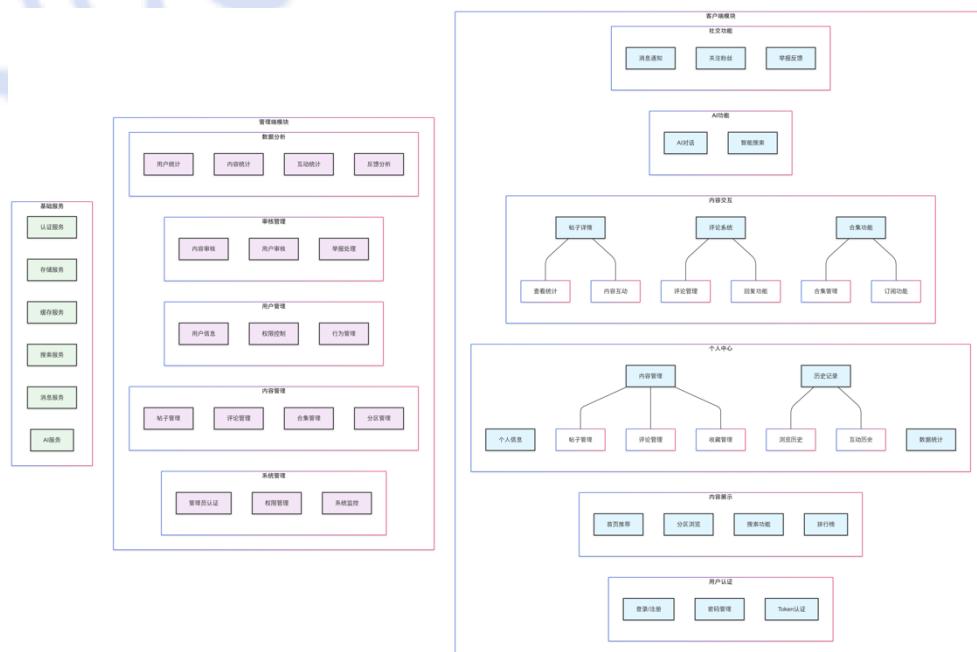
已登录用户是平台的主要参与者，他们可以通过平台提供的“内容管理”模块进行内容创建、编辑或删除，同时还能通过“平台内容”模块参与收藏、点赞、评论、分享等交互操作。这些功能形成了用户与平台之间的双向互动。与此同时，用户还可以利用“高级搜索功能”来快速定位感兴趣的内容，并通过“通知系统”接收各种平台消息，如评论回复、点赞提醒等，进一步提升了用户体验。

系统还设有管理员角色，他们负责系统的监督与管理工作。管理员可以通过“用户管理”模块审核新用户的申请，分配权限，确保用户群体的规范性。此外，管理员还可以使用“审核管理”模块对平台内容进行审核，处理违规内容，并维护平台的健康生态。在内容审查中，管理员不仅会依赖平台内置的规则，还可能根据内容举报的情况做出判定。同时，管理员通过“数据监控”模块查看日志数据、统计数据，确保系统运行状况可视化。这些功能为管理员提供了强大的监管和管理能力。

在系统层面，平台还集成了自动化的“故障处理”模块和“数据管理”模块。故障处理模块能够记录并响应系统运行中的问题，例如服务器崩溃、网络中断等情况，自动恢复功能为平台的高可用性提供了保障。数据管理模块则承担着数据存储、备份和归档的任务，确保用户内容和平台数据的安全。

## 5. 系统架构

### 5.1 系统软件架构图



本系统是一个面向知识共享与管理的综合平台，旨在满足用户对知识采集、管理、分享和互动的需求，主要包括以下两个大模块：管理端和用户端。每个模块包含多个子模块，具体如下：

### (1) 管理端模块

管理端模块主要负责系统的后台管理功能，确保知识共享平台的高效运营和维护。包括以下子模块：

**用户管理：** 用户管理模块负责对平台用户的信息进行维护和权限管理，主要功能包括用户的新增、编辑、删除和查询。

**帖子管理：** 帖子管理模块负责对用户发布的帖子进行审核和管理，支持查看、编辑、删除以及批量处理帖子内容。

**反馈管理：** 反馈管理模块用于接收和处理用户反馈，支持分类查看反馈问题、处理反馈状态并生成处理记录。

**评论管理：** 评论管理模块提供对用户评论的审核、隐藏和删除功能，确保平台评论区的内容健康和规范。

**合集管理：** 合集管理模块支持对知识合集的创建、分类和内容维护，为用户提供高效的知识聚合服务。

**统计相关：** 统计模块提供对用户行为和内容发布情况的统计分析，为运营决策提供数据支持，如用户活跃度统计、内容发布量统计等。

**审核管理：** 审核管理模块支持对知识内容和用户行为的审核流程管理，包括内容审批和权限调整等操作。

**举报管理：** 举报管理模块负责对用户的举报信息进行分类管理和处理，支持统计和生成处理报告。

**登录模块：** 登录模块是管理端的入口模块，负责对管理员的身份进行验证和授权，支持多种登录方式，如账号密码登录和单点登录。

### (2) 用户端模块

用户端模块面向普通用户，提供知识的获取、发布、互动及个性化管理服务。包括以下子模块：

**历史模块：** 历史模块记录用户的操作行为和内容浏览历史，支持查看、删除或按照分类进行筛选。

#### 个人中心模块：

个人中心模块是用户的个人管理区域，包含以下子功能：

**个人收藏管理：** 用户可以对喜欢的内容进行收藏，并在个人中心查看、管理收藏内容。

**个人合集管理：**用户可创建并管理知识合集，将相关内容进行分类整理。

**个人评论管理：**展示用户发表的评论，支持删除和查看评论详情。

**通知模块：**通知模块用于向用户推送平台消息，包括系统通知、评论提醒、点赞提醒等，支持查看和标记为已读。

**帖子发布模块：**帖子发布模块支持用户通过富文本编辑器发布新内容，添加图片、视频、标签等多种元素，并提供草稿保存功能。

**帖子详情模块：**帖子详情模块展示单篇帖子内容，支持用户评论、点赞、收藏、分享等操作，提供沉浸式阅读体验。

**个人主页模块：**个人主页展示用户的个人信息、动态、发布内容和收藏内容，支持个性化设置和信息编辑。

**登录模块：**登录模块负责对用户的身份进行验证和授权，支持手机号、邮箱、社交账号等多种登录方式。

**反馈模块：**反馈模块用于接收用户的意見和建议，用户可通过该模块提交反馈信息，查看反馈处理状态。

**评论模块：**评论模块允许用户对帖子内容进行评论互动，同时支持评论回复和点赞操作。

**合集相关模块：**合集模块支持用户浏览、创建和管理知识合集，将相关内容整合为一体，提升知识分类效率。

**AI 对话模块：**AI 对话模块为用户提供智能化的知识问答服务，结合用户需求推荐个性化内容。

**个人统计模块：**个人统计模块记录并展示用户的浏览、发布和互动数据，帮助用户了解自身的使用习惯和内容贡献度。

**分区模块：**分区模块按类别为知识内容进行分区展示，用户可根据兴趣快速找到目标内容。

**搜索模块：**搜索模块支持关键词和全文检索，提供精准、高效的内容定位服务。

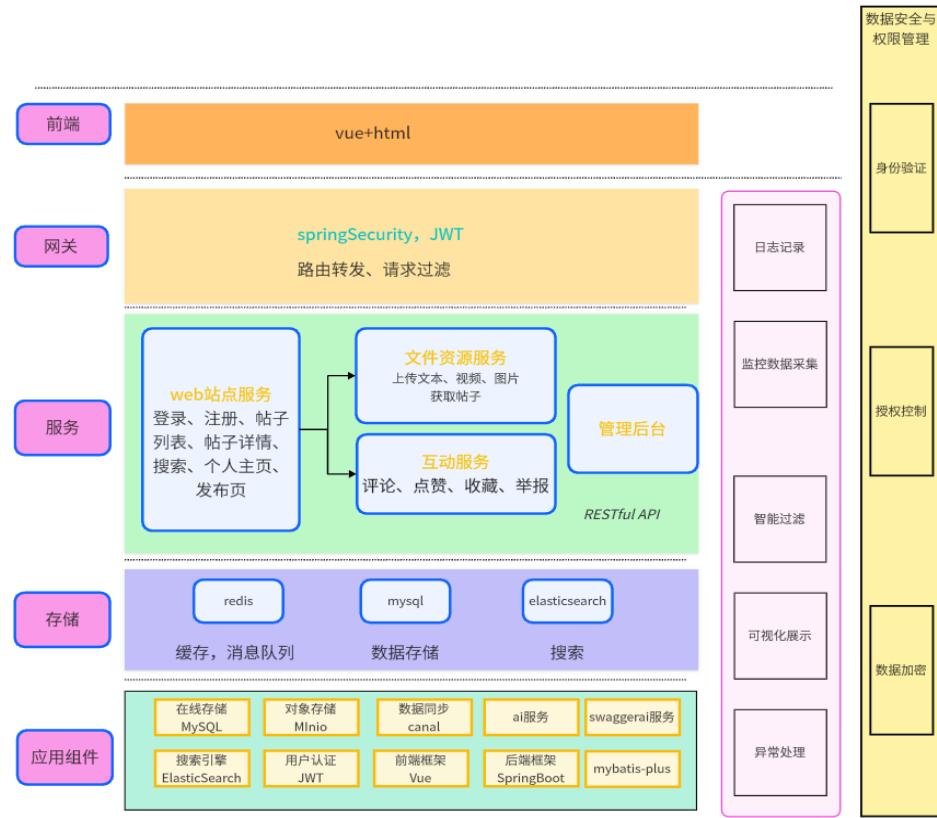
**推荐模块：**推荐模块基于用户兴趣和行为，向用户推送个性化内容，提升用户的使用体验。

**排行榜相关模块：**排行榜模块展示平台上的热门内容、点赞榜、评论榜等，帮助用户快速了解热点内容。

**首页管理模块：**首页管理模块是用户的入口模块，提供动态内容展示、推荐内容推送及快捷操作入口。

以上各模块构成了知识共享平台的核心功能，满足用户和管理员在知识管理、互动和服务等方面的需求，保障系统的高效运作。

## 5.2 系统技术架构图



以下是知识共享平台采用分层模式技术实现的详细说明，主要分为前端层、网关层、服务层、存储层和应用组件层五个层次。

### (1) 前端层

前端层是知识共享平台与用户交互的界面，负责接收用户请求、返回系统响应，并完成前端逻辑处理与视图渲染工作。展示层选用了 Vue 和 ElementUI 的技术组合，以确保开发效率与界面美观性：

**Vue:** 一个轻量级的渐进式 JavaScript 框架，支持组件化开发，提高了代码的复用性和可维护性，适合构建复杂的单页应用（SPA）。通过 Vue 的双向数据绑定功能，展示层可实现高效的动态交互效果。

**ElementUI:** 基于 Vue 的 UI 组件库，提供丰富的界面组件，如表单、表格、对话框等，方便实现统一的界面风格与交互逻辑。

此外，前端层通过引入 Axios 库与后端 API 进行通信，支持异步数据加载与缓存优化，确保页面响应速度和稳定性。

### (2) 网关层

网关层是系统的核心层，负责知识共享平台主要业务逻辑的实现，包括知识发布、搜索、管理、统计分析等功能。业务层基于 SpringBoot 框架构建，采用 Spring

Security 和 JWT (JSON Web Tokens) 进行用户认证和授权，确保安全性。实现路由转发和请求过滤，管理不同服务之间的请求路由和安全控制。

具体实现包括：

SpringBoot：简化了配置和依赖管理，提高了开发效率和运行性能。其嵌入式容器特性支持快速部署和启动，适合知识平台这种需要高频迭代的应用场景。

Spring Security：Spring 框架的核心安全模块，提供身份验证和授权功能，支持 RBAC、会话管理、密码加密等，帮助快速构建安全系统。对知识平台，可确保用户数据机密性，控制资源访问，灵活处理多用户、多权限场景。

JWT (JSON Web Tokens)：轻量级令牌，用于身份验证和信息交换。通过签名机制确保数据安全，无需服务端存储状态，适合高并发和分布式系统。对知识平台，JWT 支持无状态认证和权限校验，提升性能与用户体验。

业务逻辑实现：包括知识内容管理、用户行为分析、推荐算法、权限控制等核心功能。业务层还集成了 ApacheCamel，用于复杂的业务规则处理，如路由解析和消息过滤。

### (3) 服务层

服务层是知识共享平台的扩展层，提供灵活的功能模块以增强系统适配性和可扩展性。本平台主要提供以下服务：

Web 站点服务：提供登录、注册、帖子列表、帖子详情、搜索、个人主页和发布页等功能。

文件资源服务：支持上传文本、视频、图片，并获取帖子，管理文件资源的存储和访问。

互动服务：包括评论、点赞、收藏和举报功能，增强用户互动和社区参与度。

管理后台：通过 RESTful API 提供后台管理功能，包括监控数据采集、可视化展示、权限处理、审核处理和日志记录。

### (4) 存储层

存储层是知识共享平台的数据基础设施，负责保存业务数据、日志数据以及用户上传的文件。根据数据特性，存储层选择了多种存储方案：

关系型数据库：使用 MySQL 作为主数据库，用于存储用户数据、帖子内容、评论等结构化数据。MySQL 提供了事务支持和高可用架构，确保数据的一致性和可靠性，同时满足高频读写场景的性能需求。

缓存与消息队列：采用 Redis 作为缓存层和消息队列，负责存储热点数据、分布式锁和消息任务。Redis 的高吞吐量和低延迟特性显著提升了数据访问速度和系统响应能力。

**搜索引擎：**集成 Elasticsearch，用于存储搜索索引数据，支持全文检索和复杂查询。其分布式架构和实时分析能力满足了知识共享平台对海量搜索的高效需求。

**对象存储：**使用 MinIO 实现对象存储，用于保存用户上传的图片、视频等非结构化数据。MinIO 支持高并发访问和大文件存储，保障了数据的安全性与扩展性。

此外，存储层通过 Canal 实现数据同步，确保数据一致性与实时性，并配备了定时备份与灾备策略，防止因意外导致的数据丢失。

### (5) 应用组件层

应用组件层是知识共享平台的核心服务支撑模块，涵盖了系统功能的各个关键部分：

**ORM 框架：**使用 MyBatis-Plus 简化数据库操作，减少重复编码，同时提高开发效率和代码可维护性。

**AI 服务：**集成 AI 模型，为平台提供智能推荐、用户行为分析等功能，提升用户体验和平台运营效率。

**Swagger 服务：**集成 Swagger，为平台 API 文档的在线生成与管理提供支持，便于开发人员快速理解和使用接口。

通过整合多种组件和工具，应用组件层为平台提供了稳定高效的服务支撑，同时灵活满足了平台功能的快速扩展需求。

通过分层架构设计，知识共享平台在技术上实现了高效开发、稳定运行和灵活扩展，满足了多样化的用户需求和复杂的业务场景。各层之间功能解耦，确保系统的可维护性、可扩展性和高性能。

## 5.3 系统硬件架构

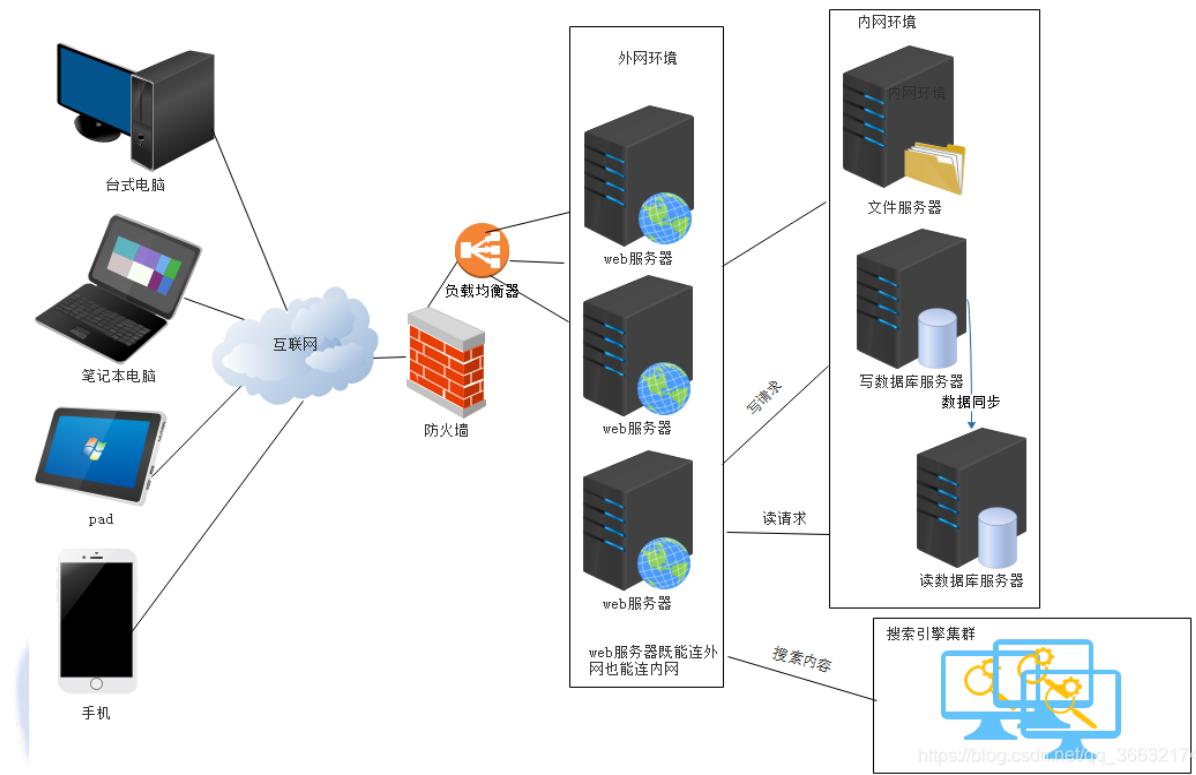
下面这张图展示了知识共享平台的系统硬件架构，清晰地划分了外网环境和内网环境，并通过网络设备实现高效的数据流转和安全保护。平台的用户可以通过台式电脑、笔记本、平板、手机等多种终端设备访问系统，这些终端通过互联网连接到平台的 Web 服务器。用户可以在平台上浏览内容、搜索知识、发表评论，这些操作都需要平台在高并发情况下保持稳定响应。

在外网环境中，负载均衡器是系统的核心组件之一，它分发来自用户的请求到多台 Web 服务器，保证流量分布均匀，并避免单点过载问题。同时，防火墙作为系统的安全屏障，保护平台免受恶意攻击，确保用户数据和平台服务的安全性。Web 服务器处理用户请求后，将复杂的写入和查询请求进一步分发到内网环境。

内网环境承载了平台的核心数据存储和管理功能。文件服务器用于存储用户上传的图片、视频等非结构化数据，与平台的对象存储功能相对应。写数据库服务器专注于处理用户的数据写入操作，比如新增帖子、评论等；读数据库服务器则负责处理高

频的查询请求，为用户提供快速的内容展示。这种读写分离的架构显著提升了数据库的性能，并通过数据同步机制（如 Canal）保证了数据的一致性。

此外，搜索引擎集群是平台的重要组成部分，它处理用户的搜索请求，为平台提供强大的全文搜索能力。在知识共享平台中，这部分由 Elasticsearch 实现，通过分布式架构支持海量数据的快速检索和精准查询。整体架构设计高效、安全、扩展性强，充分满足了知识共享平台在高并发访问、内容存储与检索上的实际需求。



## 6. 需求分配表

需求提出者	用户需求名称	用户详细描述		备注
李健	登录需求	登录	提供用户通过账号密码等方式登录平台的功能，忘记密码和重置	提供用户认证接口，支持多种认证方式如 springsecurity 等
		注册	提供新用户创建账户的功能，包括填写个人信息、设置密码等	提供用户注册接口，支持验证码机制和邮件验证

		退出登录	退出登录：提供安全的退出登录功能，保护用户信息安全	提供安全退出登录接口，管理会话和安全性检查
徐磊	数据统计与展示需求	数据统计	数据统计：在首页显示平台的关键统计数据，如知识条目数量、活跃用户数等	提供统计数据获取接口，显示关键指标
		数据展示	以图表或列表形式展示统计数据，便于用户直观理解	提供数据展示接口，支持图表生成
徐磊	关键字搜索需求	搜索框	提供一个易于使用的搜索框，支持即时搜索建议	提供基础搜索接口，支持自动补全和热门搜索推荐搜索历史
		搜索优化	通过索引和查询优化提升搜索性能	提供优化搜索性能的接口，初步集成 ElasticSearch
李健	知识分类展示需求	分类浏览	允许用户按分类或标签浏览知识库	提供分类浏览接口，支持分类和标签管理
		分类管理	管理员可添加、编辑和删除分类	提供分类管理接口，支持权限控制和日志记录
李健	知识详情展示需求	内容展示	详细展示知识内容，包括文本、图片、视频等	提供知识详情展示接口，支持多媒体内容
		社交互动	提供收藏、分享、点赞和评论功能	提供社交互动接口，支持收藏、分享、点赞和评论，分享链接等形式
徐力行	知识采编模块需求	引用管理	管理知识之间的引用关系	提供知识引用管理接口，支持引用追踪

		内容编辑	使用富文本编辑器录入和编辑知识内容	提供知识编辑接口，支持富文本编辑
		来源记录	记录知识的原始来源信息	提供来源信息管理接口，支持来源记录
郭灵杰	个人中心模块需求	我的发布	查看自己发布的所有知识	提供用户发布历史接口，支持发布管理
		我的评论	查看自己发表的所有评论	提供用户评论历史接口，支持评论管理
		我的收藏	查看自己收藏的所有知识	提供用户收藏接口，支持收藏管理
		我的点赞	查看自己点赞的所有知识	提供用户点赞记录接口，支持点赞管理
		我的草稿	管理和保存未完成的知识草稿	提供草稿管理接口，支持草稿保存和恢复
徐磊	知识流程模块需求	采编流程	定义从采集到编辑的知识处理流程	提供知识采编流程接口，支持多步骤审批
		更新流程	定义知识更新的审核和发布流程	提供知识更新流程接口，支持版本管理和变更记录
		删除流程	定义知识删除的审批和执行流程	提供知识删除流程接口，支持软删除和硬删除选项
		发布流程	定义知识发布的审核和上线流程	提供知识发布流程接口，支持预览和测试
		反馈流程	收集并处理用户对知识的反馈意见	提供反馈收集接口，支持反馈处理机制

徐力行	ElasticSearch 使用需求	采用 ElasticSearch 提升平台搜索性能	提供优化搜索性能的接口，初步集成 ElasticSearch
陈远飞	客户端和管理端划分需求	明确客户端和管理端的不同功能和权限	提供不同角色的接口，支持权限控制
陈远飞	权限控制需求	设置客户与管理员之间的权限控制机制，确保操作安全	提供基于角色的访问控制接口，支持权限管理
郭灵杰	知识审核需求	对共享的知识进行审核，确保内容质量和合规性	提供内容审核接口，支持审核标准和流程，添加违禁字检测机制
徐磊	接入对话式 AI 简化搜索需求	使用 AI 技术优化用户体验，简化搜索过程	提供 AI 辅助搜索接口，支持自然语言处理，提取关键词进行搜索
陈远飞	数据展示与监控需求	提供全面的数据展示和实时监控功能，帮助管理者掌握平台动态	提供数据展示和监控接口，支持实时数据流和仪表板
徐磊	消息通知需求	考虑增加用户和平台间消息通知机制	提供消息通知接口
徐磊	用户使用手册需求	考虑帮助用户更好的使用平台	提供帮助手册查看接口

## 7. 运行环境

### 7.1 设备

序号	设备名称	参考配置	单位	数量	说明
1	服务器	32CPU, 128G 内存, SSD 存储	台	3	新增，建议标准配置
2	数据库服务器	16CPU, 64G 内存, 高速硬盘	台	2	新增，用于数据存储

3	交换机	4810/100/1000 自适应端口	台	2	新增，网络管理
4	管理终端	固定终端 10, 移动终端 10	台	20	新增，支持平台管理

## 7.2 支持软件

**操作系统:** Windows 或 MacOS。

**应用环境:** Java8+/Node.js 版本、Tomcat、Nginx。

**数据库:** MySQL8.0+。

**搜索引擎:** ElasticSearch7.x+（可选）。

**缓存:** Redis5.x+。

**消息同步:** canal。

## 7.3 接口

系统采用 RESTful 风格的 API 设计，使用标准 HTTP 方法进行通信。所有接口都需要进行 JWT 认证（除了公开接口）。

**接口调用地址:**

`http://localhost:8080/api/client/post/detail/1`

节点	节点名称	数据类型	说明
<code>code</code>	状态码	Integer	响应状态码，200 表示成功
<code>message</code>	消息	String	响应描述信息
<code>data</code>	数据	Object	响应数据主体

节点	节点名称	数据类型	说明
<code>code</code>	错误码	Integer	错误状态码
<code>message</code>	错误信息	String	错误描述信息
<code>data</code>	错误详情	Object	详细错误信息

节点	节点名称	数据类型	最大长度	节点位置	说明
<b>Token</b>	认证令牌	String	128	请求报文	服务编码在服务注册时，由 OSS 能力集成平台定义并保证全局唯一；服务编码在接口调用时，由客户端写入请求报文头；不能为空

<b>Content-Type</b>	Msgid	String	128	请求报文	请求唯一标识在接口调用时由客户端生成并写入请求报文头；客户端生成请求唯一标识时需要保证唯一性；不能为空
<b>授权码</b>	authcode	String	512	请求报文	授权码由 OSS 能力集成平台生成并交付给客户端，一般是请求者的系统编码。授权码在接口调用时由客户端写入请求报文头不能为空
<b>服务版本号</b>	version	String	4	请求报文	服务版本号由 OSS 能力集成平台生成并交付给客户端，为从 1 到 9999 的正整数。服务版本号在接口调用时由客户端写入请求报文头，默认为 0。

节点	节点名称	数据类型	最大长度	节点位置	说明
<b>错误码</b>	errcode	String	16	异常响应报文	参见《错误代码表》
<b>错误信息</b>	errmsg	String	128	异常响应报文	参见《错误代码表》
<b>错误详情</b>	errdetail	String	4096	异常响应报文	提供异常堆栈信息或异常处理建议

错误码	说明
200	成功
400	请求参数错误
401	未授权
403	禁止访问
404	资源不存在
500	服务器内部错误

## 7.4 控制

**权限控制：**通过角色权限对应用服务器的管理端口、数据库访问进行限制。

**网络安全：**部署防火墙、安全组策略，屏蔽不必要的端口和 IP。

**数据安全：**对关键数据进行加密存储或传输；考虑使用 SSL/TLS 加密。

## 8. 人工处理过程

**数据审核:** 当用户提交的知识内容需要人工审核时，管理员在后台审核并决定发布或驳回。

**异常处理:** 当系统出现无法自动恢复的故障或流程异常时，运维人员需进行人工介入，查看日志并采取相应的故障转移方案。

**系统配置与更新:** 运维或开发人员定期对系统版本、配置、依赖进行更新，并在灰度或测试环境验证后再进行生产环境发布。

## 9. 尚未解决的问题

1. **多语言支持:** 针对海外用户或多语言环境的需求尚未规划，需要进一步评估后期扩展性。

2. **权限管理的精细化:** 对知识内容的更细粒度权限暂时未全面展开，需要进一步调研和设计。

