



《基于大小模型协同的在线文 档编辑器系统》 项目概要设计

V1.2.0

神奇三人组



版本历史

版本/状态	作者	参与者	日期	备注
v1.0.0	神奇三人组	李建康、姬舜赢、陈怡	2024.6.25	创建
v1.1.0	神奇三人组	李建康、姬舜赢、陈怡	2024.6.26	修改
v1.2.0	神奇三人组	李建康、姬舜赢、陈怡	2024.6.27	完成



目录

第一部分 引言.....	6
一、编写目的	6
二、读者对象	6
三、术语与缩写解释	6
1. RTE	6
2. LLM	7
3. RAG	7
4. OCR	7
5. ASR	8
四、参考资料	8
第二部分 项目概述.....	8
一、项目描述	8
二、项目功能描述	9
1. 用户业务	10
1.1 平台用户功能.....	10
1.2 文档编辑功能	11
1.3 AI 模型交互功能.....	12
1.4 协同编辑功能	13
第三部分 设计约束.....	15
一、需求约束	15
1. 本系统应当遵循的技术标准	15
2. 软、硬件环境标准	15
3. 接口/协议标准	15
4. 用户界面标准	15
5. 软件质量	15
二、隐含约束	16
第四部分 智能编辑器设计.....	17



一、安全设计	17
二、系统相关业务流程	17
1. 授权认证流程	17
2. 个人信息修改流程	18
3. 管理文档空间流程	19
4. 文档编辑流程	20
5. 多媒体数据插入流程	21
6. 文档导出为 PDF 流程	22
7. 权限管理流程	23
8. AI 对话流程	24
9. 智能润色流程	24
10. 多模态数据输入流程	25
11. 智能格式排版流程	26
12. 文档共享流程	27
13. 实时协作编辑流程	28
14. 协作沟通流程	29
三、业务功能概要结构	30
1. 个人空间	30
2. 文档编辑	31
3. AI 交互	32
4. 协同编辑	33
四、模块定义	35
第五部分 E-R 实体设计	36
一、实体结构图	36
二、实体描述	36
1. 用户信息实体描述	36
2. 文档信息实体描述	37
3. 文档列表信息实体描述	37



4. 协作权限	37
第六部分 总体设计.....	38
一、系统逻辑结构设计	38
二、物理架构设计	38
1. 客户端	39
2. 负载均衡服务器	39
3. web 服务器.....	39
4. 高速缓存数据库	39
5. 存储数据库	39
6. 模型库	39
三、技术架构设计	40
1. 星河社区独占机器服务部署	40
2. Nginx	40
3. MySQL 存储数据库.....	40
4. Milvus	41
5. Redis 高速缓存数据库.....	41
6. Django MTV 模式.....	42
第七部分 用户界面设计.....	43
一、桌面布局设计	43
1. 注册界面	43
2. 登陆界面	44
3. 用户个人信息界面	44
4. 文档空间界面	45
5. 文档编辑界面	45
二、业务界面风格展示	46
第八部分 运行环境和部署.....	46
一、运行环境	46
1、客户机环境:	46



2、服务器：	46
3、 开发环境：	47
二、系统性能要求	47

第一部分 引言

一、编写目的

编写本文的主要目的是把需求分析得到的用例模型转换为软件功能结构和数据结构。由前面的需求分析得到了系统得到基本需求，要实现整个系统，需要对系统的需求进行设计。本概要设计说明书从程序系统的设计考虑，包括系统的项目概述、设计约束、方案设计、E-R 实体设计、总体设计、用户界面设计，运行环境和部署等内容，以向整个设计期提供关于程序系统的逻辑和数据功能实现方式的具体描述，从而作为程序详细设计或编码的基础。

本设计是指导详细设计和项目实施的重要指导性文件，也是进行系统集成测试和重要依据。

二、读者对象

该文档的读者为用户代表、管理员代表、软件分析人员、开发管理人员和设计人员。

三、术语与缩写解释

1. RTE

富文本编辑器（Rich Text Editor，RTE）是一种可内嵌于浏览器，所见即所得的在线文本编辑器。它提供类似于 Office Word 的编辑功能，方便那些不太懂 HTML 用户使用，



富文本编辑器的应用非常广泛，它的历史与图文网页诞生的历史几乎一样长。

2. LLM

大型语言模型（LLM）是一类基础模型，经过大量数据训练，使其能够理解和生成自然语言和其他类型的内容，以执行各种任务。本项目使用的大语言模型是由百度开发的文心一言（ERNIE Bot）模型。文心一言（ERNIE Bot）是百度全新一代知识增强大语言模型，文心大模型家族的新成员，能够与人对话互动，回答问题，协助创作，高效便捷地帮助人们获取信息、知识和灵感。文心一言是知识增强的大语言模型，基于飞桨深度学习平台和文心知识增强大模型，持续从海量数据和大规模知识中融合学习具备知识增强、检索增强和对话增强的技术特色。

3. RAG

大型语言模型（LLM）相较于传统的语言模型具有更强大的能力，然而在某些情况下，它们仍可能无法提供准确的答案。为了解决大型语言模型在生成文本时面临的一系列挑战，提高模型的性能和输出质量，研究人员提出了一种新的模型架构：检索增强生成（RAG, Retrieval-Augmented Generation）。该架构巧妙地整合了从庞大知识库中检索到的相关信息，并以此为基础，指导大型语言模型生成更为精准的答案，从而显著提升了回答的准确性与深度。

4. OCR

OCR 识别（Optical Character Recognition，光学字符识别）是一种技术，用于将印刷或手写文本从图像、扫描件或照片中提取并转换为可编辑和搜索的数字文本。该技术使用图像处理和模式识别算法，将文本从静态的视觉形式转换为计算机可处理的数据，从而实现自动化的数据输入和文档管理。OCR 识别广泛应用于文档数字化、数据录入自动化以及信息检索等领域。



5. ASR

语音识别也称为自动语音识别（ASR）、计算机语音识别或语音转文本，它能够让一个程序来处理将人类语音转变为书面格式的任务。语音识别专注于将语音从口头格式转换为文本，进而实现将文本输入到大模型中的功能。

四、参考资料

《基于大小模型协同的在线文档编辑器系统》需求规格说明书

第二部分 项目概述

一、项目描述

在数字化时代，传统的文档编辑工具往往只能处理单一模式的信息，如纯文本或简单的表格，这种方式在处理大量的多模态信息时显得效率低下，用户体验较差。无论是撰写项目书、设计分工表，还是记录会议纪要，用户在使用过程中常常感到力不从心。

目前，尽管一些文档编辑器已经引入了 AI 功能，但这些功能通常仅限于基础的纠错和简单的文本处理，无法满足用户对图片、表格、代码和语音视频等多模态信息的全面处理需求。这不仅导致了大量时间的浪费，也因为功能的局限性，用户在使用时的效率低、准确性差，甚至影响了工作和学习的质量。

随着信息内容和复杂度的增加，用户对文档编辑工具的需求也变得更加多样和苛刻。他们希望能够拥有一个强大的工具，不仅可以自动化地处理多种类型的信息，还能通过智能化的方式提升工作效率，优化用户体验。同时，随着团队协作和跨设备工作的增加，用户还需要工具在数据同步和安全性方面具备更高的能力。

为了更好地满足这些需求，我们决定开发一个全新的文档编辑器系统。这个系统将结合生成式 AI 技术和多模态信息处理能力，帮助用户高效地编辑和创作内容。用户可以利用



大模型实现对文章的智能排版，自动调整格式和布局，并通过多种模板选择丰富文档的展示效果。同时，该系统将支持多模态输入和理解，自动提取和组织结构化知识，构建用户专属的知识库，提升内容的丰富度和准确性。

此外，该文档编辑器还将支持多人协作和云同步功能，确保用户在不同设备上都能保持数据的一致和安全。为了降低系统的部署和维护成本，我们将使用百度飞浆星河社区平台进行项目的开发和管理。通过对这一项目的开发，我们不仅将学习和掌握大模型与小模型协同工作的原理和流程，还将深入了解企业级软件项目的开发和管理流程，最终实现一个基于百度 ERNIE Bot SDK 的在线文档编辑器系统。

二、项目功能描述

用户业务主要包括：平台用户功能、文档编辑功能、AI 模型交互功能、协同编辑功能。

1. 平台用户功能主要包括：

授权认证、个人信息修改、管理文档空间

2. 文档编辑功能

文档编辑、多媒体数据插入、文档导出为 PDF、权限管理

3. AI 模型交互功能

AI 对话、智能润色、多模态数据输入、智能格式排版

4. 协同编辑功能

文档共享、实时协作编辑、协作沟通



1. 用户业务

1.1 平台用户功能

1.1.1 授权认证

授权认证功能业务主要是用户向平台发送请求获取编辑器系统开放的各项功能的使用权限。授权信息主要分为：

- 1) 用户名称
- 2) 密码
- 3) 登陆时间

1.1.2 个人信息修改

个人信息修改功能主要是用户点击个人信息栏，选择相应的个人信息进行添加或修改。个人信息主要分为：

- 1) 用户名称
- 2) 密码
- 3) 用户邮箱
- 4) 手机号码
- 5) 其他

1.1.3 管理文档空间

用户进入平台后可浏览当前自己空间中的文档信息。文档信息主要分为：

- 1) 文档名称
- 2) 最近修改时间
- 3) 创建者用户名



1.2 文档编辑功能

文档编辑功能业务主要是用户使用在线文档编辑系统编写属于自己的文档，包括以下功能：

1.2.1 文档编辑

文档编辑功能业务主要是用户在文档编辑区键入文本来编辑文档，同时可以使用系统提供的工具对文档中的字体或格式进行修改。

信息主要分为：

- 1) 文本内容信息
- 2) 文本历史信息

1.2.2 多媒体数据插入

多媒体数据插入功能业务主要是用户可以使用系统提供的插入功能，上传图片、音频或视频等多媒体文本数据到系统中，系统解析后插入到对应的文档位置中去。

信息主要分为：

- 1) 图片数据信息
- 2) 音频数据信息
- 3) 视频数据信息
- 4) 文档信息

1.2.3 文档导出为 PDF

文档导出为 PDF 功能业务是用户可以通过系统将编辑好的文档导出为 PDF 文件，并下载到本地。

信息主要分为：

- 1) 文档内容信息



1.2.4 权限管理

权限管理功能业务是用户可以对文档的编辑状态进行管理。

状态主要分为：

- 1) 可修改
- 2) 只读

1.3 AI 模型交互功能

平台的 AI 模型交互功能业务主要是用户通过划句选择或输入问题到对话框与 AI 交互，AI 给出编辑优化建议。包括如下功能：

1.3.1 AI 对话

用户可通过界面旁边 AI 对话框支持用户与 AI 模型的直接对话，用户可以通过此对话框输入查询问题或求助信息。

用户向 AI 发起请求的信息主要包括：

- 1) 输入的问题

AI 向用户返回的结果主要包括：

- 1) 回答结果

1.3.2 智能润色

用户可以向 AI 模型请求对一段文字进行智能修改，AI 模型根据需要为用户返回相应结果。

用户向 AI 发起请求的信息主要包括：

- 1) 待优化文本范围
- 2) 待优化文本内容
- 3) 修改类型



AI 向用户返回的结果主要包括：

- 1) 优化后文本范围
- 2) 优化后文本内容

1.3.3 多模态数据输入

用户可以上传图片、PDF 文件或视频音频至系统，系统使用 OCR 技术或者语音识别技术，提取文本返回给用户。

用户传入的信息包括：

- 1) 需处理的文件（图片、PDF、视频、音频等）

AI 模型返回的信息：

- 1) 文件中提取出的文字信息

1.3.4 智能格式排版

系统识别文档中的结构元素（如标题、正文、引用等），并根据选择的文档模板自动应用相应的格式化规则。

用户传入模型的信息：

- 1) 文档内容
- 2) 用户选择的模板

模型返回的信息：

- 1) 处理后的文档格式

1.4 协同编辑功能

1.4.1 文档共享

文档共享功能业务主要是用户可以将自己的文档共享出去，邀请其他用户协同编辑该份文档。



信息主要包括：

- 1) 文档内容信息
- 2) 文档共享链接信息
- 3) 协作用户信息
- 4) 用户的编辑权限信息

1.4.2 实时协作编辑

实时协作编辑功能业务主要是不同用户在线实时编辑同一份文档，系统将实现同步更新编辑结果给所有参与协作的用户。

信息主要包括：

- 1) 文档内容信息
- 2) 协作用户信息
- 3) 用户编辑权限信息
- 4) 用户编辑时间信息
- 5) 用户编辑内容信息

1.4.3 协作沟通

协作沟通功能业务主要是不同用户，使用系统提供的评论和批注功能，在文档上留下批注和评论，系统将把批注和评论同步给所有参与协作的用户。

信息主要包括：

- 1) 文档内容信息
- 2) 协作用户信息
- 3) 用户编辑权限信息
- 4) 用户批注内容信息



- 5) 用户批注时间信息
- 6) 用户评论内容信息
- 7) 用户评论时间信息

第三部分 设计约束

一、需求约束

1. 本系统应当遵循的技术标准

数据命名的规则遵循《Objective-C Style Guide》中相关的规定。

参考层级编码体系结构规范。

2. 软、硬件环境标准

使用 Pycharm 或 VSCode 开发环境、使用 Python 3.8 及以上版本、Django 框架、MySQL8.x 为优选数据库；

要求 Windows7 及以上版本的系统且 4G 以上内存要求，最好是 8G 内存；

较高版本的 Microsoft Edge 浏览器或 Google Chrome，无兼容其他浏览器计划。

3. 接口/协议标准

以浏览器为系统接口，基于 HTTP 协议，实现 B/S 间请求处理及数据通信。

4. 用户界面标准

可以使用原生 HTML5 为界面实现标准，推荐使用 JQuery 或 Vue 等前端框架作为界面实现标准，要求进行界面美工优化。

5. 软件质量

1) 正确性

系统必须业务能够被正确处理；

2) 健壮性

系统应能够 7*24 小时无故障运行

3) 效率性

系统可以支持 30 个终端同时发起业务，处理业务的时间不超过 30 秒钟；AI 响应时间不超过 1 分钟。

4) 易用性

界面应采用图形化操作方式，便于业务人员操作；

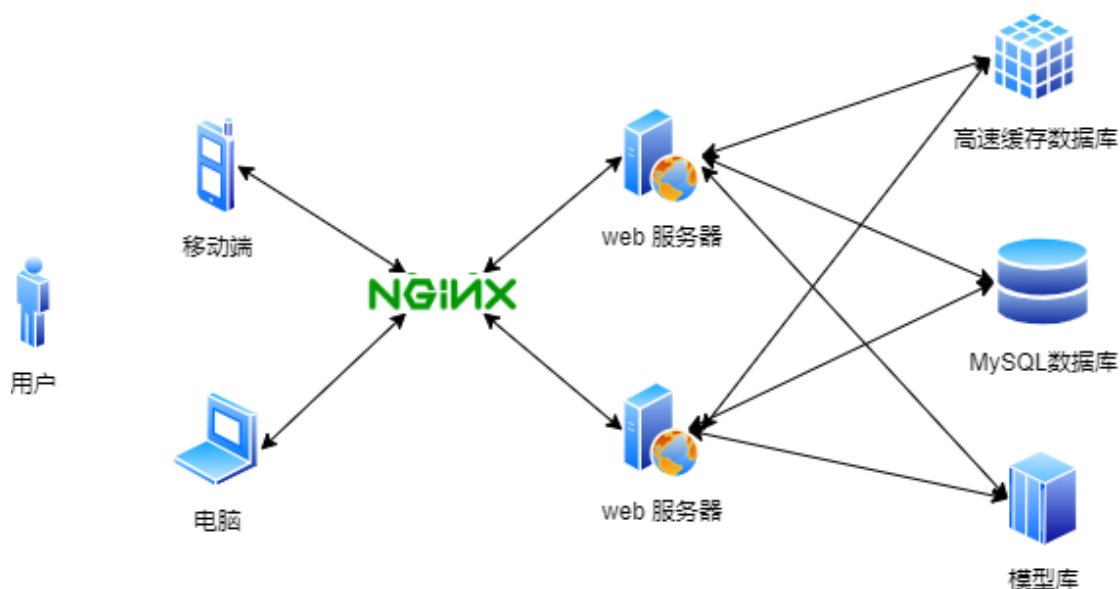
5) 安全性

豹纹中的关键数据域以密文的方式传输

6) 可拓展性

应该充分考虑到将来业务的修改或增加，避免需求变更时大规模修改程序。

7) 网络体系结构



二、隐含约束

- 1) 用户具有基本的业务技能和基本的电脑知识，对我们提供的操作界面应保证他们可

以无障碍的操作；

2) 最终软件系统可以流畅运行在 Windows7+版本以上的客户 OS 中；且客户会在开发方要求的浏览器中，运行该软件；

3) 最终交付软件系统，只是用以帮助客户解决实际问题，对人为故意破坏行为以及数据恶性更新行为，不承担责任

4) 管理员可以对信息进行正确的判断，不能删除符合要求的线索和找寻通知，我们默认管理员遵守职业操守，对所接触到的公司数据，进行无条件保密，并为此承担相应的法律。

第四部分 智能编辑器设计

一、安全设计

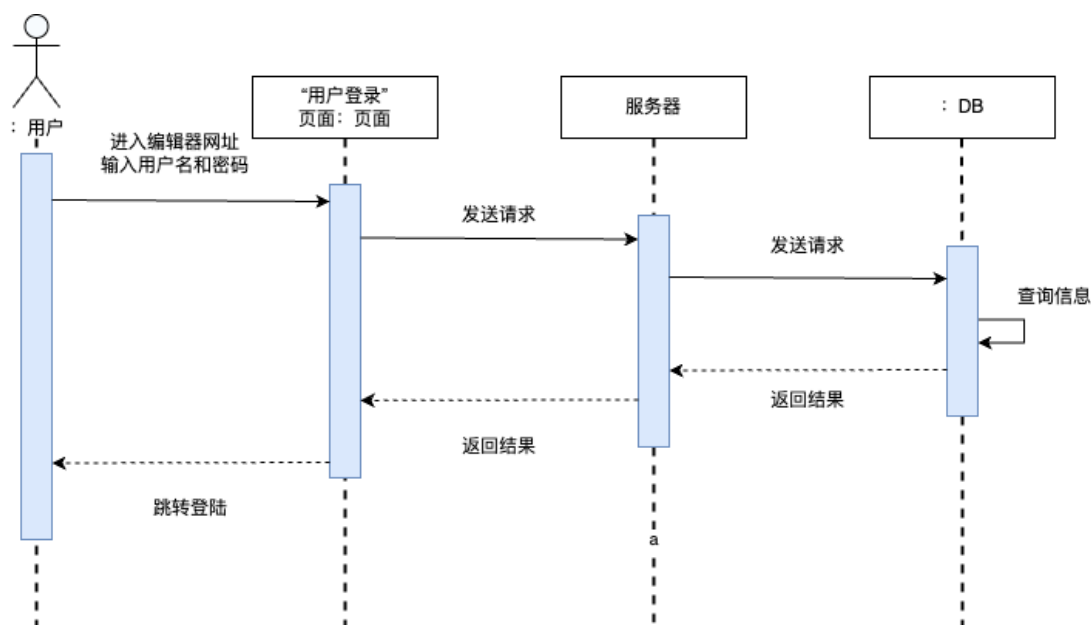
用户通过自身信息以及账号密码进行注册和登录验证。

系统采用 MySQL 数据库。

AI 模型输出符合法律规范。

二、系统相关业务流程

1. 授权认证流程



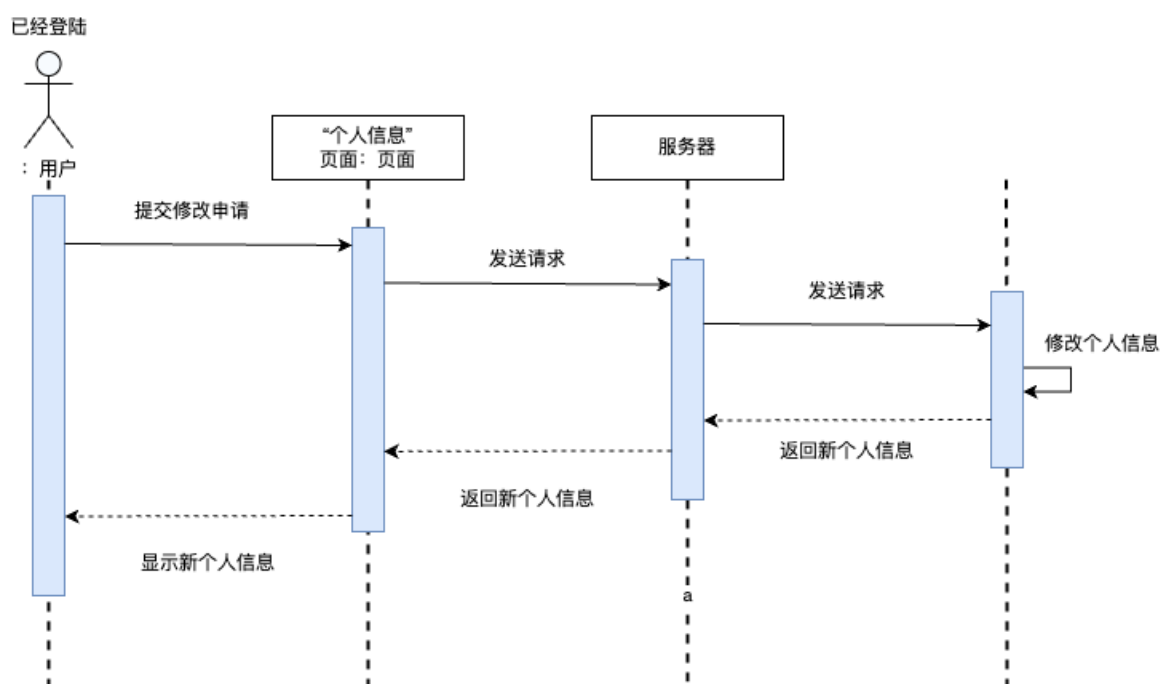


说明:

授权认证流程: 输入信息, 查询信息, 返回登陆状态, 主要流程如下:

- 1) 进入编辑器网址, 输入用户名和密码, 如果注册还需填写其他个人信息
- 2) 点击登陆或者注册按钮发送请求
- 3) 登陆成功, 跳转到编辑器个人页面
- 4) 登陆失败, 提示用户信息错误, 清空密码
- 5) 注册成功, 提示用户注册成功并跳转到登陆页面
- 6) 注册失败, 提示用户邮箱或手机号被使用

2. 个人信息修改流程



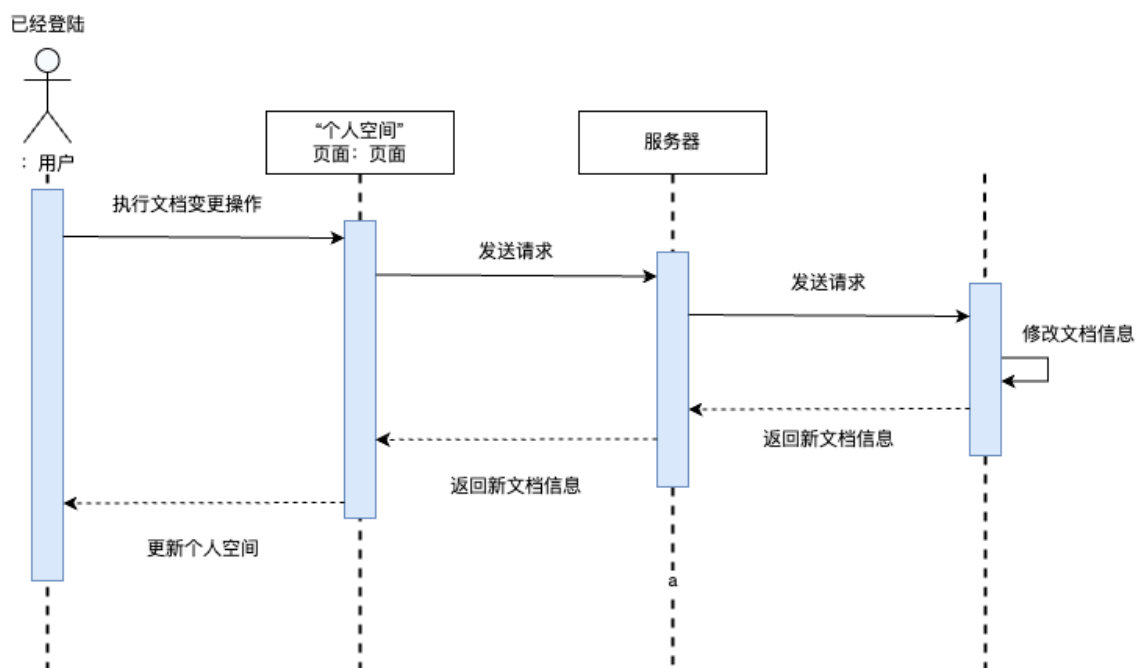
说明:

个人信息修改流程: 修改个人信息, 发送修改申请, 更新个人信息, 返回新个人信息, 主要流程如下:

- 1) 点击个人信息旁修改按钮并填写新个人信息
- 2) 点击提交发送修改申请
- 3) 数据库进行个人信息更新

- 4) 更新成功, 返回新个人信息并更新显示
- 5) 修改失败, 提示用户个人信息修改失败

3. 管理文档空间流程

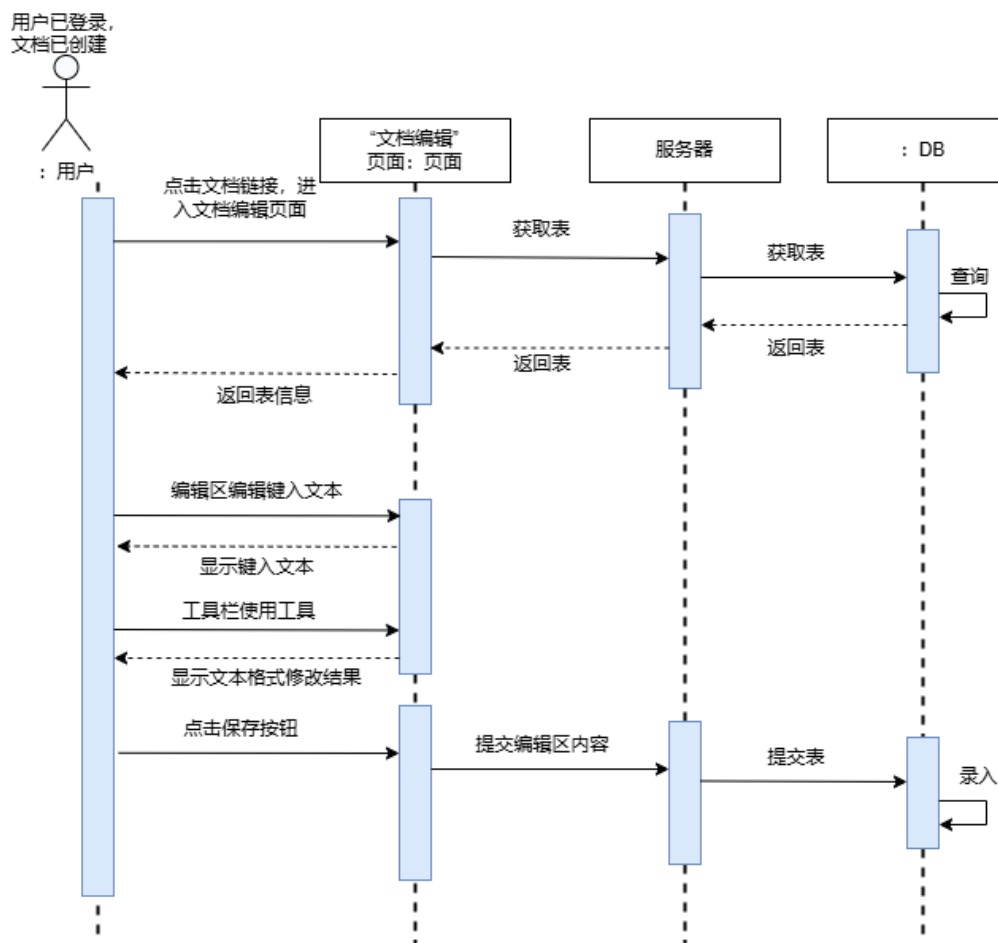


说明:

管理文档空间流程: 创建, 修改或删除文档, 发送更改请求, 修改文档信息, 返回修改结果, 主要流程如下:

- 1) 点击创建文档按钮或右键文档进行修改或删除
- 2) 点击确认发送更改请求
- 3) 数据库处理文档信息增删修改
- 4) 更改成功, 更新用户个人页面显示
- 5) 更新失败, 提示用户更改失败

4. 文档编辑流程

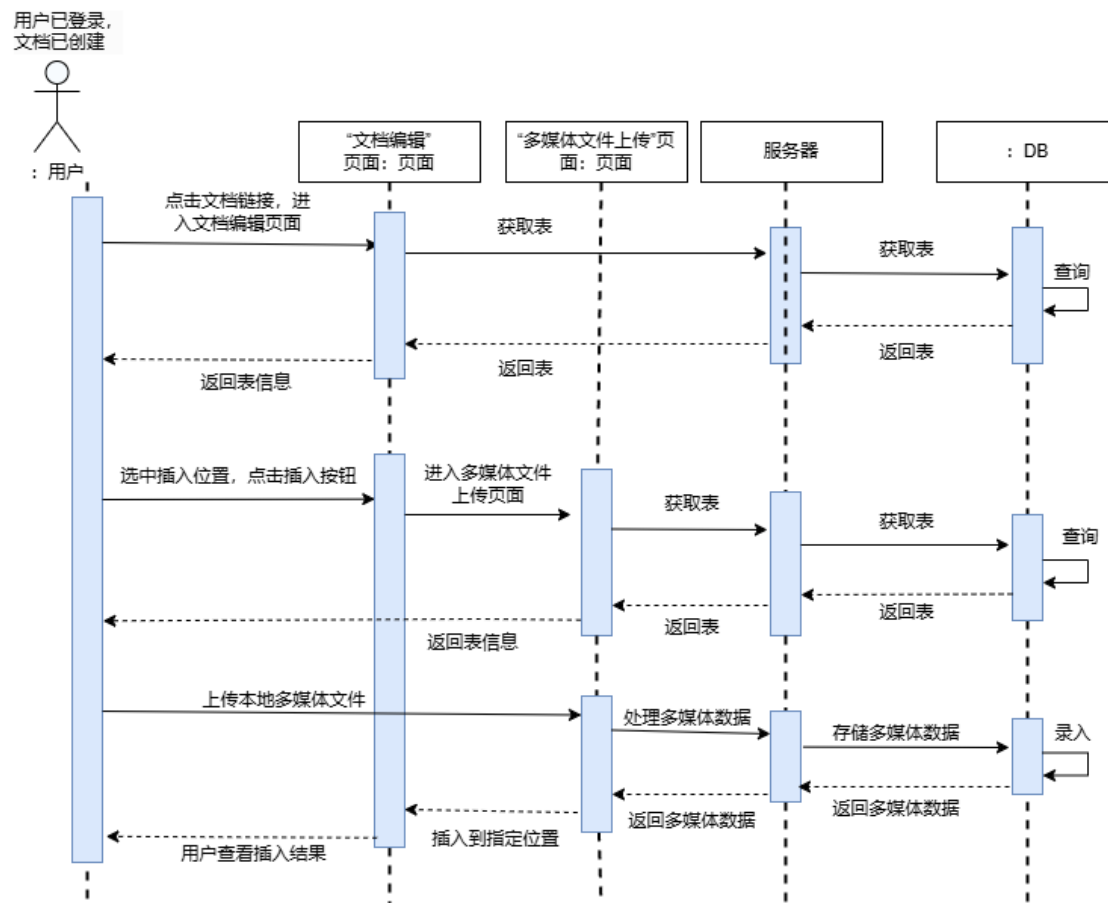


说明:

文档编辑流程主要包括进入“文档编辑页面”和编辑文档，这个过程包括请求、编辑、保存文档内容。主要流程如下（以下流程默认用户已经登陆并创建好文档）：

- 1) 点击文档链接
- 2) 显示文档编辑页面所有内容
- 3) 在编辑区内键入文本
- 4) 使用工具栏工具调整编辑区文本格式
- 5) 保存文档内容

5. 多媒体数据插入流程

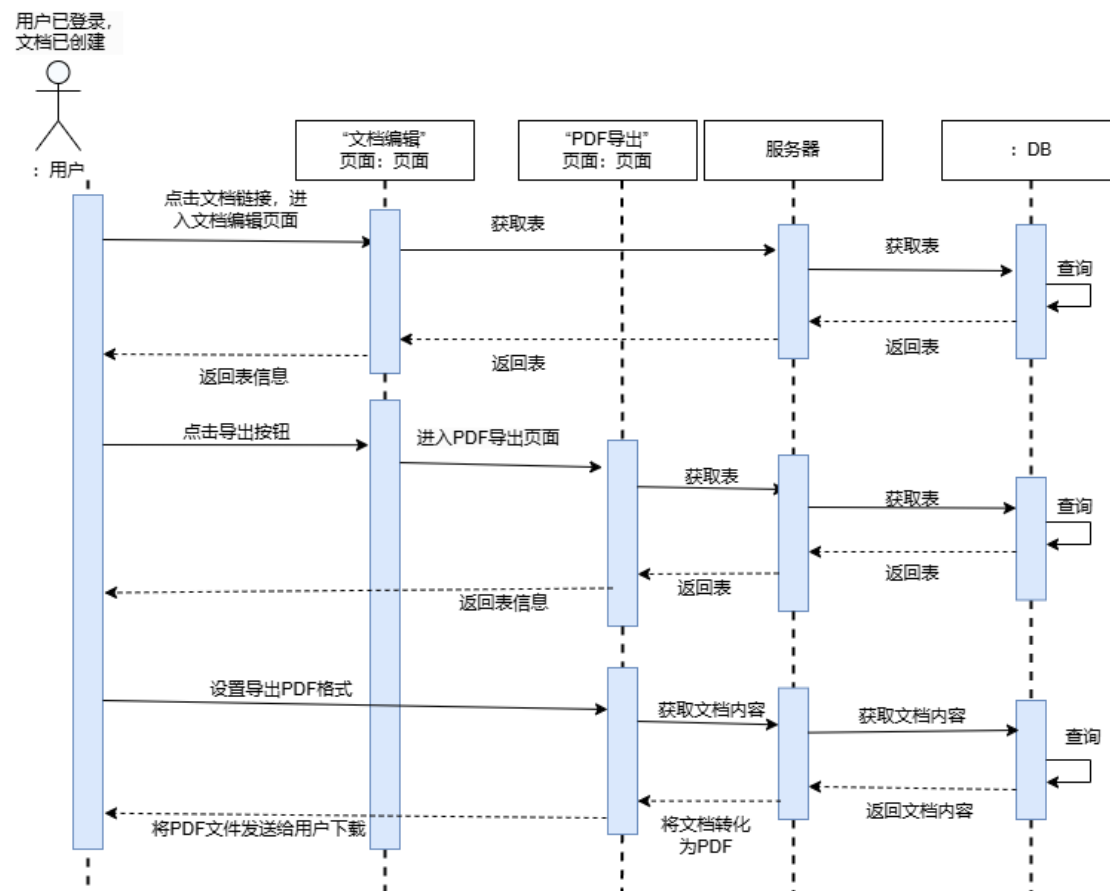


说明：

多媒体数据插入流程主要包括进入“多媒体文件上传页面”、上传多媒体文件和插入文档，这个过程包括请求、上传、编辑、保存多媒体数据。主要流程如下（以下流程默认用户已经登陆并创建好文档）：

- 1) 选中文本插入位置
- 2) 点击工具栏中插入按钮
- 3) 找到需要插入的多媒体文件类型
- 4) 上传本地多媒体文件
- 5) 文件数据存储到数据库
- 6) 将文件插入到文件对应位置

6. 文档导出为 PDF 流程

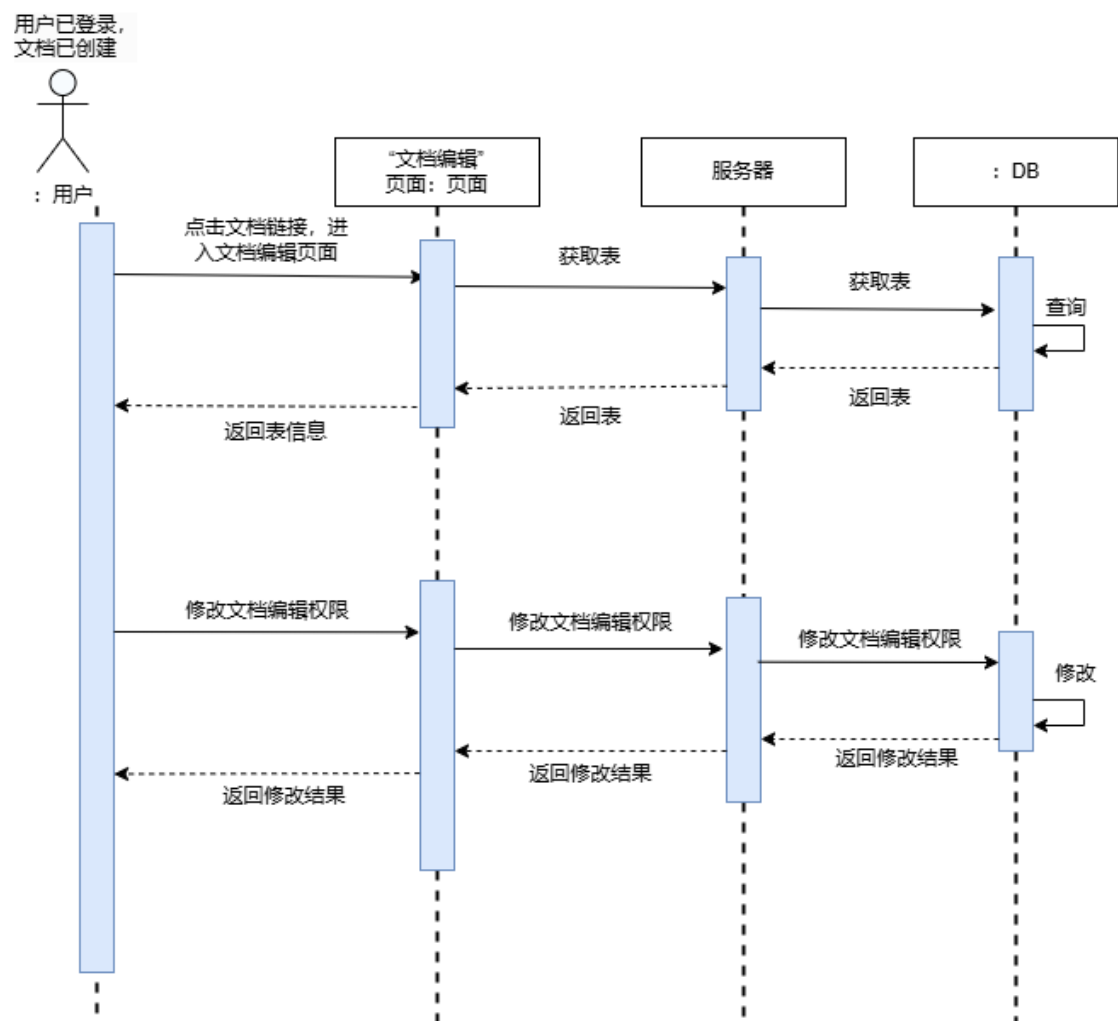


说明:

文档导出为 PDF 流程主要包括进入“PDF 导出页面”、将文档内容转换为 PDF、将 PDF 导出到用户本地设备，这个过程包括请求、处理、返回文件。主要流程如下（以下流程默认用户已经登陆并创建好文档）：

- 1) 用户点击导出按钮
- 2) 设置导出的 PDF 格式
- 3) 服务器请求导出文档数据
- 4) 服务器将文档转换为 PDF 格式
- 5) 服务器将转换后的 PDF 文件发送给用户下载

7. 权限管理流程

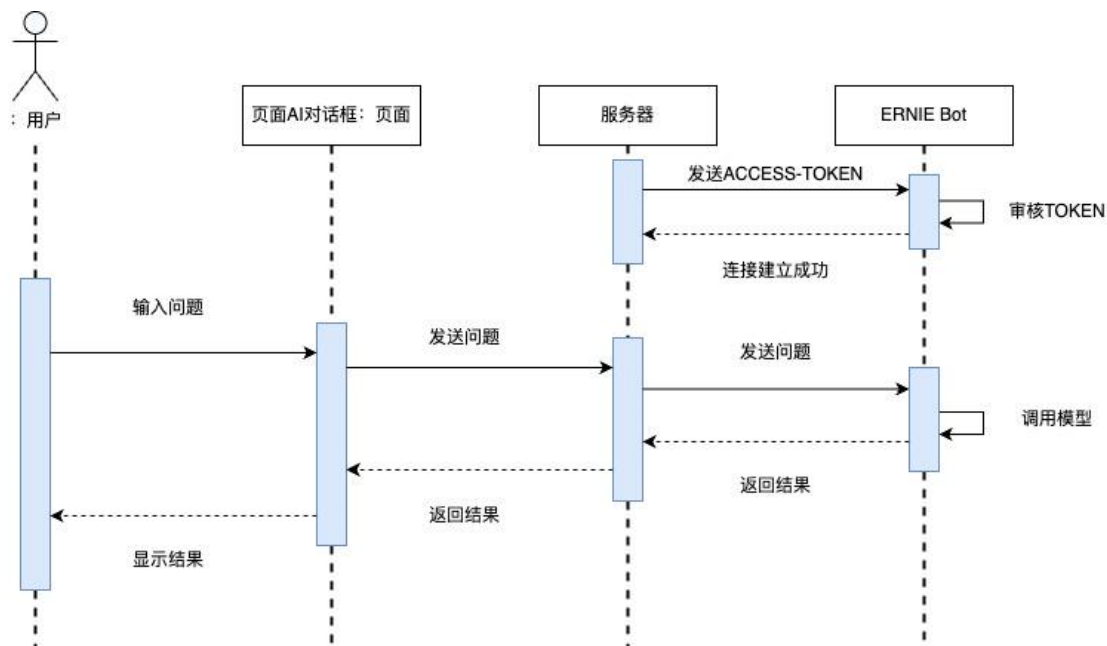


说明：

文档导出为 PDF 流程主要包括进入“文档编辑页面”和设置文档编辑权限，这个过程包括请求、修改编辑权限，主要流程如下（以下流程默认用户已经登陆并创建好文档）：

- 1) 用户点击权限管理按钮
- 2) 修改文档编辑权限

8. AI 对话流程

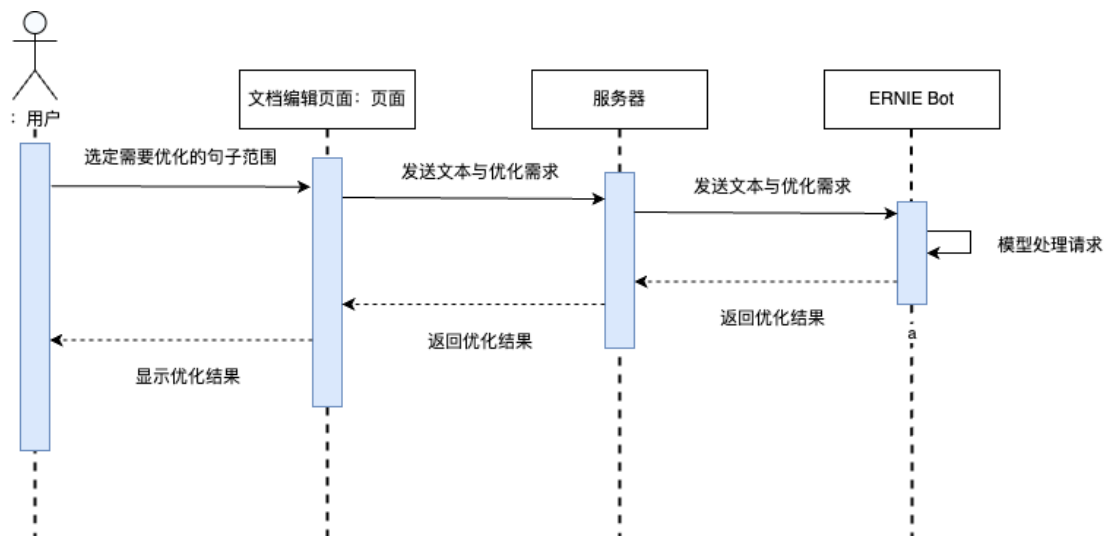


说明:

AI 对话流程由普通用户为业务主体，主要包括输入问题和 AI 回答的结果。主要流程如下：

- 1) 在 AI 对话框的输入框内输入问题
- 2) 点击“发送”
- 3) 对话框中显示 ERNIE Bot 的回答

9. 智能润色流程



说明：

智能润色流程由普通用户为业务主体，主要包括用户划句和 AI 处理。主要流程如下：

1) 用户选择某个文档，进入文档编辑页面

2) 划出需要处理的句子

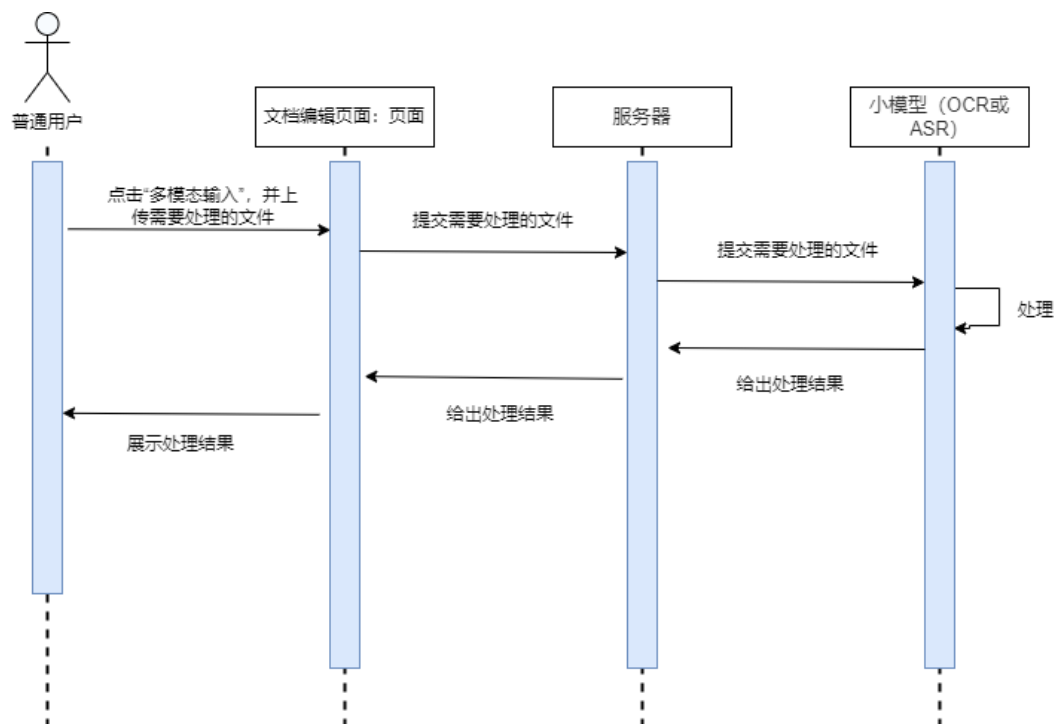
3) 显示处理类型列别：

- a. 摘要
- b. 修饰
- c. 续写
- d. 病句改写
- e. 翻译

4) 点击某个优化类型

5) 显示优化后的结果

10. 多模态数据输入流程

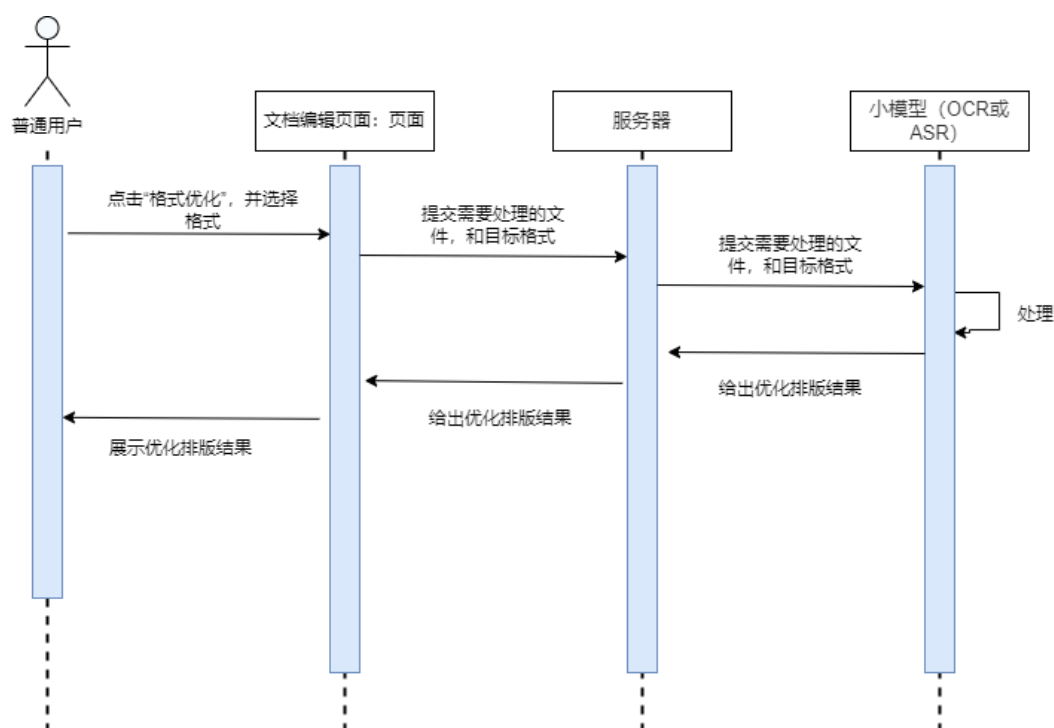


说明：

多模态数据输入流程由普通用户为业务主体，主要包括上传文件和 ORC 等小模型处理的结果。主要流程如下：

- 1) 点击“多模态数据输入”
- 2) 上传文件（图片、PDF、视频、音频等）
- 3) 点击“开始处理”
- 4) 显示处理后的结果

11. 智能格式排版流程

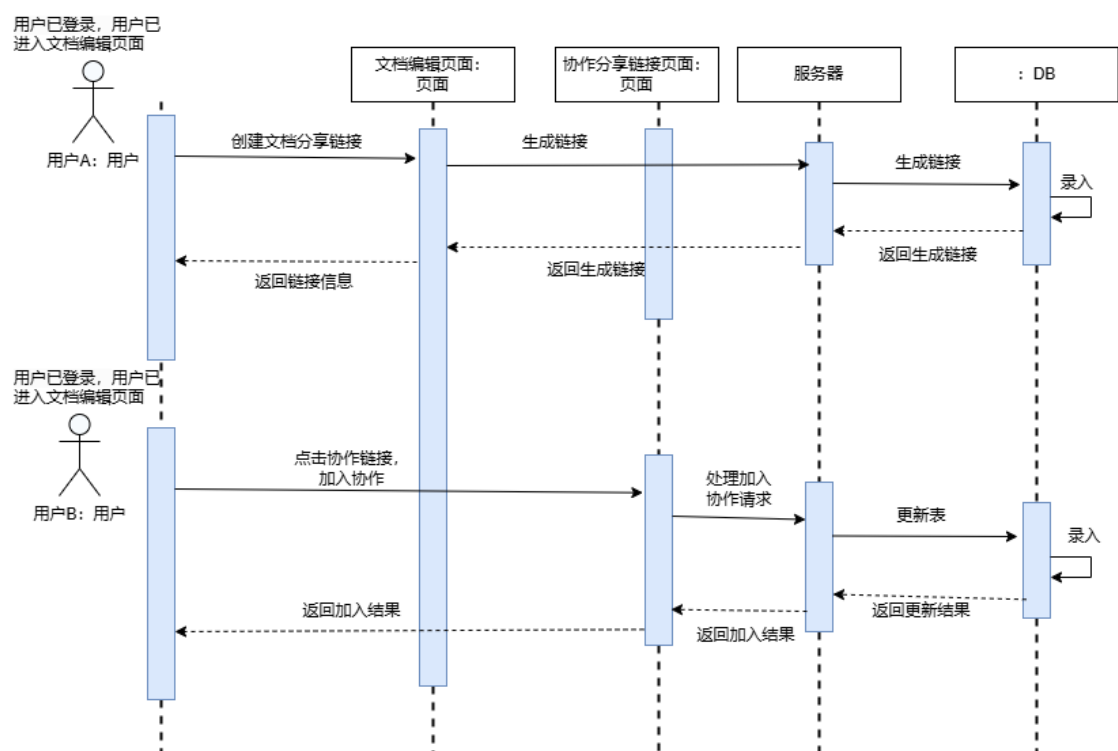


说明：

智能格式排版流程由普通用户为业务主体，主要包括选择格式和显示排版的结果。主要流程如下：

- 1) 点击“智能排版”
- 2) 选择排版格式类型
- 3) 点击“确认”
- 4) 显示排版后的文档

12. 文档共享流程

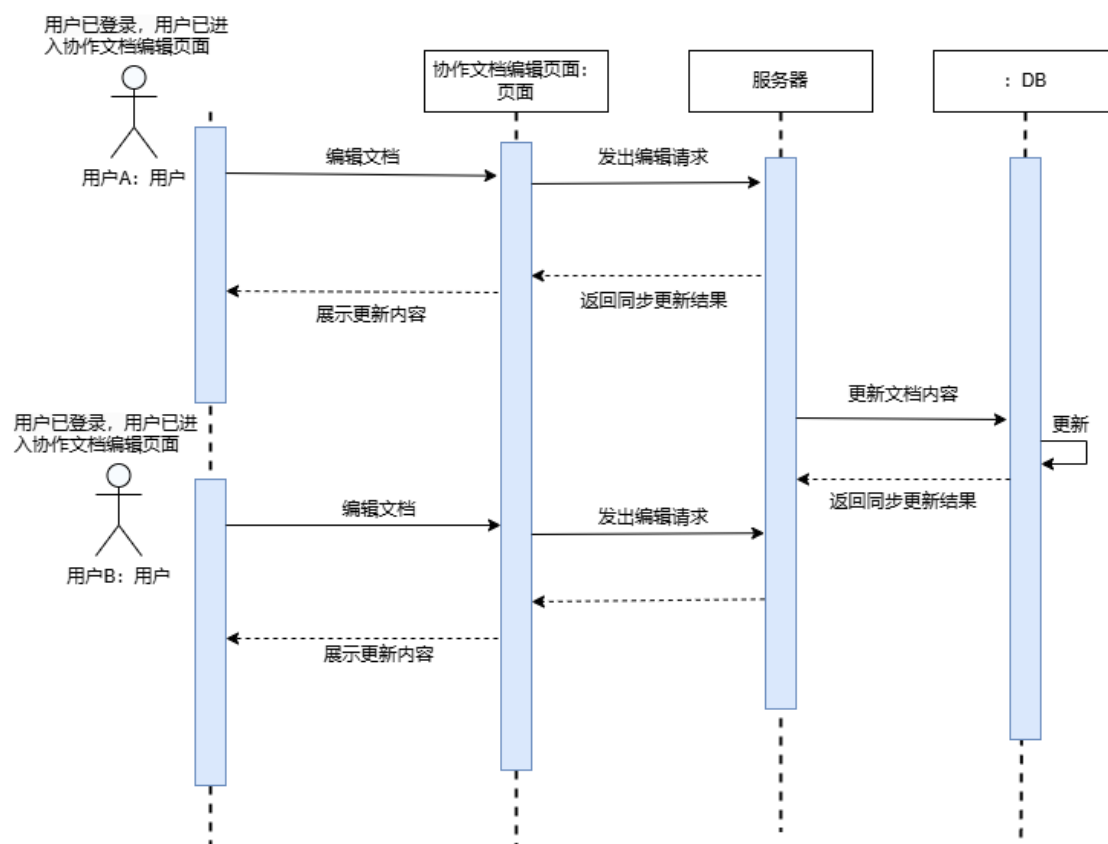


说明:

文档共享流程主要包括用户 A 共享协作文档，用户 B 加入协作团队，主要流程如下（以下流程默认用户已经登陆并进入文档编辑页面）：

- 1) 用户 A 生成共享文档链接并分享给用户 B
- 2) 用户 B 点击链接加入协作团队

13. 实时协作编辑流程

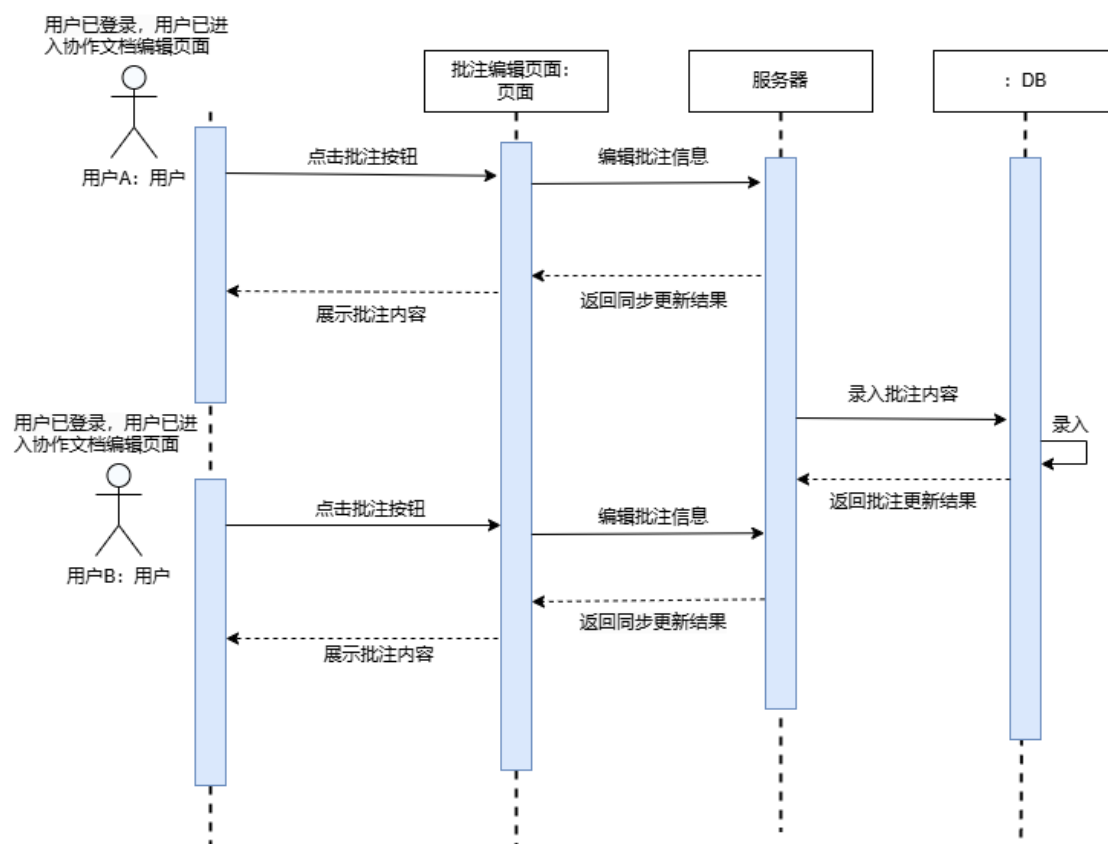


说明:

实时协作编辑流程主要包括用户 A 和用户 B 共同编辑文档，系统将同步编辑信息给每一个用户，主要流程如下（以下流程默认用户已经登陆并进入协作文档编辑页面）：

- 1) 初始化文档编辑页面
- 2) 用户 A 点击上方工具栏中的创建协作文档
- 3) 用户 A 发出生成协作链接请求
- 4) 服务器生成协作链接
- 5) 数据库更新协作信息后将链接返回给用户 A
- 6) 用户 A 将链接分享给用户 B
- 7) 用户 B 点击链接发出加入协作文档请求
- 8) 服务器处理加入请求
- 9) 数据库更新协作信息

14. 协作沟通流程



说明:

协作沟通流程主要包括用户为文档添加评论和批注，系统将评论和批注同步给每一个用户，主要流程如下（以下流程默认用户已经登陆并加入协作文档）：

- 1) 初始化协作文档编辑页面
- 2) 用户 A 选中文本行
- 3) 用户 A 点击文本行右侧批注按钮
- 4) 用户 A 在批注界面输入批注信息
- 5) 服务器记录用户 A 的批注信息并发送给数据库
- 6) 数据库更新用户 A 的批注信息
- 7) 服务器将用户 A 的批注信息同步给用户 B
- 8) 用户 B 页面上显示用户 A 的批注信息

三、业务功能概要结构



1. 个人空间



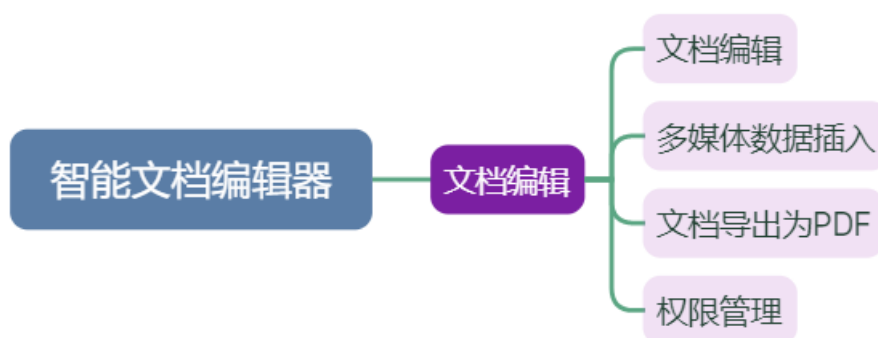
1.1 个人信息修改

- 1) 初始化个人信息页面
- 2) 点击个人信息旁修改按钮
- 3) 在相应位置修改个人信息
- 4) 点击确认保存修改

1.2 文档列表

- 1) 初始化文档列表
- 2) 点击创建文档或右键文档进行修改或删除
- 3) 双击文档跳转至文档编辑页面

2. 文档编辑



2.1 文档编辑

- 1) 点击文档链接
- 2) 初始化文档编辑页面
- 3) 在编辑区内键入文本
- 4) 使用工具栏工具调整编辑区文本格式
- 5) 保存文档内容

2.2 多媒体数据插入

- 1) 初始化文档编辑页面
- 2) 选中插入位置
- 3) 点击工具栏中的插入按钮
- 4) 找到需要插入的多媒体文件类型
- 5) 上传本地多媒体文件
- 6) 服务器解析文件数据并发送给数据库
- 7) 数据库存储文件数据
- 8) 服务器返回文件数据给用户
- 9) 系统将文件插入到文件选中位置



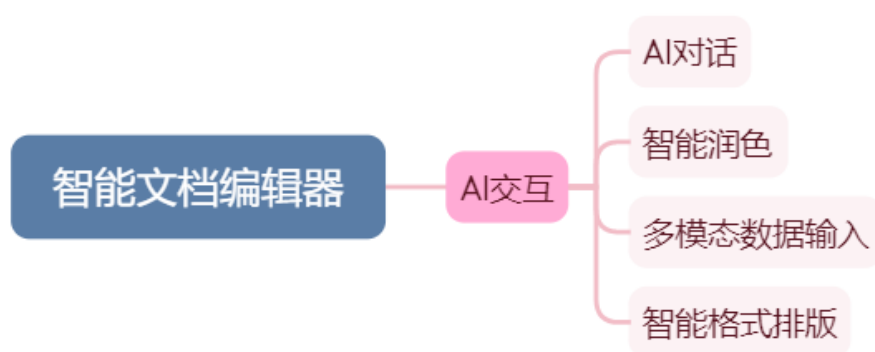
2.3 文档导出为 PDF

- 1) 初始化文档编辑页面
- 2) 用户点击工具栏中导出按钮
- 3) 设置导出的 PDF 格式
- 4) 服务器请求导出文档数据
- 5) 服务器将文档转换为 PDF 格式
- 6) 服务器将转换后的 PDF 文件发送给用户下载

2.4 权限管理

- 1) 初始化文档编辑页面
- 2) 用户点击工具栏中权限管理按钮
- 3) 修改文档编辑权限、
- 4) 服务器记录修改结果，并发送给数据库
- 5) 数据库存储修改结果
- 6) 显示修改权限后的文档

3. AI 交互



3.1 AI 对话

- 1) 初始化 AI 对话框



- 2) 用户输入问题并点击“发送”
- 3) 向服务器请求回答
- 4) 显示回答的结果

3.2 智能润色

- 1) 用户划句，选择需要优化的类型
- 2) 向服务器发出优化请求
- 3) 返回优化结果

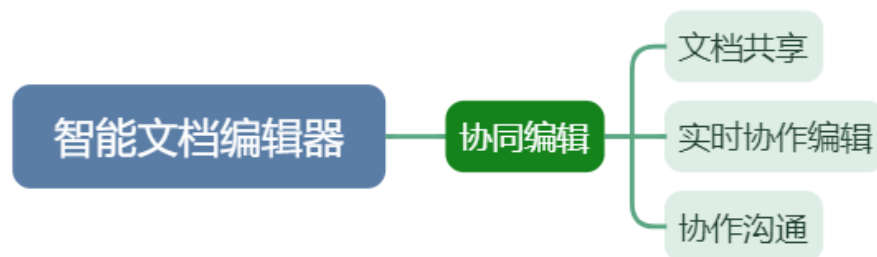
3.3 多模态数据输入

- 1) 初始化多模态数据上传界面
- 2) 上传文件
- 3) 向服务器发出处理请求
- 4) 显示处理结果

3.4 智能格式排版

- 1) 初始化选择格式界面
- 2) 选择需要的格式
- 3) 向服务器发出处理请求
- 4) 显示处理结果

4. 协同编辑





4.1 文档共享

- 1) 初始化文档编辑页面
- 2) 用户 A 点击上方工具栏中的创建协作文档
- 3) 用户 A 发出生成协作链接请求
- 4) 服务器生成协作链接
- 5) 数据库更新协作信息后将链接返回给用户 A
- 6) 用户 A 将链接分享给用户 B
- 7) 用户 B 点击链接发出加入协作文档请求
- 8) 服务器处理加入请求
- 9) 数据库更新协作信息

4.2 实时协作编辑

- 1) 初始化协作文档编辑页面
- 2) 用户 A 对文档进行编辑
- 3) 服务器将编辑信息记录并同步给用户 B
- 4) 用户 B 页面上展示用户 A 的编辑结果

4.3 协作沟通

- 1) 初始化协作文档编辑页面
- 2) 用户 A 选中文本行
- 3) 用户 A 点击文本行右侧批注按钮
- 4) 用户 A 在批注界面输入批注信息
- 5) 服务器记录用户 A 的批注信息并发送给数据库
- 6) 数据库更新用户 A 的批注信息
- 7) 服务器将用户 A 的批注信息同步给用户 B
- 8) 用户 B 页面上显示用户 A 的批注信息



四、模块定义

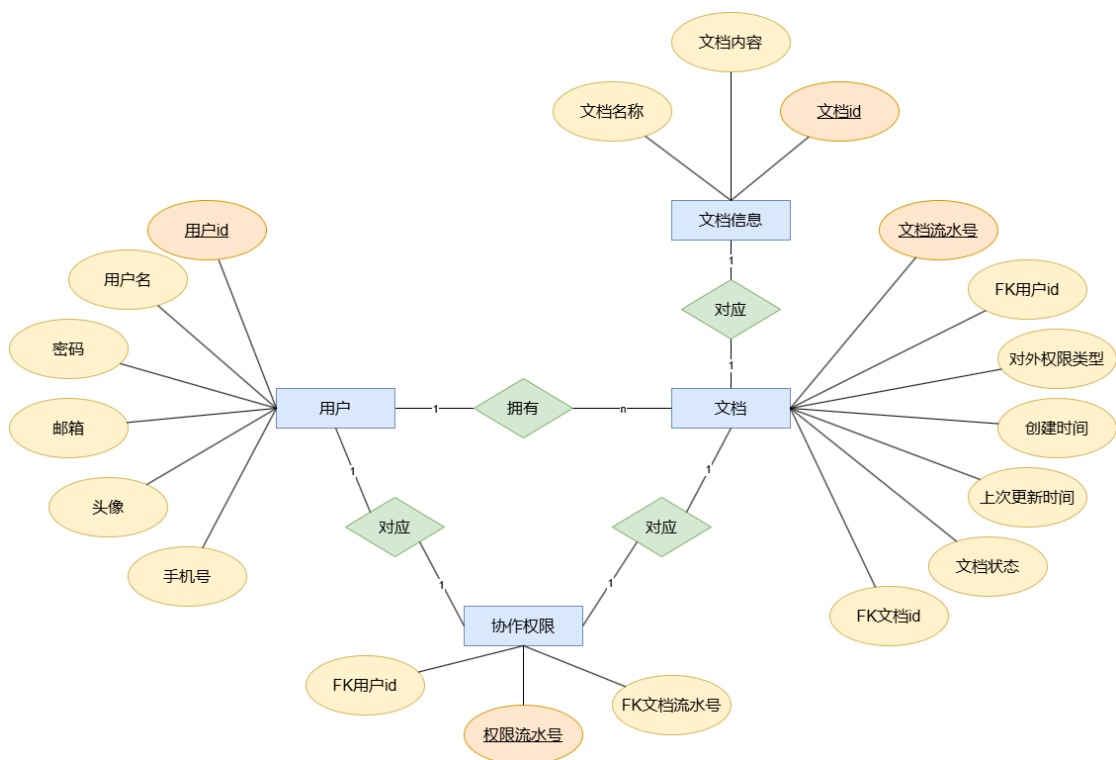
模块	组件	规格/内容
个人空间	个人信息修改	个人信息修改功能主要是用户点击个人信息栏，选择相应的个人信息进行添加或修改。
	文档列表	个人信息修改功能主要是用户点击个人信息栏，选择相应的个人信息进行添加或修改。
文档编辑	文档编辑	文档编辑主要是用户在文档编辑区键入文本来编辑文档，同时可以使用系统提供的工具对文档中的字体或格式进行修改。
	多媒体数据插入	多媒体数据插入主要是用户可以使用系统提供的插入功能，上传图片、音频或视频等多媒体文本数据到系统中，系统解析后插入到对应的文档位置中去。
	文档导出为 PDF	文档导出为 PDF 功能是通过系统可以将编辑好的文档导出为 PDF 文件，并下载到本地。
	权限管理	权限管理功能是对文档的编辑状态进行管理。
AI 交互	AI 对话	AI 对话是用户通过 AI 对话框直接向大模型提问。
	智能润色	智能润色是用户通过划词使用 AI 进行辅助优化文档。
	多模态数据输入	多模态数据输入是用户可上传图片、PDF、视频、音频等文件，调用小模型提取中文文件的文字。
	智能格式排版	智能格式排版是将文档排版为用户选择的格式。
协同编辑	文档共享	文档共享主要是用户可以将自己的文档共享出去，邀请其他用户协同编辑该份文档。
	实时协作编辑	实时协作编辑功能主要是不同用户在线实时编辑同一份文档，系统将实现同步更新编辑结果给所



		有参与协作的用户。
	协作沟通	协作沟通功能主要是不同用户，使用系统提供的评论和批注功能，在文档上留下批注和评论，系统将把批注和评论同步给所有参与协作的用户。

第五部分 E-R 实体设计

一、实体结构图



二、实体描述

1. 用户信息实体描述

编号	数据项	建议存储名	数据类型	精度与宽度约定
----	-----	-------	------	---------



1	用户 id(PK)	uId	整数	–
2	用户名	uName	字符串	20
3	邮箱	uMail	字符串	30
4	手机号	uPhone	字符串	20
5	用户密码	uPassword	字符串	20
6	用户头像	uPictures	字符串	100
7	最大文档空间	uMax	整数	–

2. 文档信息实体描述

编号	数据项	建议存储名	数据类型	精度与宽度约定
1	文档 id (PK)	dId	整数	–
2	文档名称	dTitle	字符串	50
3	文档内容	dContent	字符串	–

3. 文档列表信息实体描述

编号	数据项	建议存储名	数据类型	精度与宽度约定
1	文档流水号	fId	整数	–
2	用户 id	uId	整数	–
3	对外权限类型	fAuthority	字符串	10
4	创建时间	fCreateDate	日期	10
5	上次更新时间	fUpdateDate	日期	10
6	文档状态	fState	字符串	10
7	文档 id	dId	整数	–

4. 协作权限

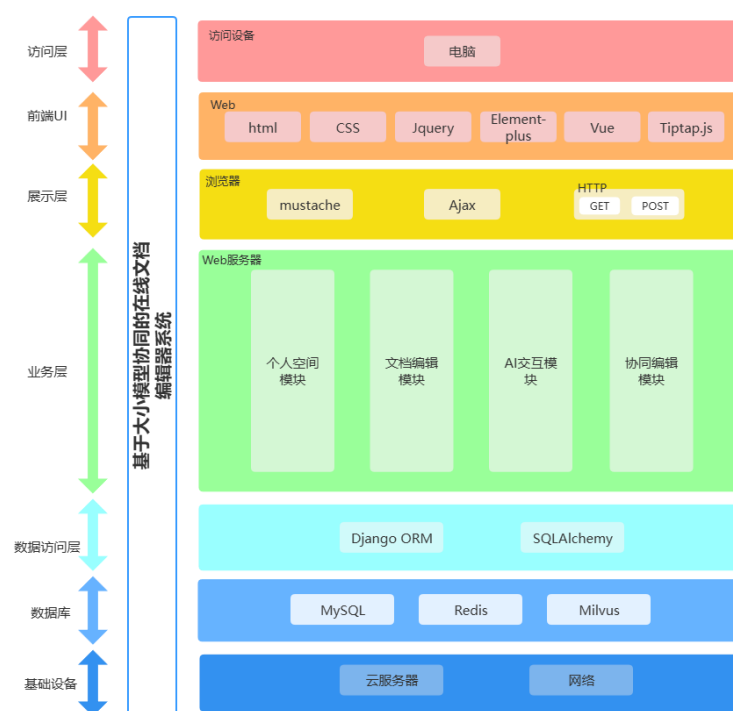
编号	数据项	建议存储名	数据类型	精度与宽度约定
1	文档流水号	fId	整数	–



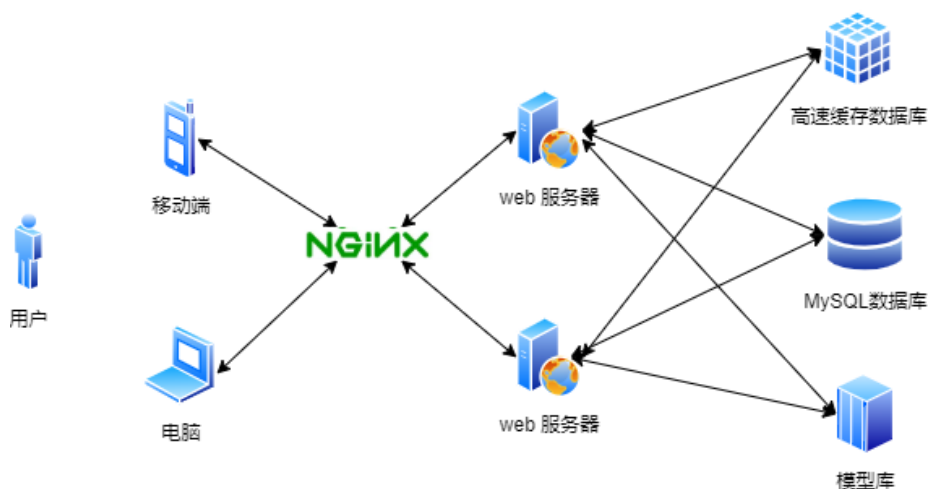
2	用户 id	uId	整数	-
3	权限流水号	aId	整数	-

第六部分 总体设计

一、系统逻辑结构设计



二、物理架构设计





1. 客户端

电脑端手机端都需要使用浏览器访问，建议使用 Chrome 浏览器。

2. 负载均衡服务器

系统使用负载均衡服务器实现 http 请求的分发请求。分配到不同的 web 服务器解决系统的并发问题。本系统采用软件的解决方式，即 nginx 负载均衡服务器。

3. web 服务器

系统采用 Django 配置的服务器处理客户端的 http 请求。并且返回 http 应答。

4. 高速缓存数据库

高速缓存数据库用于存储 session 数据，解决多个 web 服务器之间 session 数据不共享的问题。本系统采用 redis 作为高速缓存服务器。

5. 存储数据库

系统使用 mysql 数据库存储用户以及系统数据。Mysql 版本为 8.0 及以上。

6. 模型库

系统主要使用百度 AI studios 上的小模型与 ERNIE Bot。



三、技术架构设计

1. 星河社区独占机器服务部署

服务部署功能为开发者提供快速发布 API 的能力。支持开发者将 Notebook 项目中自定义编写的各类程序，包括全栈应用、大模型推理服务、一言插件后端等部署为服务 API，并支持公网调用，以及查看服务调用详情。

独占机器资源的服务部署适合部署需要 CPU 或 GPU 服务器资源的各类后端服务和应用。部署完成后会独占一台机器，按使用时长进行付费，支持部署为公开访问和限制访问的 API。

2. Nginx

Nginx (engine x) 是一个高性能的 HTTP 和反向代理 web 服务器，同时也提供了 IMAP/POP3/SMTP 服务。

Nginx 可以在大多数 Unix Linux OS 上编译运行，并有 Windows 移植版。一般情况下，对于新建站点，建议使用最新稳定版作为生产版本，已有站点的升级紧迫性不高。Nginx 的源代码使用 2-clause BSD-like license。

Nginx 作为负载均衡服务：Nginx 既可以在内部直接支持 Rails 和 PHP 程序对外进行服务，也可以支持作为 HTTP 代理服务对外进行服务。Nginx 采用 C 进行编写，不论是系统资源开销还是 CPU 使用效率都比 Perlbal 要好很多。

3. MySQL 存储数据库

MySQL 是一个关系型数据库管理系统，由瑞典 MySQL AB 公司开发，属于 Oracle 旗下产品。MySQL 是最流行的关系型数据库管理系统之一，在 WEB 应用方面，MySQL 是最好的 RDBMS (Relational Database Management System, 关系数据库管理系统) 应用软件之一。



MySQL 是一种关系型数据库管理系统，关系数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性。

4. Milvus

Milvus 是一款云原生向量数据库，它具备高可用、高性能、易拓展的特点，用于海量向量数据的实时召回。

Milvus 基于 FAISS、Annoy、HNSW 等向量搜索库构建，核心是解决稠密向量相似度检索的问题。在向量检索库的基础上，Milvus 支持数据分区分片、数据持久化、增量数据摄取、标量向量混合查询、time travel 等功能，同时大幅优化了向量检索的性能，可满足任何向量检索场景的应用需求。通常，建议用户使用 Kubernetes 部署 Milvus，以获得最佳可用性和弹性。

Milvus 采用共享存储架构，存储计算完全分离，计算节点支持横向扩展。从架构上来看，Milvus 遵循数据流和控制流分离，整体分为了四个层次，分别为接入层（access layer）、协调服务（coordinator service）、执行节点（worker node）和存储层（storage）。各个层次相互独立，独立扩展和容灾。

5. Redis 高速缓存数据库

Redis (Remote Dictionary Server)，即远程字典服务，是一个开源的使用 ANSI C 语言编写、支持网络、可基于内存亦可持久化的日志型、Key-Value 数据库，并提供多种语言的 API。

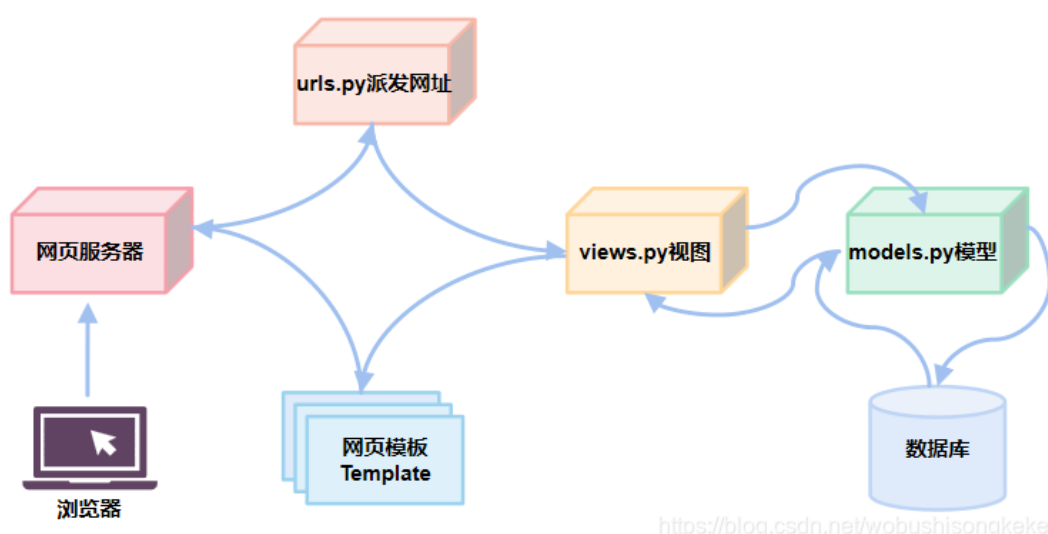
Redis 是一个高性能的 key-value 数据库。redis 的出现，很大程度补偿了 memcached 这类 key/value 存储的不足，在部分场合可以对关系数据库起到很好的补充作用。

Redis 支持主从同步。数据可以从主服务器向任意数量的从服务器上同步，从服务器可以是关联其他从服务器的主服务器。这使得 Redis 可执行单层树复制。存盘可以有意无意的对数据进行写操作。由于完全实现了发布/订阅机制，使得从数据库在任何地方同步树

时，可订阅一个频道并接收主服务器完整的消息发布记录。同步对读取操作的可扩展性和数据冗余很有帮助。

Redis 官方测试结果测试完成了 50 个并发执行 100000 个请求。设置和获取的值是一个 256 字节字符串。Linux box 是运行 Linux 2.6, 这是 X3320 Xeon 2.5 ghz。文本执行使用 loopback 接口 (127.0.0.1)。结果: 读的速度是 110000 次/s, 写的速度是 81000 次/s。

6. Django MTV 模式



M——Model——models.py

Model 是 Django 表示数据的模式，以 Python 的类为基础在 models.py 中设置数据项与数据格式，基本上每个类对应一个数据库中的数据表。在后端 Django 会自动把这个类中的设置对应到数据库系统中，不管使用的是哪一种在数据库。因此，定义每个数据项时，除了数据项名称外，也要定义此项目的格式以及这张表和其他表格相互之间的关系（即数据关联）。定义完毕后，网站的其他程序就可以使用 Python 语句来操作这些数据内容，不用关心实际使用的 SQL 指令以及使用的是哪一种数据库。在 models.py 中定义所有需要用的数据格式，一般是以数据库的形式来存储的，定义后的 Model 数据类要把它 import 到 views.py 中。

T——Template——template 文件夹

把取得的数据用美观且有弹性的方式输出，是在 Template 中处理。使用 templates 来做每个网页的外观框架，送至 template 中要被使用的数据尽量是可以直接显示的简单形



式，不要试图在 template 文件中使用复杂的方法处理这些送进来的变量，如果需要对变量进行更复杂的运算，那么这些工作应该放在 views.py 中完成。

V——View——views.py

把数据取出来，或是如何存进去等程序逻辑，则是在 View 中，也就是在 view.py 中处理。View 是 Django 最重要的程序逻辑所在的地方，网站大部分程序设计都放在这里。这里放了许多要操作的数据，以及安排哪些数据需要被显示出来的函数，在函数中把这些数据传送给网页服务器或交由 Template 的渲染器后再送到网页服务器中。这些放在 views.py 中的函数，再由 urls.py 中的设计进行对应和派发。

第七部分 用户界面设计

一、桌面布局设计

1. 注册界面

LOGO

A user registration form mockup with a light blue background. At the top, there is a placeholder for a 'LOGO'. Below it, the text '编辑器名字' (Editor Name) is displayed, followed by a subtitle '一款基于大小模型协同的智能文档编辑器' (A smart document editor based on the synergy of large and small models). The form contains two input fields: '账号' (Account) and '密码' (Password). To the right of the password field is a link '重置密码' (Reset Password). Below the input fields is a '登录' (Login) button. At the bottom of the form, there is a link '没有密码? 快速注册' (No password? Register quickly).



2. 登陆界面

LOGO

注册

手机号

密码

确认密码

短信验证码

注册

返回登录界面

3. 用户个人信息界面

用户名 头像			
创建文档	最近文档		
最近文件	选择文件 未选择文件		
共享给我	文件名	创建者	打开时间
我的空间	test文档	用户A	1分钟前
回收站	test文档	用户A	1分钟前
	test文档	用户A	1分钟前
	test文档	用户A	1分钟前
	test文档	用户A	1分钟前
	test文档	用户A	1分钟前
	test文档	用户A	1分钟前
	test文档	用户A	1分钟前
	test文档	用户A	1分钟前
	test文档	用户A	1分钟前
我的容量	test文档	用户A	1分钟前
100MB/1G			



4. 文档空间界面

[返回个人空间](#)

头像

用户昵称

个人信息

头像

头像

编辑

昵称

用户昵称

编辑

账号

12345678

手机号

sxjy*****

更改

密码

188****8888

重置

5. 文档编辑界面

[返回个人空间](#) | [文档题目](#) 用户名 头像

🔍 ↺ B U I 标题 字体 尺寸 插入

文档大纲

一级标题

二级标题

一级标题

二级标题

文档编辑内容

AI对话区

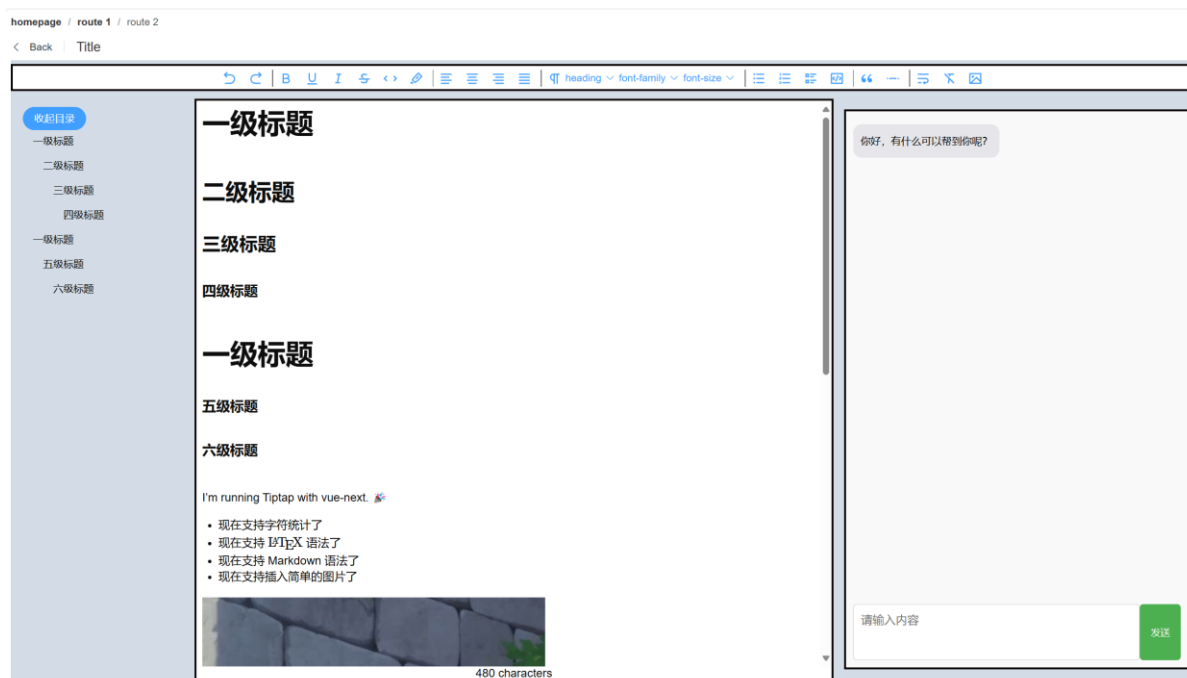
你好。我是文心—眼大模型，有什么可以帮你的

我有xxxxx问题，你可以帮我解决一下吗

请输入内容 发送



二、业务界面风格展示



第八部分 运行环境和部署

一、运行环境

1、客户机环境：

- (1) Intel i5 或以上微处理器（CPU）；
- (2) Microsoft Windows 10 操作系统；
- (3) 4GB 以上内存，建议使用 8GB 内存；
- (4) 较高版本的 Microsoft Edge、google Chrome 内核浏览器或其他浏览器；

2、服务器：

- (1) 星河社区独占机器服务



(2) Python 3.8

(3) MySQL 8.*

3、开发环境：

项目	名称	版本
开发平台	Python	3.8
开发工具	PyCharm	PyCharm 2023.2.1
页面开发工具	Vue	3.x
代码管理工具	Git	-
测试工具	Postman	8.8.0

二、系统性能要求

#	项目	模块	级别	技术参数
1	设计实现 技术指标	系统架构	A	平台用户采用 B/S 模式
2		面向对象开发语言与框架	A	采用 Python 开发语言和 Django 等框架
3		注释和文档	A	符合 CMMI 软件开发过程标准文档（至少提供：需求，概要、详细设计，测试报告，部署和环境，用户手册），代码注释量≥30%
4		模块化和适合实训	A	SOA 设计、模块化，保证系统各模块单元较强的独立性，适合实训教学
5		测试覆盖率	A	功能覆盖率≥100%，业务覆盖率≥100%，语言覆盖率≥100%，逻辑覆盖率≥80%
6	资源利用 率要求	CPU 占用率	B	≤50%利用率（附近标准配置）
7		内存使用率	B	≤75%利用率（附近标准配置）
8	响应时间 要求	服务器	B	≤100ms（附近标准配置）
9		网络	B	≤100ms（附近标准配置）
10		客户端	B	≤5ms（附近标准配置）
11	系统稳定	成熟性	A	真实的用户成功使用本系统
12	性要求	稳定性	A	无故障运行时间≥365 天，系统恢复时间≤2 小时



13		先进性	B	采用目前体验最好的 Windows7 及以上版本系统
14		典型意义	B	项目具有典型意义，有推广价值
15	Web 服务 接口要求	WebService 服务器端	A	支持协议定制，支持安全过滤，支持消息队列，多种模式重发，支持成功检测，消息延迟 $\leq 2s$ ，丢包 $\leq 0.001\%$
16		WebService 移动终端	A	支持协议定制，支持安全过滤，支持消息队列，多种模式重发，支持成功检测，消息延迟 $\leq 2s$ ，丢包 $\leq 0.001\%$
18	集成部署 环境	服务器	A	Linux 操作系统
19		数据库	A	Mysql8.0 及以上数据库系统
21		客户端	A	Windows7 及以上