

项目名称		密级
文脉医典——医学文献智能识别与检索系统		仅供收件方查阅
项目编号	版本	文档编号
7	1.0	Project ID_SD_001

文脉医典——医学文献智能识别与检索系统

概要设计说明书

拟制	翟芸坤 饶艺 谭雪瑶 昌宇	日期	12.30
评审人	王文鑫	日期	12.30
批准	王文鑫	日期	12.30



武汉市软酷网络科技有限公司

版权所有 不得复制

Revision Record

修订记录

Date 日期	Revision Version 修订版本	CR ID /Defect ID CR/ Defect号	Sec No. 修改章节	Change Description 修改描述	Author 作者

目 录

1 简介.....	6
1.1 目的.....	6
1.2 范围.....	6
1.2.1 软件名称.....	6
1.2.2 软件应用.....	6
2 概要设计.....	7
2.1 第0层设计描述.....	7
2.1.2 设计思路.....	7
2.2 第1层设计描述.....	8
2.2.1 系统结构.....	8
2.2.1.1 系统结构描述.....	8
2.2.1.2 业务流程说明.....	9
2.2.2 分解描述.....	11
2.2.2.1 用户中心.....	11
2.2.2.1.1 注册.....	11
2.2.2.1.2 登录.....	13
2.2.2.1.3 绑定用户邮箱.....	14
2.2.2.2 智能文献检索.....	17
2.2.2.2.1 关键词、短语和自然语言搜索.....	17
2.2.2.3 个性化推荐.....	19
2.2.2.3.1 用户行为分析.....	19
2.2.2.3.2 推荐文献.....	23
2.2.2.4 后台文献管理模块.....	27
2.2.2.4.1 文件上传.....	27
2.2.2.4.2 文件删除.....	29
2.2.2.4.3 文件查询.....	30
2.2.2.4.4 文件编辑.....	32
2.2.2.5 文献管理和组织.....	33
2.2.2.5.1 书签功能.....	34
2.2.2.5.2 标注功能.....	38
2.2.2.5.3 笔记功能.....	42
2.2.2.6 知识图谱.....	47
2.2.2.6.1 知识集成与构建.....	47
2.2.2.6.2 可视化展示.....	51
2.2.2.7 文献展示与阅读模块.....	53
2.2.2.7.1 显示文献基本信息功能.....	54
2.2.2.7.2 显示文献引用次数和阅读量功能.....	56
2.2.2.7.3 PDF/HTML 格式的文献展示功能.....	58
2.2.2.7.4 PDF/HTML 格式的文献下载功能.....	60
2.2.2.8 后台运行 - 用户管理模块.....	62
2.2.2.8.1 创建 / 删除用户账户功能.....	62
2.2.2.8.2 更新用户资料功能.....	64

2.2.2.8.3 用户权限管理功能.....	66
2.2.3 接口描述.....	68
2.2.3.1 注册接口.....	68
2.2.3.2 登录接口.....	68
2.2.3.3 密码找回.....	69
2.2.3.4 个人信息修改.....	69
2.2.3.5 关键词、短语和自然语言搜索.....	69
2.2.3.6 基于时间、作者、引用等条件对搜索结果进行排序.....	70
2.2.3.7 搜索结果过滤接口，支持按日期、类型等条件过滤.....	70
2.2.3.8 个性化推荐接口.....	70
2.2.3.9 文献书签功能接口.....	70
2.2.3.10 文献标注功能接口.....	71
2.2.3.11 文献笔记功能接口.....	71
2.2.3.12 获取文献的知识图谱数据接口.....	71
2.2.3.13 获取文献的基本信息接口.....	72
2.2.3.14 获取文献的展示格式接口.....	72
2.2.3.15 文献全文下载接口.....	72
2.2.3.16 文献全文打印接口.....	72
2.2.3.17 创建新用户功能接口.....	73
2.2.3.18 删除用户账户功能接口.....	73
2.2.3.19 更新用户资料功能接口.....	73
2.2.3.20 获取用户日活跃报表接口.....	74
2.2.3.21 功能页面分析报告接口.....	74
2.2.3.22 用户点击率分析接口.....	74
2.2.3.23 数据爬取功能接口.....	75
3 数据结构/数据库设计.....	75
3.1 概念模型.....	75
3.2 数据库表设计.....	76
3.2.1 用户表.....	76
3.2.2 文献表.....	76
3.2.3 文献OCR识别结果表.....	77
3.2.4 检索日志表.....	77
3.2.5 数据分析表.....	77
4 界面设计.....	78
4.1 首页.....	78
4.2 登录.....	78
4.2 注册.....	79
4.3 pdf格式文献上传.....	79
4.4 智能文献检索.....	80
4.5 知识图谱.....	80
5 Error Design 出错处理设计.....	80

Keywords 关键词:

智能检索 医学文献智能识别与检索系统

Abstract 摘要:

医学智能识别和检索系统。本系统集成先进的医学图像处理技术和深度学习算法，旨在实现对医学影像、病历文档及医学文献的高效智能识别与精准检索。系统支持多种数据格式的输入，包括但不限于DICOM影像、PDF病历及各类医学文献，通过自动化解析与特征提取技术，快速定位并分析关键医学信息。此外，系统构建了一个全面的医学知识图谱，利用语义理解与关联分析，促进跨领域医学知识的融合与推理，助力临床决策与科研探索。

本系统还具备个性化推荐功能，基于用户的查询历史与偏好，运用智能推荐算法，推送相关病例、研究论文及最新医学进展，促进知识的持续更新与学习。为保障数据安全与隐私，系统实施了严格的数据加密与访问控制机制，确保医学数据在传输与存储过程中的安全性。

本文档将详细阐述项目的总体架构、具体功能需求、系统性能标准、API接口规范、设计约束条件、软件质量保障措施以及需求优先级划分，旨在为系统开发团队、分析师及测试人员提供全面深入的理解，确保系统能够准确、高效地满足医学领域对智能识别与检索的迫切需求。

List of abbreviations 缩略语清单:

Abbreviations 缩略语	Full spelling 英文全名	Chinese explanation 中文解释

1 简介

1.1 目的

本系统设计文档是为设计人员和开发人员而编写的，它说明了本系统的各项功能和性能需求，明确标识各功能的实现过程，阐述使用范围及背景，提供客户解决问题或达到目标所需的条件或权能，提供一个度量和遵循的标准。

本说明书的预期读者为中间用户（指软件的管理人员、设计人员、开发人员、测试人员、维护人员）。

1.2 范围

1.2.1 软件名称

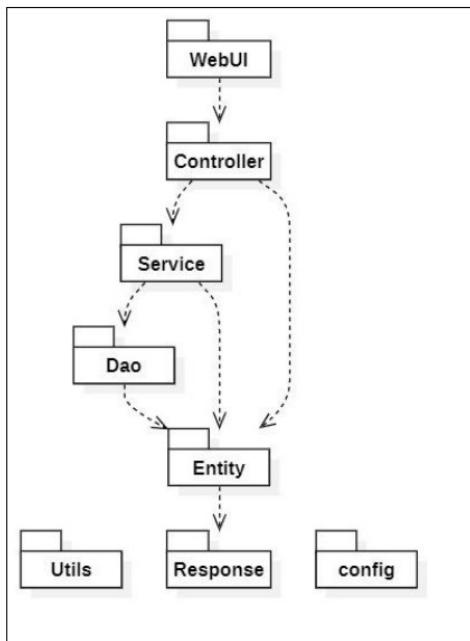
文脉医典-医学智能文献识别和检索系统

1.2.2 软件应用

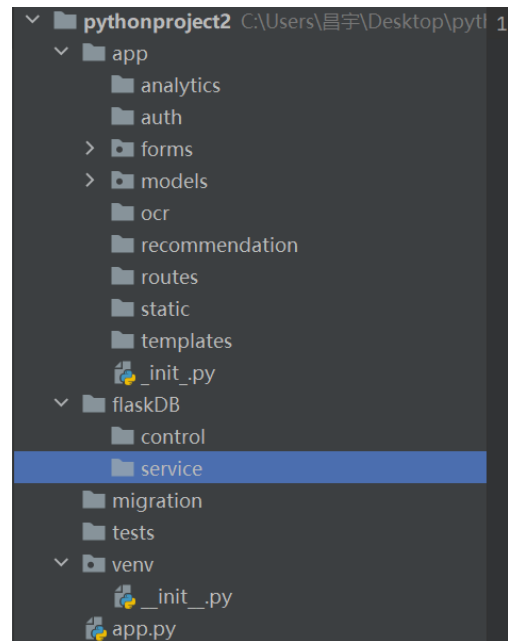
文脉医典是一款集医学文献检索、智能推荐、深度阅读、互动讨论与知识分享于一体的综合性在线平台。它不仅为医学专业人士（如医生、护士、科研人员、医学生等）提供了便捷、高效的文献获取途径，同时也向广大健康关注者开放，满足他们对医学知识的探索与学习需求。

2 概要设计

2.1 第0层设计描述



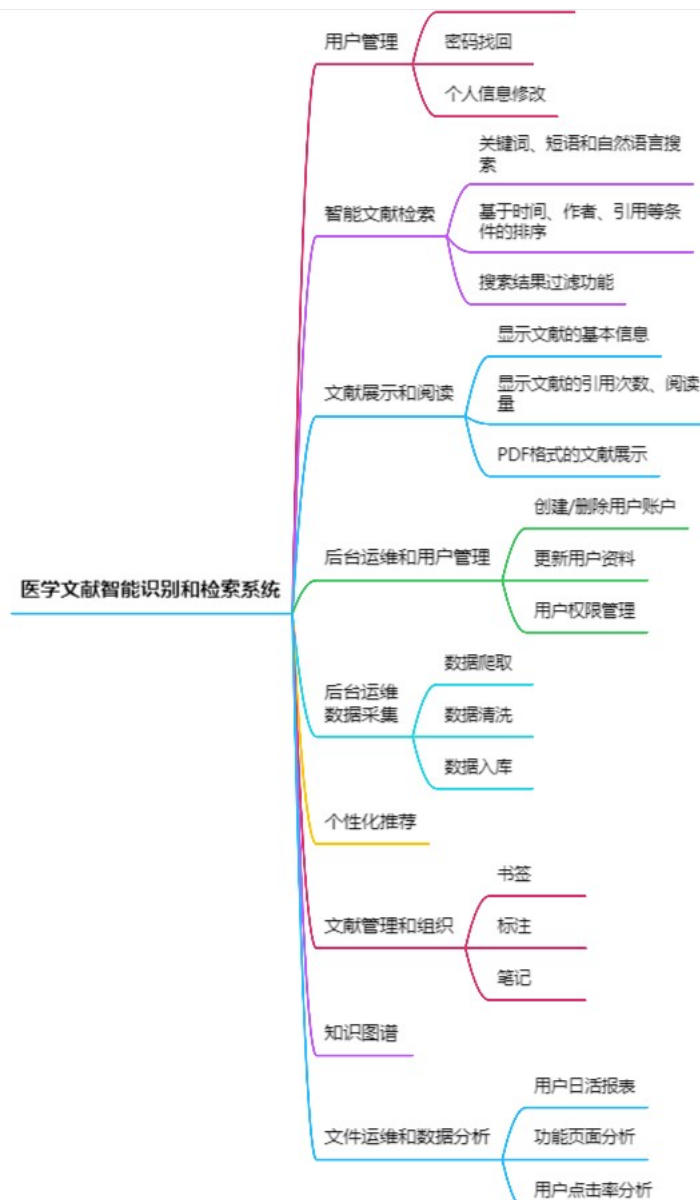
2.1.2 设计思路



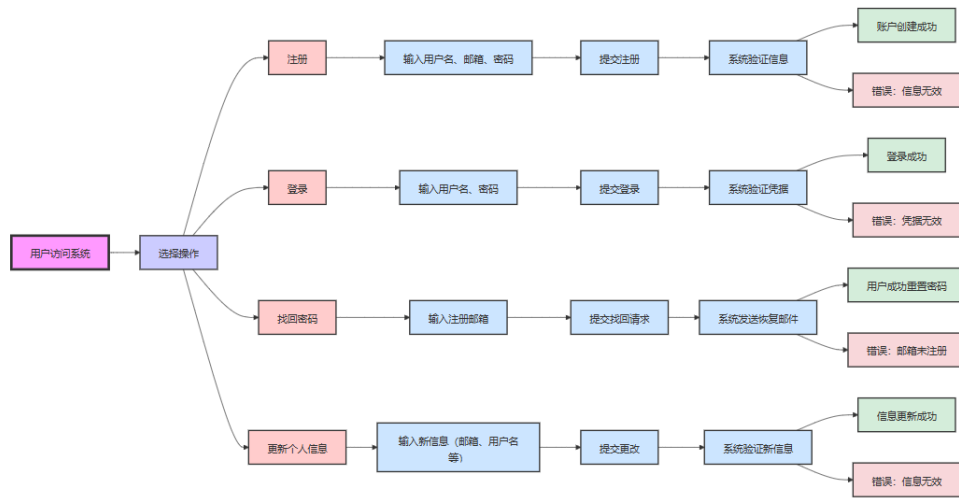
2.2 第1层设计描述

2.2.1 系统结构

2.2.1.1 系统结构描述

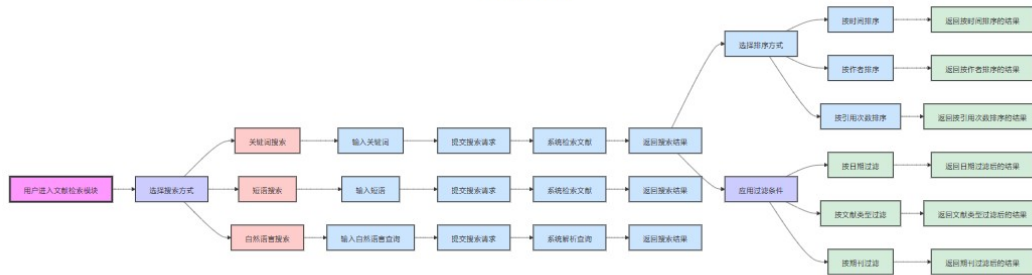


2.2.1.2 业务流程说明

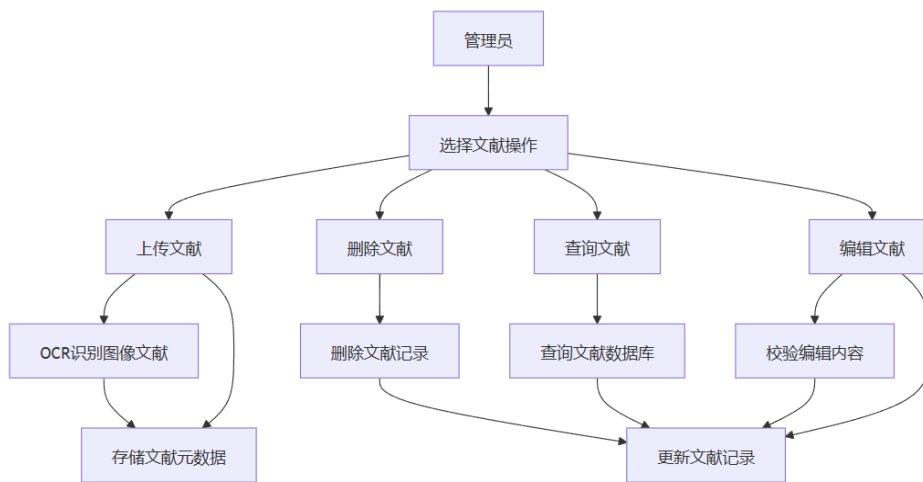


用户登录模块流程图

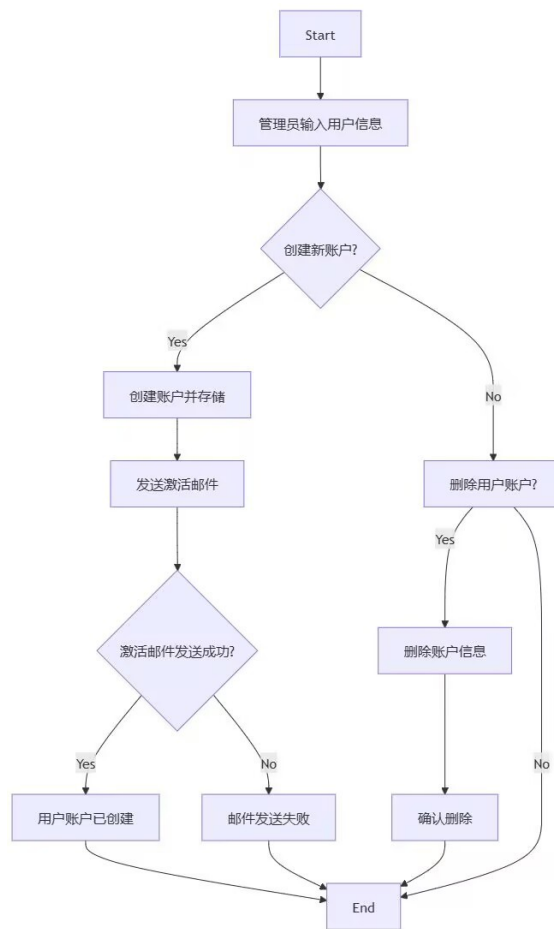
智能文献检索模块流程图



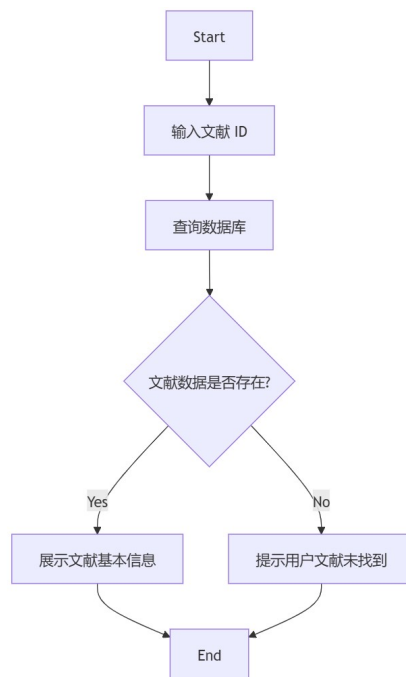
智能文献检索模块流程图



文献管理模块流程图



用户管理模块流程图



后台查询文献流程图

2.2.2 分解描述

2.2.2.1 用户中心

1 简介

用户基本功能模块包含用户注册、用户登录、用户忘记密码找回、用户修改密码、用户邮箱信息绑定等基本功能。用户可以进行注册账号，登录已经注册的账号，在登录后可以绑定用户邮箱，修改密码。如果忘记加密码，可以使用绑定的邮箱对密码进行找回。

2 功能列表

功能	描述
注册	用户在本平台没有账号需要通过注册账号才能使用平台其他功能
登录	用户已注册账号，通过输入用户名和密码登录平台
绑定用户邮箱	用户登录平台后，可以绑定本人邮箱便于密码找回

2.2.2.1.1 注册

1 功能设计描述

用户在本平台没有账号需要通过注册账号才能使用平台其他功能。

2 类

1) User 类:

作用：使用 SQLAlchemy 定义用户的数据模型，包含 id、username 和 password 三个属性。id 是主键，username 是唯一且不为空的，password 也不为空。该类映射到数据库中的 users 表，用于存储用户的信息。

2) UserDao 类:

作用：包含数据访问操作，具体为 add_user 静态方法，用于创建新的 User 对象并将其添加到数据库会话中，最后提交会话，完成将用户信息存储到数据库的操作。

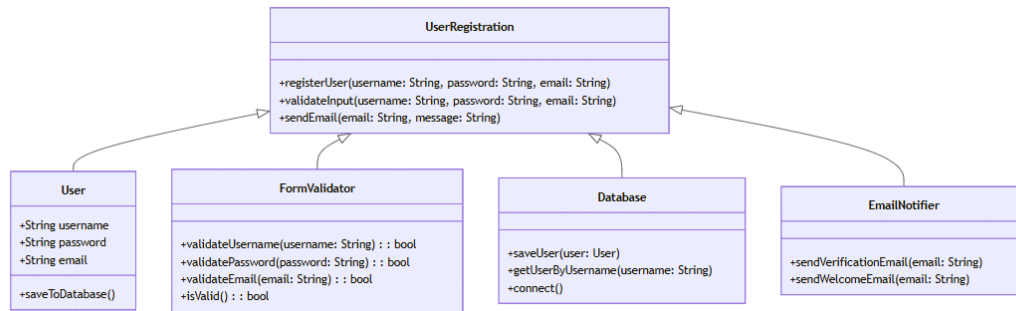
3) UserService 类:

作用：包含业务逻辑操作，register_user 静态方法会调用 UserDao 的 add_user 方法进行用户

注册操作，并处理可能的异常，实现用户注册的业务逻辑处理。

4) UserController 类:

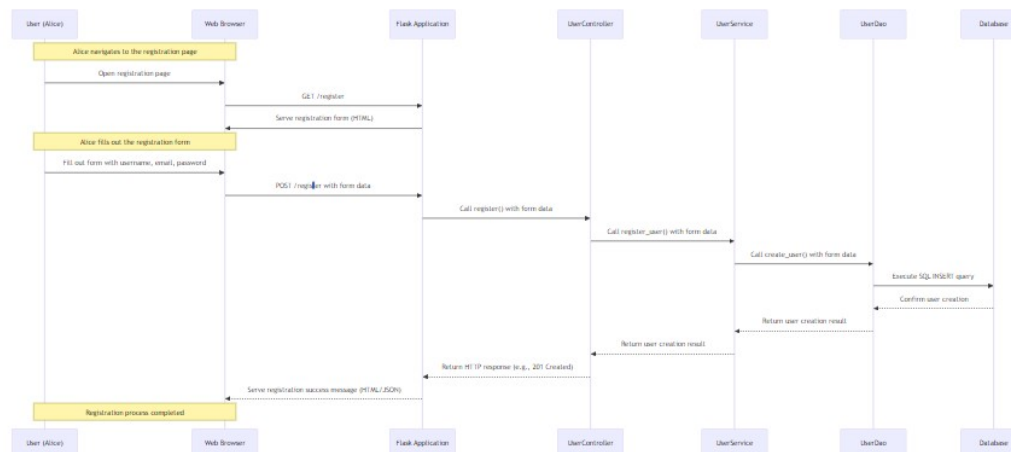
作用：使用 @app.route 装饰器将 /register 路径映射到 register 方法，处理 POST 请求。从请求中获取 username 和 password 信息，调用 UserService 的 register_user 方法进行注册操作，并根据结果返回不同的 JSON 响应和 HTTP 状态码，实现 HTTP 请求的处理和响应的生成。



3 文件列表

名称	类型	存放位置	说明
register.html	Html	Wenmai-web/templates	用户注册页面模板
__init__.py	py	Wenmai-web/views	视图模块的初始化文件
Auth.py	py	Wenmai-web/views	认证相关视图，包括注册

4 时序图（顺序图）



2.2.2.1.2 登录

1 功能设计描述

用户在本平台号需要通过登录账号才能使用平台其他功能。

2 类

1) User 类

功能：定义用户对象的属性和基本方法。包括用户ID、用户名、电子邮件和密码哈希等属性。提供设置密码（使用哈希算法）和验证密码uesrmapper集成进入的方法。

2) UserDao 类

功能：包含用户对象的数据访问操作。提供与用户数据相关的CRUD（创建、读取、更新、删除）方法，这些方法内部会调用 User 类（或 UserMapper，如果单独存在的话）的方法来执行具体的数据库操作。

3) UserService 接口

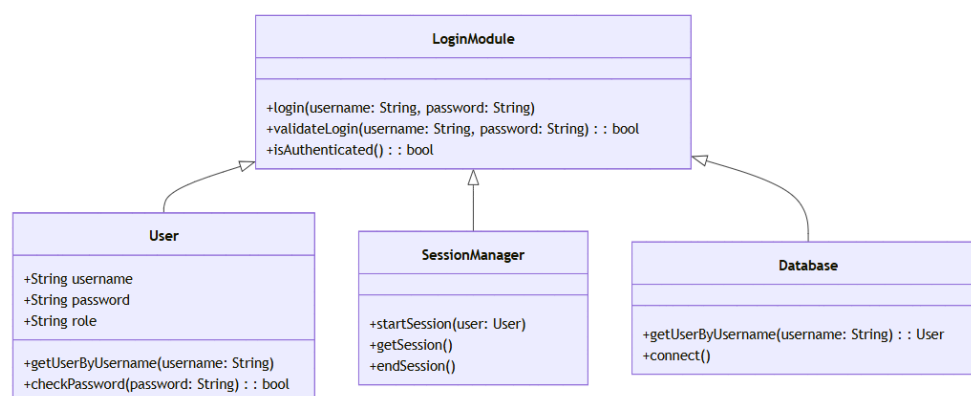
功能：定义用户数据具体操作的接口。这是一个抽象类或接口，定义了用户服务层应该提供的方法，如用户登录、注册、获取用户信息等。

4) UserServiceImpl 类

功能：实现 UserService 接口的具体类。这个类会包含具体的业务逻辑，如验证用户登录信息、注册新用户等。它会调用 UserDao 类的方法来执行数据访问操作。

5) UserController 类

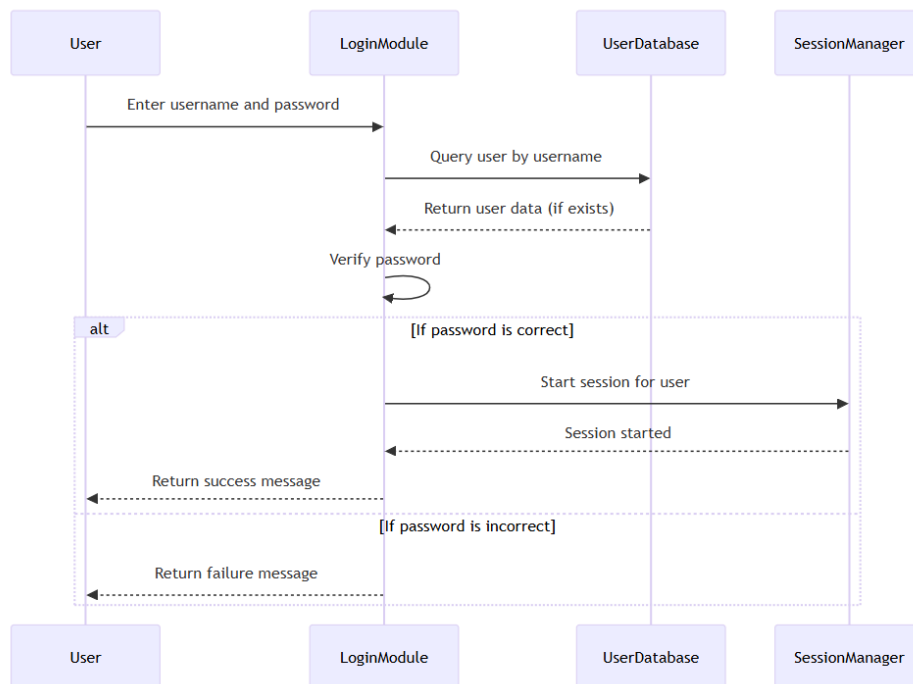
功能：包含用户对象的所有操作，即处理与用户相关的HTTP请求。在 Flask 中，这通常通过视图函数或视图类（如使用 Flask-Classy 或 Flask-RESTful）来实现。UserController 会调用 UserService 类的方法来处理具体的业务逻辑，并返回相应的HTTP响应。



3 文件列表

名称	类型	存放位置	说明
Login.html	html	Wenmai-web/ templates	用户登录页面模板
__init__.py	py	Wenmai-web/ views	视图模块的初始化文件
Auth.py	py	Wenmai-web/ views	认证相关视图，包括登录

4 时序图（顺序图）



2.2.2.1.3 绑定用户邮箱

1 功能设计描述

用户通过手机号注册后，完善个人信息绑定邮箱。

2 类

1) UserController

处理用户请求，绑定用户邮箱。调用 bindEmail() 方法时，会验证邮箱格式，并向用户发送邮箱验证邮件。

2) UserDao

负责数据库操作，提供获取用户信息和保存邮箱地址的方法（如 getUserById() 和 saveEmail()）。

3) User

定义用户对象的基本属性，例如 userId、email 和 isEmailVerified，并提供邮箱绑定和邮箱验证的方法（如 bindEmail() 和 verifyEmailCode()）。

4) UserService

提供用户业务逻辑接口，定义了绑定邮箱、验证邮箱格式、生成验证码等操作的接口。

5) UserServiceImpl

实现 UserService 接口，具体实现绑定邮箱、验证邮箱格式、生成验证码、验证验证码等操作。

6) UserMapper

负责与数据库交互，查询和更新用户信息，特别是更新用户的邮箱地址。

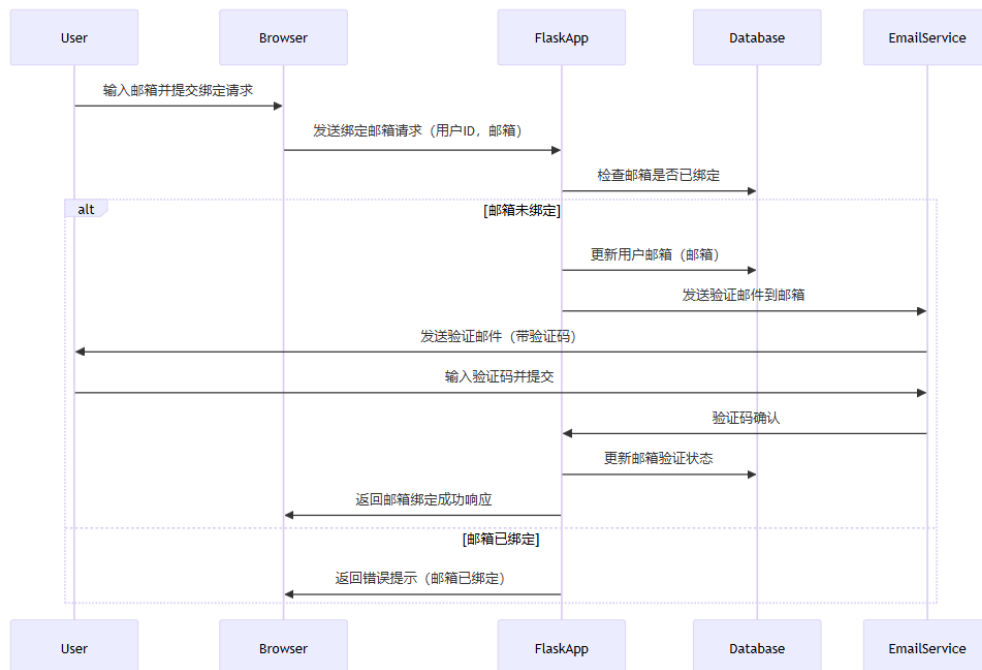
3 文件列表

名称	类型	存放位置	说明
addDoc.jsp	JSP	WebRoot/jsp/ document/issue/ addDoc.jsp	添加公文的面
bind_email.html	Html	Wenmai-web/ templates	用户绑定邮箱页面模板， 用户输入邮箱地址并提交
init.py	py	Wenmai-web/views	视图模块的初始化文件， 负责初始化与邮箱绑定相 关的视图模块

BindEmail.py	py	Wenmai-web/views	处理与邮箱绑定相关的视图逻辑，包括展示绑定邮箱页面、处理用户提交的邮箱、发送验证邮件等
bind_email_service.py	py	Wenmai-web/services	提供邮箱绑定的业务逻辑，包括验证邮箱格式、生成和验证验证码、保存邮箱等
bind_email_verification.py	py	Wenmai-web/services	处理发送验证码和验证验证码的业务逻辑
email_mapper.py	py	Wenmai-web/mappers	与数据库交互，更新用户的邮箱地址，并保存用户验证状态等
send_email.py	py	Wenmai-web/utils	用于发送验证邮件（包括生成邮件内容、调用邮件发

			送服务等)
email_validation.py	py	Wenmai-web/utills	用于验证邮箱格式以及验证码的有效性

4 时序图 (顺序图)



2.2.2.2 智能文献检索

1、简介

智能文献检索模块包含文献搜索、高级检索、检索结果过滤、文献详情查看、个性化推荐及用户收藏夹等基本功能。用户可以通过简单的关键词输入进行广泛的文献搜索，或者利用高级检索功能，通过设定特定的筛选条件（如作者、发表年份、期刊名称、关键词组合等）来精确查找目标文献。检索完成后，系统提供丰富的结果过滤选项，帮助用户快速定位到最相关的文献。

用户可点击任意检索结果进入文献详情页面，查看文献的摘要、全文（如果可用）、引用信息、作者介绍等详细信息。此外，智能文献检索模块还具备个性化推荐功能，根据用户的检索历史和兴趣偏好，智能推送可能感兴趣的文献，提升用户的检索效率。

为了方便用户管理自己感兴趣的文献，系统还提供了用户收藏夹功能。用户可以将重要的文献添加到收藏夹中，随时随地进行查看和管理。这些功能共同构成了一个高效、便捷的智能文献检索系统，助力用户快速获取所需的学术资源。

2、功能列表

功能	描述
关键词、短语和自然语言搜索	基于自然语言的搜索
基于时间、作者、引用等条件的排序	返回基于匹配度高低的排序

2.2.2.2.1 关键词、短语和自然语言搜索

1 功能设计描述

在输入框中，输入自然语言进行模糊查询。

2 类

1) SearchController

功能: 该类处理与用户请求相关的控制逻辑，接收来自用户的搜索请求，并调用相应的服务层进行处理

2) SearchDao

功能: 该类负责与数据库交互，从数据库中检索用户的搜索请求结果，提供原始数据访问接口

3.Document

功能: 定义文献对象的属性, 例如标题、摘要和全文等

4. SearchService

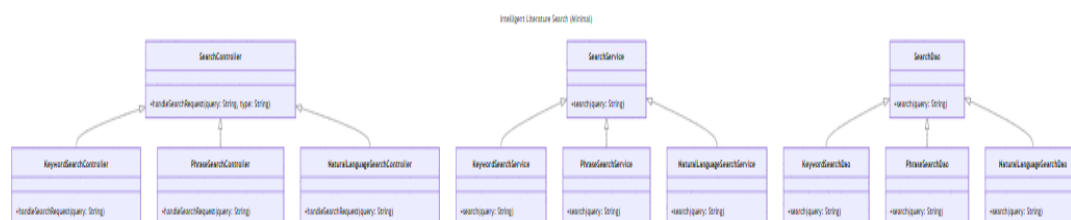
功能: 该类处理具体的搜索操作逻辑, 封装了业务层的搜索请求

5. SearchServiceImpl

功能: 该类实现了 SearchService 接口, 执行具体的搜索操作。它处理与不同搜索类型 (关键词、短语、自然语言) 相关的具体实现

6. SearchMapper

功能: 该类主要用于将搜索请求的参数映射到数据库或其他服务接口。它可以用于执行具体的查询操作, 连接到后端服务或数据库

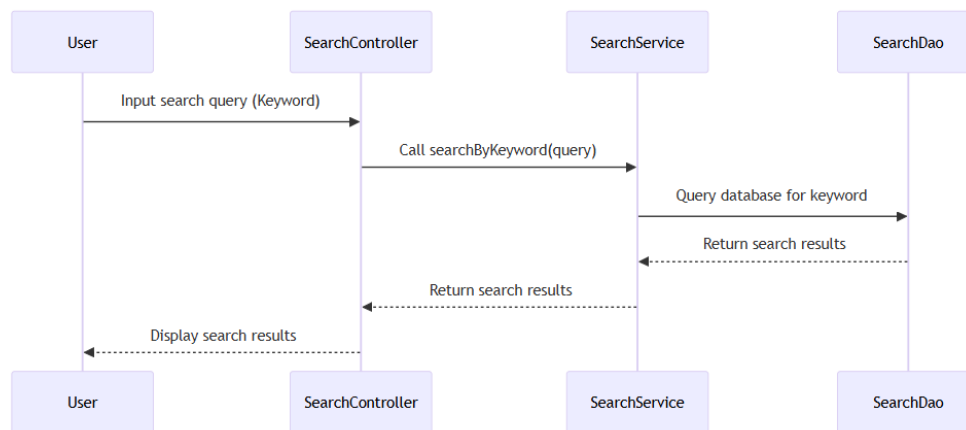


3 文件列表

名称	类型	存放位置	说明
natural_language_search.html	html	Wenmai-web/templates	自然语言搜索页面模板, 用户输入自然语言查询并提交进行搜索
natural_language_search.py	py	Wenmai-web/views	处理自然语言搜索的视图逻辑, 包括展示搜索页面、接收用户的自然语言输入并提交查询
natural_language_search_service.py	py	Wenmai-web/services	提供自然语言搜索的业务逻辑, 解析用户的自然语言查询并调用相关搜索服务

natural_language_search_mapper.py	py	Wenmai-web/mappers	与数据库或搜索引擎交互，处理自然语言查询并返回匹配的结果
-----------------------------------	----	--------------------	------------------------------

4 时序图（顺序图）



2.2.2.3 个性化推荐

1、简介

在医疗文献检索系统中，个性化推荐模块是一个重要的功能，它致力于为不同用户提供契合其需求的医疗文献推荐服务。不仅面向医疗专业人员、研究人员和患者等普通用户，同时也为管理员提供特定的管理和监控功能。该模块借助数据分析与机器学习技术，深入剖析用户与文献的关联，实现精准推荐。管理员能够通过该模块维护系统的推荐策略和数据，确保推荐服务的稳定、高效运行，而普通用户则能更便捷地获取有价值的医疗文献。

2、功能列表

用户行为分析：通过分析用户的检索历史、浏览记录、下载行为等数据，构建精准的用户画像。系统能够深入了解用户的兴趣偏好、专业领域需求。

推荐文献：为用户提供个性化的文献推荐。

2.2.2.3.1 用户行为分析

1 功能设计描述

用户画像构建：收集用户的基本信息（如职业、专业领域）、检索历史、浏览记录、收藏偏好等多维度数据，为每个用户构建详细的画像，以全面了解用户的兴趣和需求特点。

收集用户反馈：允许用户对推荐结果进行评价（如喜欢、不喜欢、有用、无用等），收集用户的反馈信息，以便不断优化推荐算法，提高推荐的准确性和质量。

2 类

1) User(用户)

表示使用医疗文献检索系统的用户，具有唯一标识（如用户 ID）、姓名、年龄、专业等基本信息。

2) UserBehaviorAnalysis (用户行为分析)

作为核心类，负责协调和管理用户行为分析的各项功能，包括用户画像构建和用户反馈收集。

3) UserProfileBuilder (用户画像构建器)

负责构建用户画像。它会从用户的检索历史、浏览记录等数据中提取特征，生成用户画像。

4) UserFeedbackCollector (用户反馈收集器)

负责收集用户对检索结果、系统功能等方面的反馈信息。

5) UserProfile (用户画像)

存储用户画像的相关信息，例如用户的兴趣领域、常用检索词、对文献类型的偏好等。

6) UserFeedback (用户反馈)

存储用户反馈的具体内容，包括反馈时间、反馈类型（如满意度评价、功能建议等）以及反馈描述。

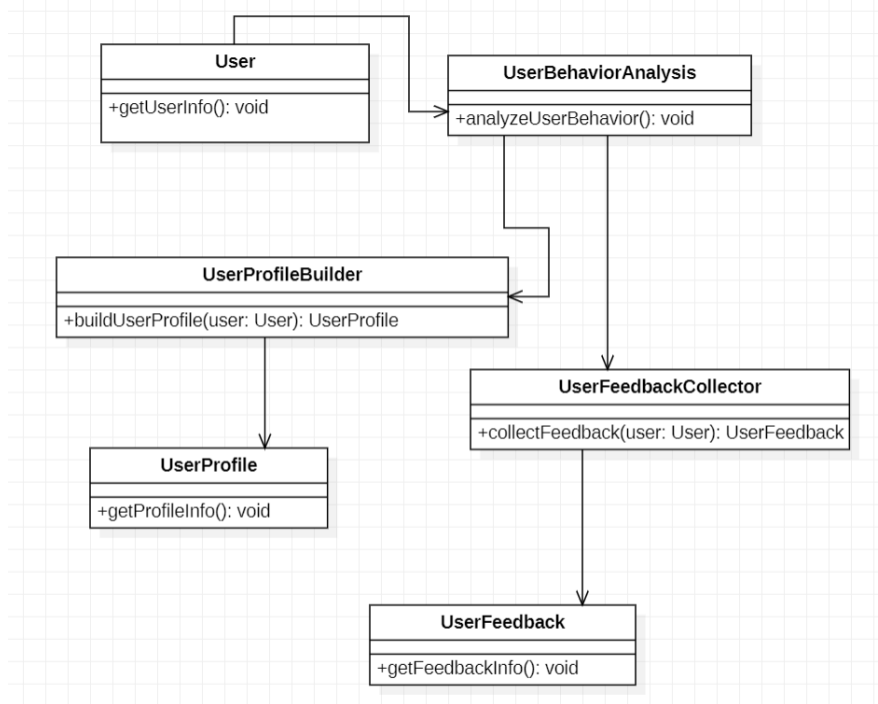


图2.2.2.3.1 用户行为分析类图

3 文件列表

名称	类型	存放位置	说明
user_basic_info.csv	CSV	/data/user_profile/ user_basic_info/	这个目录专门用于存放用户画像相关的基础信息文件，将用户基本信息文件放在此目录下，便于管理和维护
search_history_YYYYMMDD.json	JSON	/data/user_profile/ search_history/	按日期生成检索历史文件，方便进行数据的归档和查询。检索历史数据较多，单独设立目录便于管理
browse_history_YYYYMMDD.json	JSON	/data/user_profile/ browse_history/	与检索历史文件类似，按日期存放浏览记录文件，便于数据的组织和分析
download_history.csv	CSV	/data/user_profile/ download_history/	将下载记录文件存储在独立目录下，方便对用户的下载行为进行统计和分析
user_profile.json	JSON	/data/user_profile/	作为用户画像构建的最终结果文件，直接存放在用户画像数据的

			根目录下，方便系统其他模块调用
feedback_records_YYYYMMDD.json	JSON	/data/user_feedback/feedback_records/	按日期记录用户反馈信息，便于跟踪用户反馈的时间序列变化
user_behavior_analysis_config.json	JSON	/config/user_behavior_analysis/	存储用户行为分析模块的各种配置参数，如数据采集频率、画像更新周期等，方便系统管理员进行配置管理
user_behavior_analysis_log_YYYYMMDD.log	LOG	/logs/user_behavior_analysis/	记录用户行为分析模块的运行日志，包括数据处理过程中的错误信息、关键操作记录等，便于系统维护和故障排查
user_profile_display.html	html	/web/templates/user_behavior_analysis/	该页面用于向用户展示生成的用户画像信息，例如用户的兴趣领域、专业偏好等。它通过与后端交互获取数据，并以直观的方式呈现给用户。“/web/templates/”是存放系统网页模板的目录，“user_behavior_analysis”子目录专门用于存放与用户行为分析相关的页面
feedback_submission.html	html	/web/templates/user_behavior_analysis/	提供给用户一个界面来提交对系统的反馈，包括满意度评价、功能建议等。页面包含相应的表单元素，用户填写后提交给后端处理
user_profile_builder.py	python	/app/user_behavior_analysis/profile_builder/	该文件包含构建用户画像的主要逻辑函数。它从用户基本信息、检索历史、浏览记录等文件中读取数据，经过一系列的数据处理和特征提取，生成用户画像数据，并将其存储到相应的文件中
feedback_collector.py	python	/app/user_behavior_analysis/feedback_collection/	负责处理用户在前端页面提交的反馈信息。它接收前端传递的数据，进行验证和存储，将反馈信息写入反馈记录文件中
data_io.py	python	/app/user_behavior_analysis/utills/	包含一系列用于读取和写入各种数据文件（如 CSV、JSON）的函数。这些函数被用户画像构建和反馈收集模块调用，实现数据的持久化和读取操作
main_behavior_analysis.py	python	/app/user_behavior_analysis	作为用户行为分析模块的入口程序，负责协调各个子模块的工作。

		/	它可以定时调用用户画像构建和反馈收集模块，实现数据的定期更新和分析。
api.py	python	/app/ user_behavior_analysis /	定义了与前端页面交互的 API 接口。例如，提供获取用户画像数据的接口给 “user_profile_display.html” 页面，接收 “feedback_submission.html” 页面提交的反馈数据等

4 时序图（顺序图）

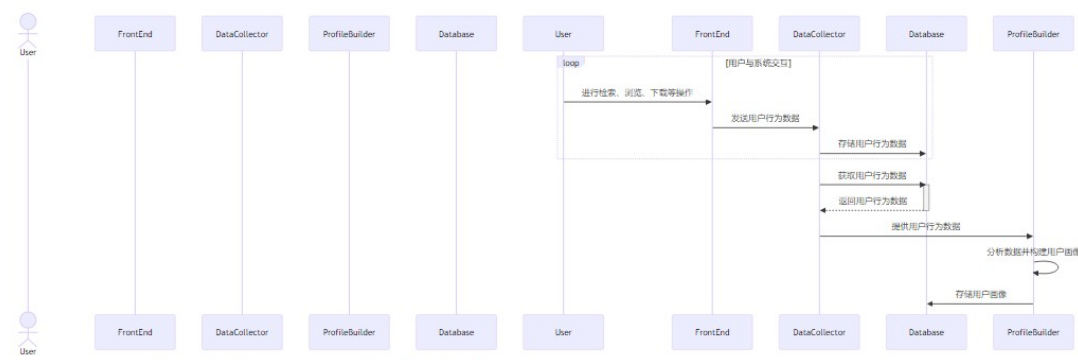


图2.2.2.3.1.1 用户画像构建时序图

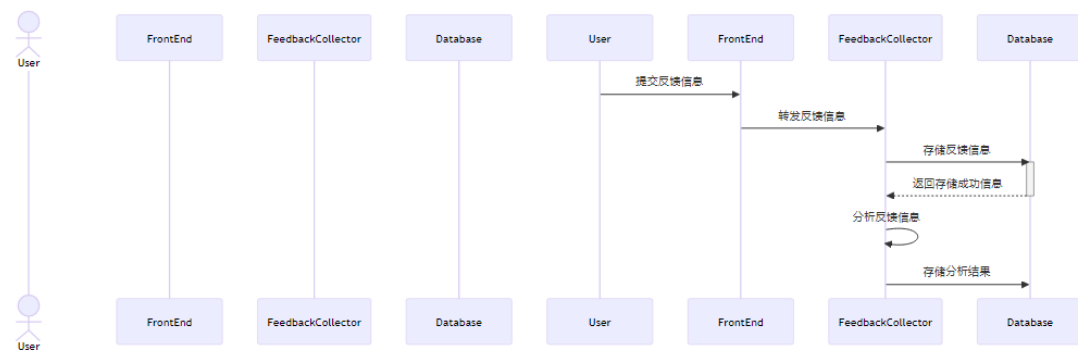


图2.2.2.3.1.2 收集用户反馈时序图

2.2.2.3.2 推荐文献

1 功能设计描述

推荐算法引擎：运用协同过滤算法（基于用户行为的相似性，推荐相似用户喜欢的文献）、内容过滤算法（根据文献的内容特征，如主题、关键词等，匹配用户兴趣）以及深度学习算法（挖掘复杂的用户 - 文献关系）等多种算法，对文献进行筛选和排序，生成个性化推荐列表。

推荐策略管理：管理员可以配置和调整推荐算法的参数，添加或删除特定的推荐规则，以适应不同的业务需求和用户群体。

推荐结果展示：以直观、清晰的方式将推荐的文献展示给用户，包括文献的标题、作者、摘要、发表年份等关键信息，并提供便捷的查看和下载入口。

2 类

1) MedicalLiterature (医疗文献)

表示医疗领域的文献，包含文献的唯一标识（如文献 ID）、标题、作者、摘要、发表年份等基本信息。

2) User (用户)

使用医疗文献检索系统的用户，具有用户 ID、姓名、年龄、专业等基本属性，还可能包含用户画像相关信息，用于个性化推荐。

3) RecommendationSystem (推荐系统)

作为整个推荐文献功能的核心类，负责协调推荐算法引擎、推荐策略管理和推荐结果展示之间的交互。

4) RecommendationAlgorithmEngine (推荐算法引擎)

包含各种推荐算法，根据用户信息和文献数据生成推荐文献列表。

5) RecommendationStrategyManager (推荐策略管理)

管理和配置推荐策略，如选择不同的推荐算法、设置推荐的数量和类型等。

6) RecommendationResultPresenter (推荐结果展示)

将推荐算法引擎生成的推荐结果以合适的方式展示给用户，可能涉及前端页面的渲染或其他展示形式。

7) RecommendationResult (推荐结果)

存储推荐文献的列表以及相关的元数据，如推荐时间、推荐依据等。

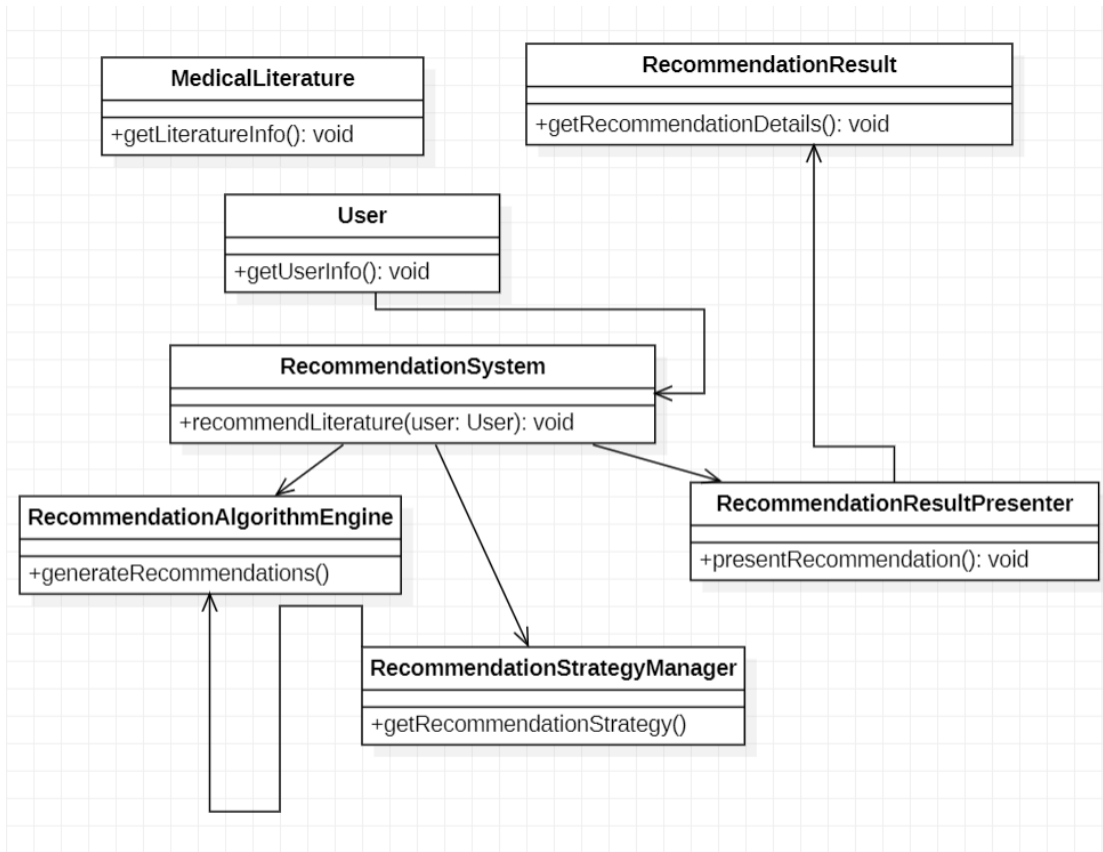


图2.2.2.3.2 推荐文献类图

3 文件列表

名称	类型	存放位置	说明
algorithm_config.json	JSON	/app/recommendation/algorithm_engine/config/	存储各种推荐算法的配置参数，如协同过滤算法的相似度计算方法、参数设置等。不同的推荐算法可能有不同的配置项，此文件方便对算法进行调整和优化
training_data.csv	CSV	/app/recommendation/algorithm_engine/data/	包含用于训练推荐算法的数据，如用户的检索历史、浏览记录、下载记录以及对文献的评分等信息。这些数据是训练算法模型的基础，通过对历史数据的学习，算法能够更好地预测用户的兴趣和需求
algorithm_model.pkl	PKL	/app/recommendation/algorithm_engine/models/	保存训练好的推荐算法模型。当有新的用户请求推荐时，算法引擎会加载这些模型，根据用户的当前信息生成推荐结果
strategy_config.json	JSON	/app/recommendation/strategy_management/config/	定义了系统支持的各种推荐策略及其参数。例如，一种策略可能是在工作日推荐专业相关的文献，

			周末推荐科普类文献；另一种策略可能是根据用户的活跃度调整推荐的数量。此文件用于管理和配置这些策略
strategy_definitions.py	python	/app/recommendation/strategy_management/strategies/	包含实现不同推荐策略的 Python 代码。每个策略可以定义为一个类，包含选择推荐算法、设置推荐参数等逻辑。这些策略类可以根据需要在推荐系统中灵活调用
strategy_history.csv	CSV	/app/recommendation/strategy_management/history/	记录系统在不同时间使用的推荐策略信息，包括策略名称、应用时间、应用对象（用户群体或特定用户）等。通过分析策略历史记录，可以评估不同策略的效果，为后续的策略调整提供依据
recommendation_display.html	html	/web/templates/recommendation/	用于生成推荐结果展示页面的前端模板。通过与后端交互获取推荐结果数据，使用 HTML 和 CSS 进行页面布局和样式设计，将推荐的文献以直观的方式呈现给用户
recommendation_style.css	CSS	/web/static/css/recommendation/	定义推荐结果展示页面的样式，包括文字排版、颜色搭配、按钮样式等，提升用户界面的美观度和用户体验
recommendation_script.js	JavaScript	/web/static/js/recommendation/	实现前端页面的交互功能，如点击推荐文献链接跳转到详细页面、加载更多推荐文献等。通过 JavaScript 代码，可以与后端 API 进行通信，获取最新的推荐结果数据
medical_literature.db	database	/data/medical_literature/	存储系统中所有医疗文献的详细信息，包括文献的标题、作者、摘要、全文链接等。推荐算法引擎在生成推荐结果时，需要从这个数据库中获取文献数据
user_info.db	database	/data/users/	存储系统用户的基本信息，如用户 ID、姓名、年龄、专业、用户画像等。这些信息对于个性化推荐至关重要，推荐算法引擎和推荐策略管理模块都需要根据用户信息来生成和调整推荐结果

4 时序图（顺序图）

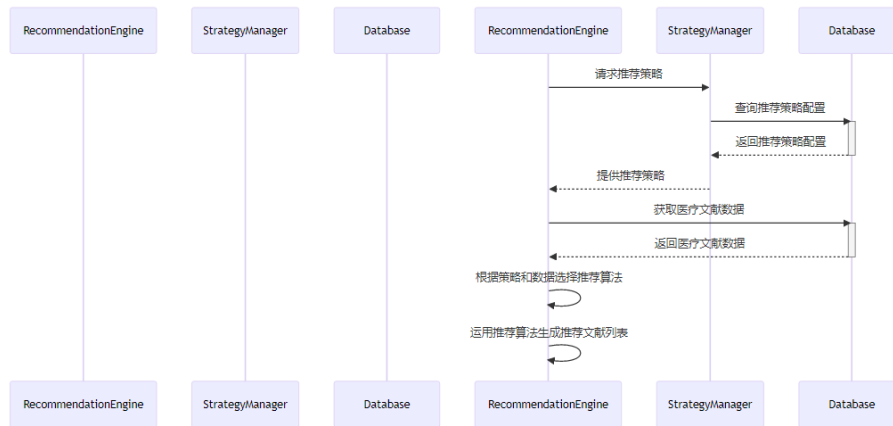


图2.2.2.3.2.1 推荐算法引擎时序图

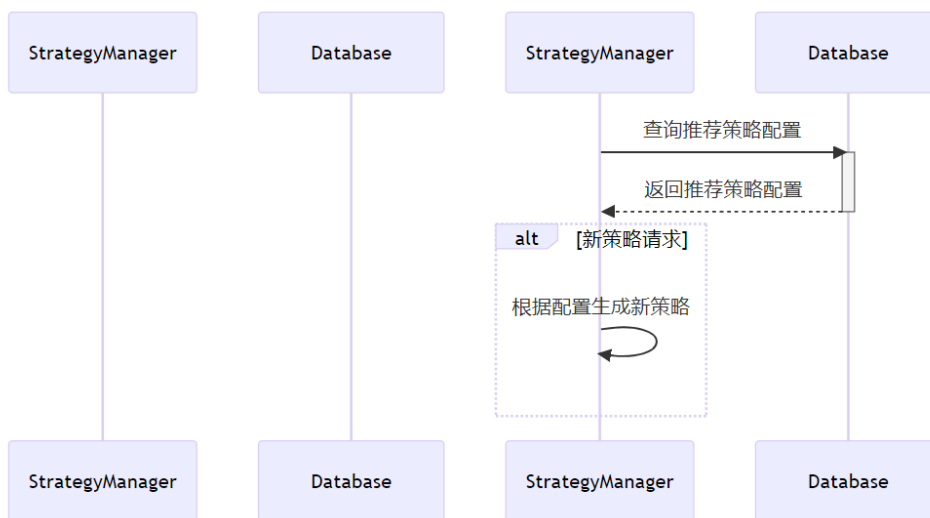


图2.2.2.3.2.2 推荐策略管理时序图

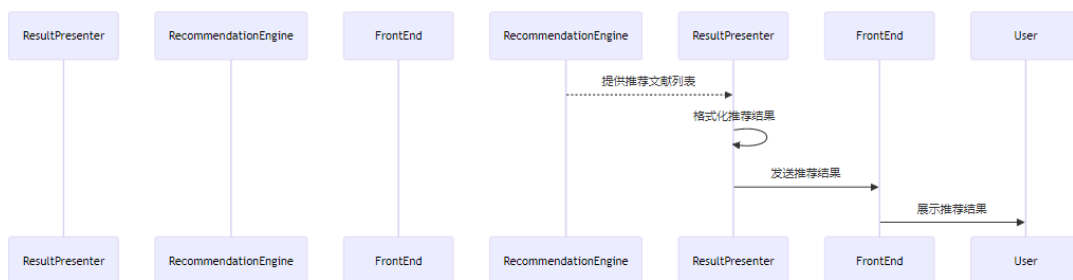


图2.2.2.3.2.3 推荐结果展示时序图

2.2.2.4 后台文献管理模块

1、简介

文献管理功能是文脉医典医学文献智能识别与检索系统的核心模块之一。该功能主要用于接收、管理和维护医学文献数据，包括文献的上传、删除及统计分析等操作。通过此功能，系统

管理员或授权用户能够对平台内的医学文献进行有效的组织、删除冗余或无效文献，并定期生成文献统计报告，以便对文献内容进行整体分析和管理工作。此功能确保平台中文献数据的准确性、完整性和时效性，提升文献管理效率。

2、功能列表

文献上传	管理员将新的医学文献上传到平台
文献删除	管理员可以删除平台内的冗余、无效或过时的文献
文献查询	管理员可根据不同的条件对文献进行查询和筛选
文献编辑	管理员可以修改文献的基本信息

2.2.2.4.1 文件上传

1 功能设计描述

文献上传模块：允许管理员或授权用户将新的医学文献上传到平台，上传的文献会经过 OCR 识别和格式转换后存储至 MySQL 数据库。

文件存储模块：上传的文献文件会被存储在服务器的文件系统中，并生成相应的文件路径，数据库中保存文献的基本信息（如标题、作者、出版日期等）。

OCR 识别模块：对于上传的图像格式文献，系统会调用 OCR 进行识别，将图像中的文本提取出来并保存为文献内容。

2 类

1) DocumentUploader

负责文献的上传，调用 OCR 模块处理图像格式文献，并通过数据库模块存储文献的基本信息。

2) OCRProcessor

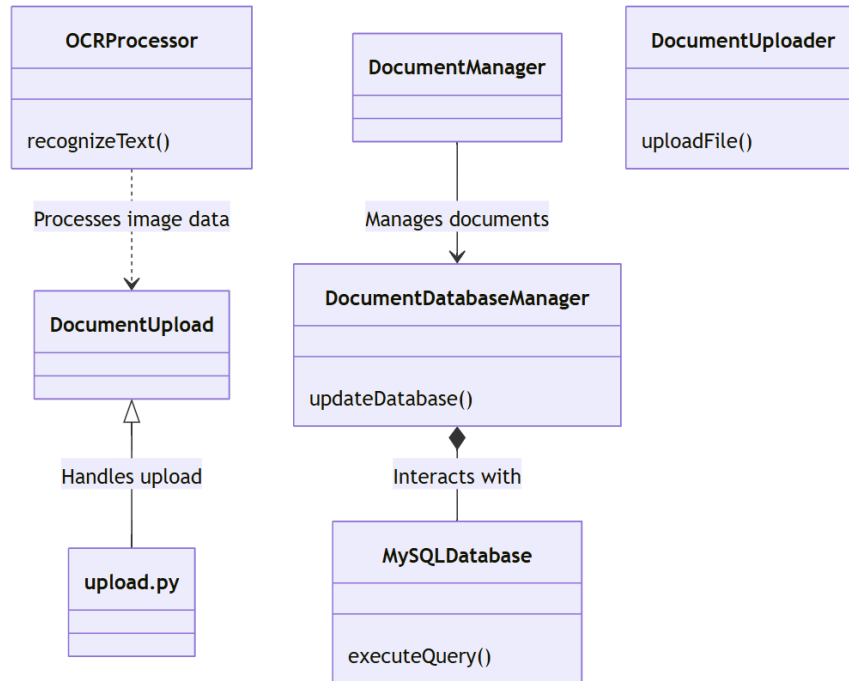
处理图像格式文献的 OCR 识别。

3) DocumentDatabaseManager

与 MySQL 数据库进行交互，负责文献元数据的存储。

4) Document

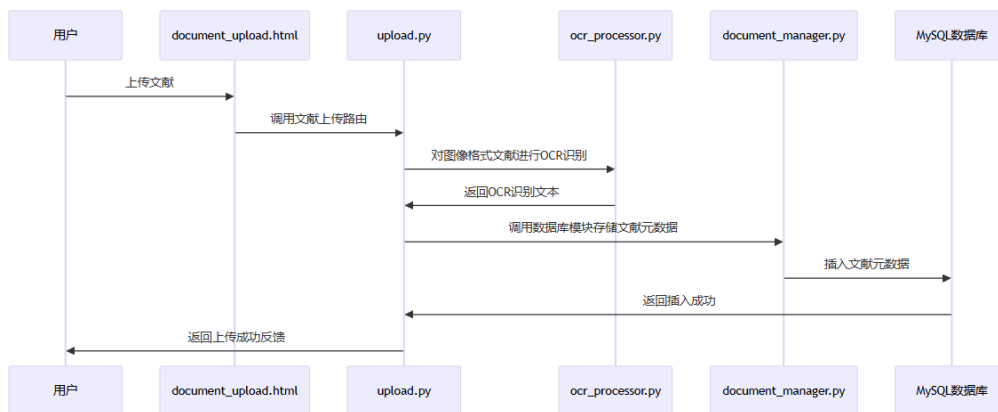
文献实体类，保存文献的基本信息。



3 文件列表

名称	类型	存放位置	说明
document_upload.html	HTML	wenmai-web/ templates/document/	文献上传页面
upload.py	Python	wenmai-web/app/ routes/document/	处理上传请求的 Flask 路由文件
ocr_processor.py	Python	wenmai-web/app/ utils/ocr/	OCR 处理模块
document_manager.py	Python	wenmai-web/app/db/	数据库操作模块
models.py	Python	wenmai-web/app/ models/	文献实体类

4 时序图 (顺序图)



2.2.2.4.2 文件删除

1 功能设计描述

文献删除模块：管理员可以选择要删除的文献。删除操作会触发数据库记录的删除和相关索引的更新。

数据更新模块：文献删除后，系统会自动更新相关统计信息。

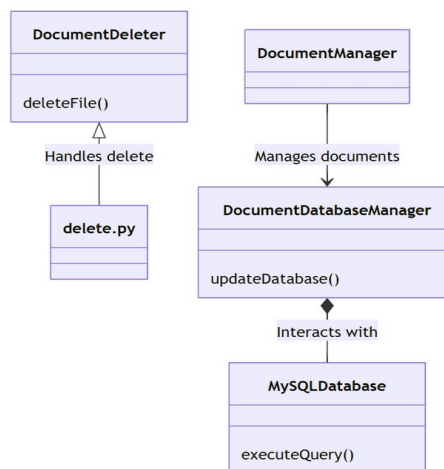
2 类

1) DocumentDeleter

负责文献删除操作，验证文献是否可以删除并删除数据库记录。

2) DocumentDatabaseManager

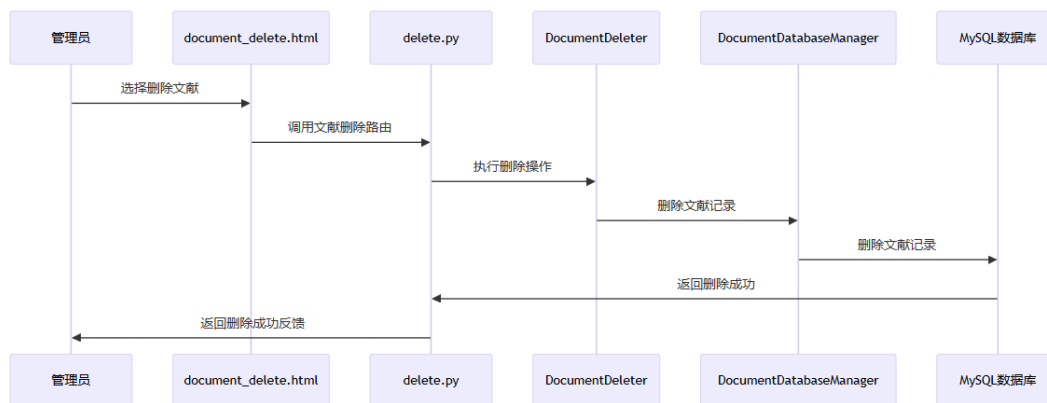
执行数据库删除操作。



3 文件列表

名称	类型	存放位置	说明
document_delete.html	HTML	wenmai-web/ templates/document/	文献删除页面
delete.py	Python	wenmai-web/app/ routes/document/	处理删除请求的 Flask 路由文件
document_manager.py	Python	wenmai-web/app/db/	数据库操作模块

4 时序图（顺序图）



2.2.2.4.3 文件查询

1 功能设计描述

文献查询模块：管理员通过指定的查询条件（如标题、作者、关键词等）查找文献。系统会根据这些条件执行查询并返回匹配的文献列表。

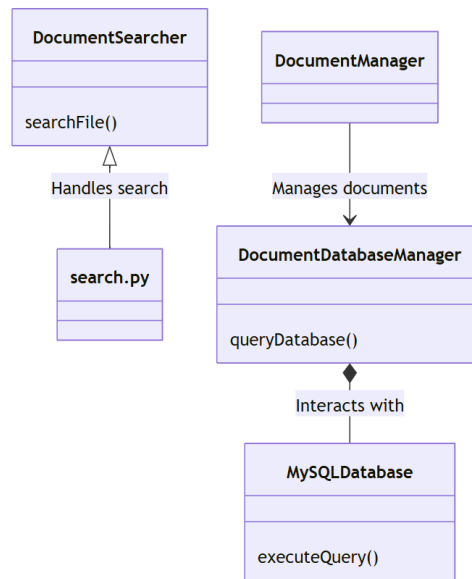
2 类

1) DocumentSearcher

负责查询操作，根据输入的查询条件搜索文献。

2) DocumentDatabaseManager

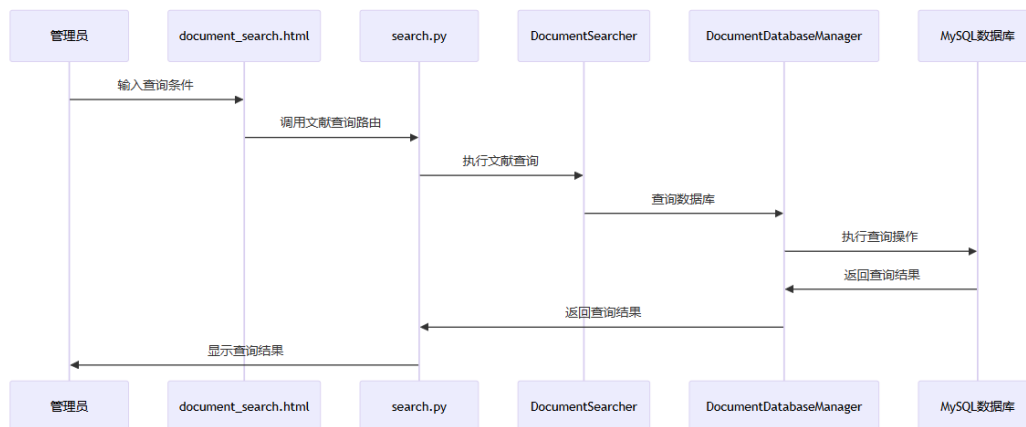
负责与 MySQL 数据库交互，执行文献查询。



3 文件列表

名称	类型	存放位置	说明
document_search.html	HTML	wenmai-web/ templates/document/	文献查询页面
search.py	Python	wenmai-web/app/ routes/document/	处理查询请求的 Flask 路由文件
document_manager.py	Python	wenmai-web/app/db/	数据库操作模块

4 时序图（顺序图）



2.2.2.4.4 文件编辑

1 功能设计描述

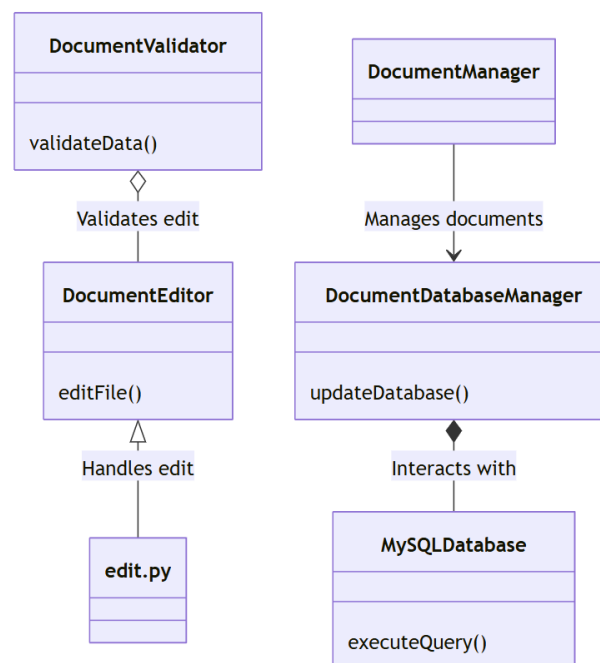
文献编辑模块：管理员可以对已有文献的基本信息进行修改，包括标题、作者、出版日期、关键词等。系统会根据提供的文献 ID 查找对应的文献记录，允许用户修改其中的字段并保存更新后的数据。

验证模块：在文献编辑时，系统将验证输入的信息是否符合格式要求，如检查日期格式、标题长度、关键词等。

数据更新模块：更新文献信息后，系统会同步更新数据库中的记录，并且如果有任何索引或引用需要更新，系统也会进行相应的处理。

2 类

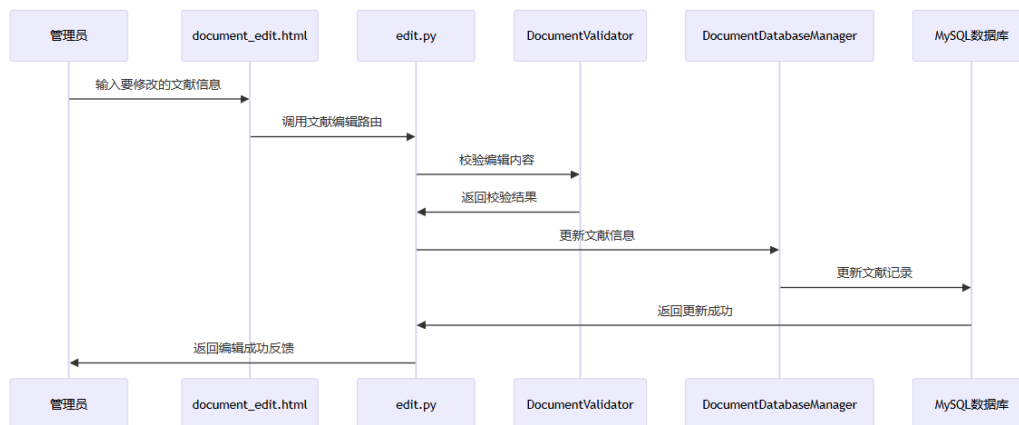
- 1) **DocumentEditor**：负责编辑文献基本信息，调用 DocumentDatabaseManager 更新数据库中的文献记录。
- 2) **DocumentDatabaseManager**：负责与 MySQL 数据库交互，更新文献记录。
- 3) **DocumentValidator**：负责对编辑的文献信息进行验证，确保输入格式正确。



3 文件列表

名称	类型	存放位置	说明
document_edit.html	HTML	wenmai-web/ templates/document/	文献编辑页面
document_edit.css	CSS	wenmai-web/static/css/	文献编辑页面的样式文件
document_edit.js	JS	wenmai-web/static/js/	文献编辑页面的脚本文件
edit.py	Python	wenmai-web/app/ routes/document/	处理编辑请求的 Flask 路由文件
document_manager.py	Python	wenmai-web/app/db/	数据库操作模块

4 时序图 (顺序图)



2.2.2.5 文献管理和组织

1、简介

该模块是医疗文献检索系统的重要组成部分，旨在帮助用户更高效地管理和利用检索到的医疗文献。通过书签、标注和笔记等子功能，用户可以对文献进行个性化的标记和记录，方便后续快速定位、回顾和整理重要信息，提升对医疗文献的处理和分析能力。

2、功能列表

书签功能：通过添加书签，能快速筛选出文献的核心内容，有助于在短时间内把握文献主旨。

标注功能：使用标注功能，可突出显示文献中的关键语句、段落，使重要信息一目了然。

笔记功能：阅读过程中，可随时记录自己的想法、疑问、见解。

2.2.2.5.1 书签功能

1 功能设计描述

添加书签：用户能够在阅读文献时，为感兴趣的页面或段落添加书签，方便快速回到特定位置。

删除书签：允许用户删除不再需要的书签，以保持书签列表的整洁。

书签管理：提供对书签的整理功能，如重命名书签、对书签进行分类等，便于用户更好地组织和查找书签。

2 类

1) MedicalLiterature (医疗文献)

代表医疗领域的文献，包含文献的唯一标识（如文献 ID）、标题、作者、摘要、发表年份等基本信息。

2) User (用户)

使用医疗文献检索系统的用户，具有用户 ID、姓名、年龄、专业等基本属性，同时拥有与书签相关的操作方法。

3) BookmarkManager (书签管理器)

负责管理用户的书签操作，包括添加书签、删除书签以及提供书签管理相关功能。

4) Bookmark (书签)

表示用户对某篇医疗文献添加的书签，包含书签的唯一标识、对应的文献以及添加书签的用户信息。

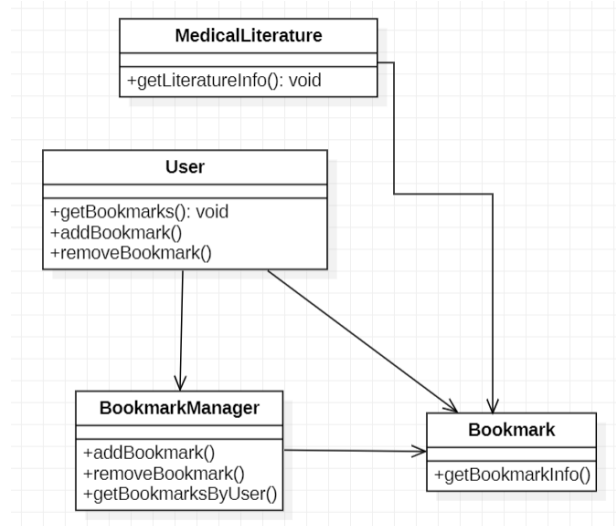


图2.2.2.5.1 书签功能类图

3 文件列表

名称	类型	存放位置	说明
bookmarks.db	database	/data/bookmarks/	这是核心的数据存储文件，用于记录所有用户的书签信息。数据库表结构可能包含书签 ID（唯一标识）、用户 ID、文献 ID、添加时间等字段。通过这个文件，系统可以持久化保存书签数据，方便进行添加、删除和查询操作
users.csv	CSV	/data/users/	存储用户的基本信息，如用户 ID、姓名、注册邮箱、密码（加密存储）等。在书签管理功能中，需要通过用户 ID 关联用户信息，例如在显示用户的书签列表时，可能需要展示用户的姓名等信息
medical_literature.csv	CSV	/data/ medical_literature/	记录系统中所有医疗文献的详细信息，包括文献 ID、标题、作者、摘要、发表期刊、发表年份等。在书签管理和展示功能中，需要根

			据文献 ID 从该文件获取文献的相关信息，以便向用户展示书签对应的文献详情
add_bookmark.py	python	/app/recommendation/ bookmark_functions/	该 Python 文件实现添加书签的具体逻辑。它接收用户 ID 和文献 ID 作为参数，与书签数据库进行交互，将新的书签记录插入到数据库中。文件中可能包含数据库连接、SQL 插入语句等操作
delete_bookmark.py	python	/app/recommendation/ bookmark_functions/	负责实现删除书签的功能。它接收书签 ID 或用户 ID 和文献 ID 作为参数，从书签数据库中删除对应的记录。文件中包含数据库连接、SQL 删除语句等逻辑
bookmark_management.py	python	/app/recommendation/ bookmark_functions/	实现书签管理的综合功能，如查询用户的所有书签、根据文献 ID 查询被哪些用户添加为书签等。该文件通过调用数据库查询语句，获取所需的书签信息，并可以对这些信息进行处理和展示
db_config.json	JSON	/config/	存储数据库连接的相关配置信息，如数据库类型（如 SQLite、MySQL 等）、数据库文件路径（对于 SQLite）、数据库服务器地址、端口、用户名和密码（对于其他数据库）等。各个功能实现文件通过读取该配置文件来建立与数据库的连接
bookmark_disp	html	/web/templates/	用于展示用户的书签列表。它通过

lay.html		recommendation/	与后端交互，获取用户的书签信息和对应的文献信息，并以表格或列表的形式展示给用户。页面中包含添加书签、删除书签等操作的按钮
bookmark_style.css	CSS	/web/static/css/ recommendation/	定义书签展示页面的样式，包括页面布局、字体、颜色、按钮样式等，提升用户界面的美观度和用户体验
bookmark_script.js	JavaScript	/web/static/js/ recommendation/	实现前端页面与后端的交互逻辑，如发送添加书签、删除书签的请求，接收后端返回的操作结果并进行相应的提示。它还可以实现一些页面动态效果，如实时更新书签列表
bookmark_operations.log	LOG	/logs/ recommendation/ bookmarks/	记录用户对书签进行的所有操作，包括添加书签、删除书签的时间、用户 ID、文献 ID 等信息。通过分析日志文件，可以了解用户的书签使用习惯，也有助于排查系统故障和安全审计

4 时序图 (顺序图)

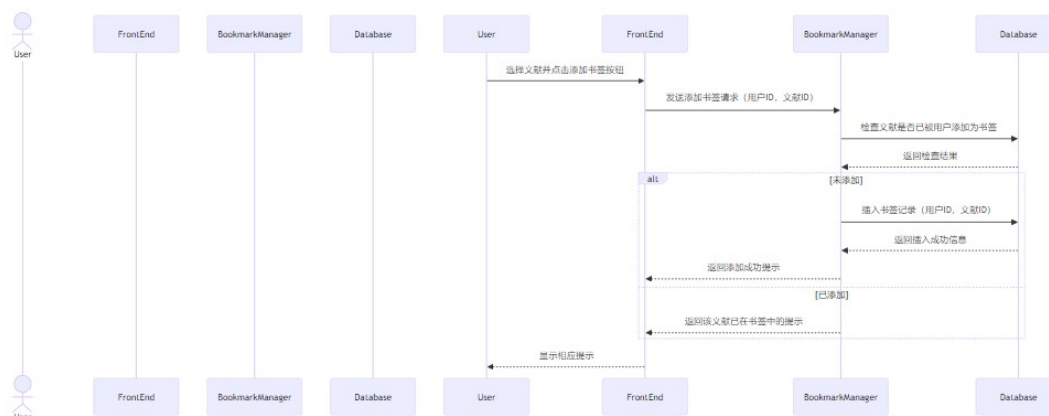


图2.2.2.5.1.1 添加书签活动图



图2.2.2.5.1.2 删除书签活动图

2.2.2.5.2 标注功能

1 功能设计描述

文本标注：用户可以在文献内容中选择特定的文本进行标注，突出显示重要信息。

标注颜色和样式设计：支持用户自定义标注的颜色和样式，以区分不同类型或重要程度的标注。

删除标注：用户可以删除不需要的标注，还原文献原始状态。

2 类

1) MedicalLiterature (医疗文献)

表示医疗领域的文献，包含文献的唯一标识（如文献 ID）、标题、作者、摘要、发表年份等基本信息。

2) Annotation (标注)

表示对文献内容的标注，包含标注的文本、关联的文献、标注颜色、标注样式以及用户信息等。

3) AnnotationColor (标注颜色)

表示标注的颜色，包含颜色的名称和对应的 RGB 值。

4) AnnotationStyle (标注样式)

表示标注的样式，例如字体大小、字体样式（加粗、斜体等）。

5) User (用户)

表示系统用户，包含用户的基本信息，如用户名、密码等，并管理用户创建的标注。

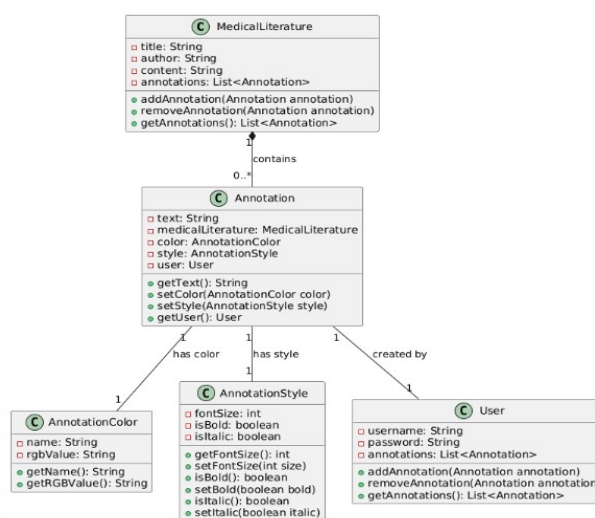


图2.2.2.5.2 标注功能类图

3 文件列表

名称	类型	存放位置	说明
annotations.db	database	/data/annotations/	用于存储所有用户对医疗文献的标注信息。数据库表结构可能包含标注 ID（唯一标识）、用户 ID、文献 ID、标注文本、标注起始位置、标注结束位置、标注颜色、标注样式等字段。通过这个文件，系统可以持久化保存标注数据，方便进行添加、查询和删除操作
users.csv	CSV	/data/users/	存储用户的基本信息，如用户 ID、姓名、注册邮箱、密码（加密

			存储) 等。在标注功能中, 需要通过用户 ID 关联用户信息, 例如在显示标注信息时, 可能需要展示标注用户的姓名等
medical_literature.csv	CSV	/data/ medical_literature/	记录系统中所有医疗文献的详细信息, 包括文献 ID、标题、作者、摘要、全文内容等。在标注功能中, 需要根据文献 ID 从该文件获取文献内容, 以便展示标注在文献中的位置和内容
text_annotation.py	python	/app/ annotation_functions/	该 Python 文件实现文本标注的具体逻辑。它接收用户 ID、文献 ID、标注文本、标注起始位置和结束位置等参数, 与标注数据库进行交互, 将新的标注记录插入到数据库中。文件中可能包含数据库连接、SQL 插入语句等操作
annotation_style.py	python	/app/ annotation_functions/	负责实现标注颜色和样式的设计功能。它可以提供一些预设的颜色和样式选项, 供用户选择, 也可以接收用户自定义的颜色和样式信息, 并将其存储到数据库中。文件中可能包含与颜色和样式处理相关的函数和数据库操作
delete_annotation.py	python	/app/ annotation_functions/	实现删除标注的功能。它接收标注 ID 或用户 ID、文献 ID 等参数, 从标注数据库中删除对应的标注记录。文件中包含数据库连接、SQL 删除语句等逻辑

db_config.json	JSON	/config/	存储数据库连接的相关配置信息，如数据库类型（如 SQLite、MySQL 等）、数据库文件路径（对于 SQLite）、数据库服务器地址、端口、用户名和密码（对于其他数据库）等。各个功能实现文件通过读取该配置文件来建立与数据库的连接
annotation_display.html	html	/web/templates/ annotation/	用于展示医疗文献及其标注信息。它通过与后端交互，获取文献内容和标注信息，并以可视化的方式展示给用户，标注部分会根据设定的颜色和样式进行显示。页面中包含文本标注、选择标注颜色和样式以及删除标注等操作的按钮
annotation_style.css	CSS	/web/static/css/ annotation/	定义标注展示页面的整体样式，包括页面布局、字体、颜色等，以及标注在文献中的显示样式，如背景颜色、边框样式等，提升用户界面的美观度和用户体验
annotation_script.js	JavaScript	/web/static/js/ annotation/	实现前端页面与后端的交互逻辑，如发送文本标注、修改标注颜色和样式、删除标注等请求，接收后端返回的操作结果并进行相应的提示。它还可以实现一些页面动态效果，如实时更新标注信息
annotation_operations.log	LOG	/logs/annotation/	记录用户对标注进行的所有操作，包括文本标注、修改标注颜色和样式、删除标注的时间、用户

		ID、文献 ID、标注 ID 等信息。通过分析日志文件，可以了解用户的标注使用习惯，也有助于排查系统故障和安全审计
--	--	---

4 时序图（顺序图）

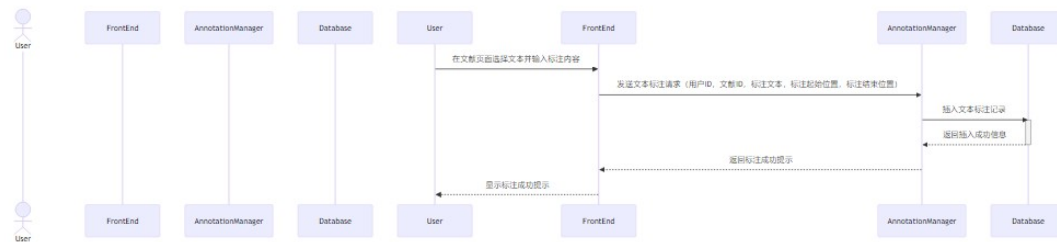


图2.2.2.5.2.1 文本标注时序图

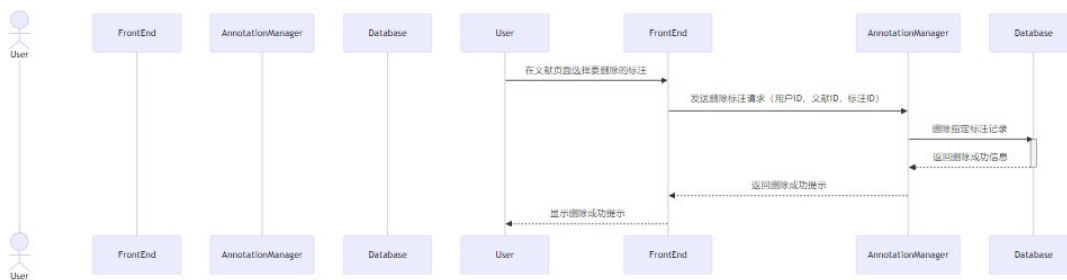


图2.2.2.5.2.2 删除标注时序图

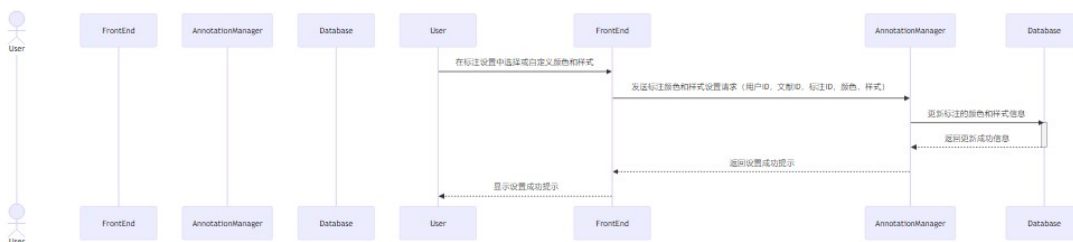


图2.2.2.5.2.3 标注颜色和样式设计时序图

2.2.2.5.3 笔记功能

1 功能设计描述

添加笔记：用户能够针对文献的特定内容添加相关的笔记，记录自己的想法、见解或疑问。

编辑笔记：支持对已添加笔记进行编辑，修改内容、格式等

删除笔记：允许用户删除不再需要的笔记。

笔记关联文献：清晰展示笔记与对应文献的关联关系，方便用户快速查看与某一文献相关的所有笔记。

2 类

1) MedicalLiterature (医疗文献)

表示医疗领域的文献，包含文献的唯一标识（如文献 ID）、标题、作者、摘要、发表年份等基本信息。

2) User (用户)

使用医疗文献检索系统的用户，具有用户 ID、姓名、年龄、专业等基本属性，与笔记功能相关联，拥有添加、编辑、删除笔记以及将笔记关联到文献的操作。

3) Note (笔记)

代表用户创建的笔记，包含笔记 ID（唯一标识）、用户 ID（关联创建该笔记的用户）、笔记内容、创建时间、最后编辑时间等信息。笔记可以与一篇或多篇医疗文献相关联。

4) NoteManager (笔记管理器)

负责管理所有与笔记相关的操作，包括添加笔记、编辑笔记、删除笔记以及处理笔记与文献的关联关系。

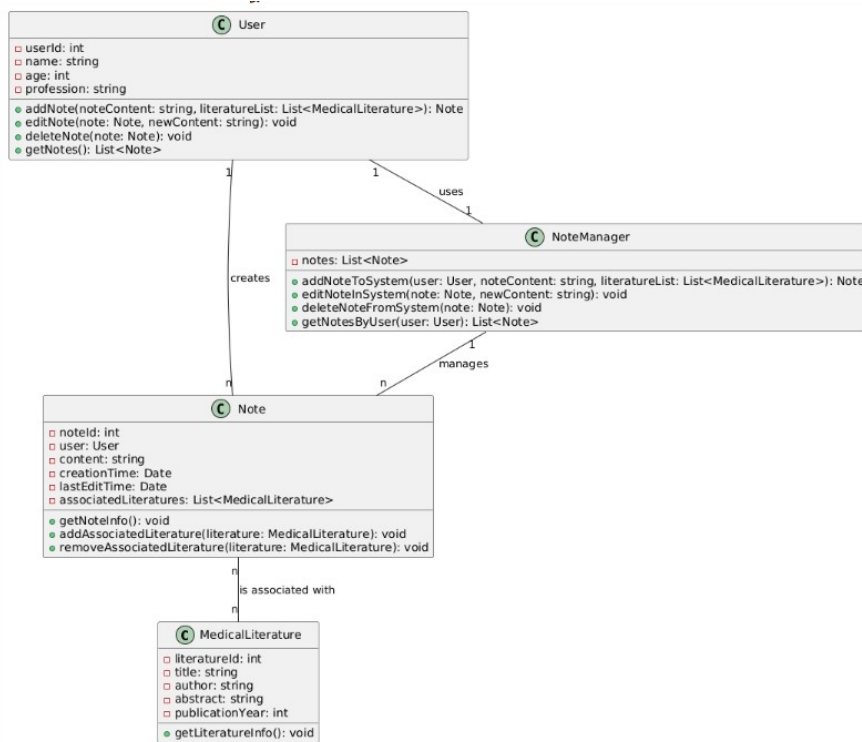


图2.2.2.5.3 笔记功能类图

3 文件列表

名称	类型	存放位置	说明
notes.db	database	/data/notes/	这是存储笔记数据的核心文件。数据库中包含多个表，如 notes 表用于记录笔记的基本信息
users.csv		/data/users/	存储用户的基本信息，如用户 ID、姓名、年龄、职业、注册邮箱等。在笔记功能中，通过用户 ID 关联用户的其他信息，例如在显示笔记时可能需要展示笔记创建者的姓名等
medical_literature.csv		/data/medical_literature/	记录系统中所有医疗文献的详细信息，包括文献 ID、标题、作者、摘要、发表年份、全文链接等。在处理笔记与文献的关联时，需要从该文件获取文献的相关信息
add_note.py		/app/note_functions/	该 Python 文件实现添加笔记的功

			能。它接收用户 ID、笔记内容以及要关联的文献 ID 列表作为参数，与笔记数据库进行交互，将新的笔记信息插入到 notes 表中，并在 note_literature_relation 表中建立笔记与文献的关联。文件中包含数据库连接、SQL 插入语句等操作
edit_note.py		/app/note_functions/	负责实现编辑笔记的功能。它接收笔记 ID 和新的笔记内容作为参数，更新 notes 表中的笔记内容和最后编辑时间。文件中包含数据库连接、SQL 更新语句等逻辑
delete_note.py		/app/note_functions/	实现删除笔记的功能。它接收笔记 ID 作为参数，从 notes 表中删除对应的笔记记录，并同时删除 note_literature_relation 表中与该笔记相关的所有关联记录。文件中包含数据库连接、SQL 删除语句等操作
associate_note_literature.py		/app/note_functions/	处理笔记与文献的关联操作。它可以接收笔记 ID 和文献 ID 列表，在 note_literature_relation 表中建立或更新笔记与文献的关联关系。文件中包含数据库连接、SQL 插入或更新语句等逻辑
db_config.json		/config/	存储数据库连接的相关配置信息，如数据库类型（如 SQLite、MySQL 等）、数据库文件路径（对于 SQLite）、数据库服务器地址、端口、用户名和密码（对于其他数据

			库) 等。各个功能实现文件通过读取该配置文件来建立与数据库的连接
note_display.html		/web/templates/note/	用于展示用户的笔记列表以及笔记详情。通过与后端交互, 获取用户的笔记信息以及关联的文献信息, 并以合适的格式展示给用户。页面中包含添加笔记、编辑笔记、删除笔记以及管理笔记与文献关联的操作按钮
note_style.css		/web/static/css/note/	定义笔记展示页面的样式, 包括页面布局、字体、颜色、按钮样式等, 以及笔记内容的显示样式, 提升用户界面的美观度和用户体验
note_script.js		/web/static/js/note/	实现前端页面与后端的交互逻辑, 如发送添加笔记、编辑笔记、删除笔记以及关联文献的请求, 接收后端返回的操作结果并进行相应的提示。它还可以实现一些页面动态效果, 如实时更新笔记列表
note_operation.log		/logs/note/	记录用户对笔记进行的所有操作, 包括添加笔记、编辑笔记、删除笔记、关联文献的时间、用户 ID、笔记 ID、文献 ID 等信息。通过分析日志文件, 可以了解用户的笔记使用习惯, 也有助于排查系统故障和安全审计

4 时序图（顺序图）

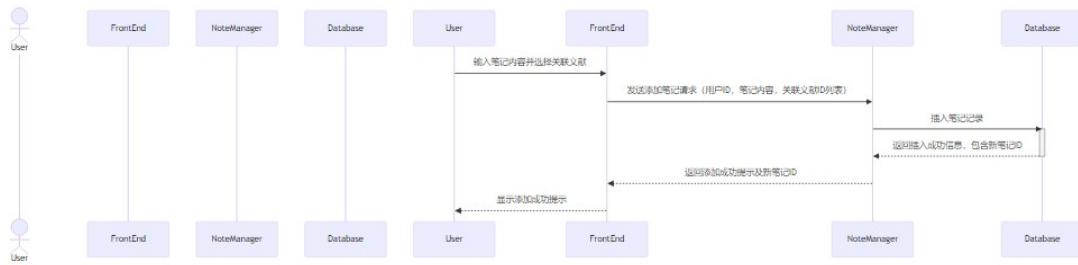


图2.2.2.5.3.1 添加笔记时序图

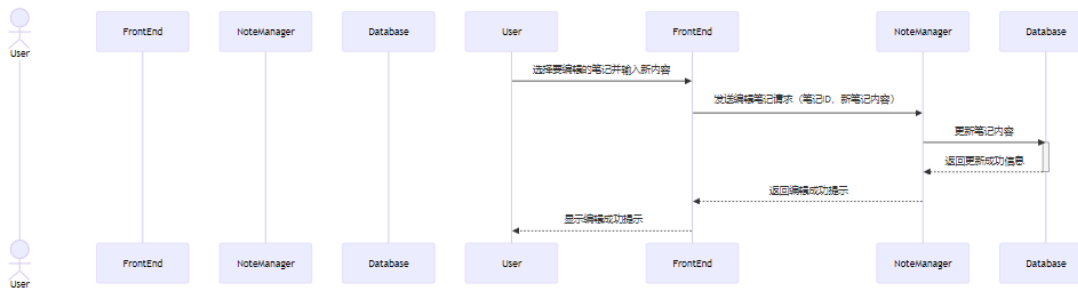


图2.2.2.5.3.2 编辑笔记时序图

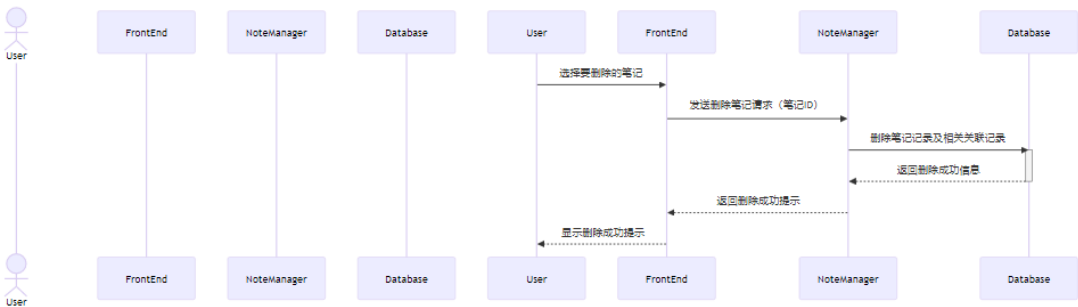


图2.2.2.5.3.3 删除笔记时序图

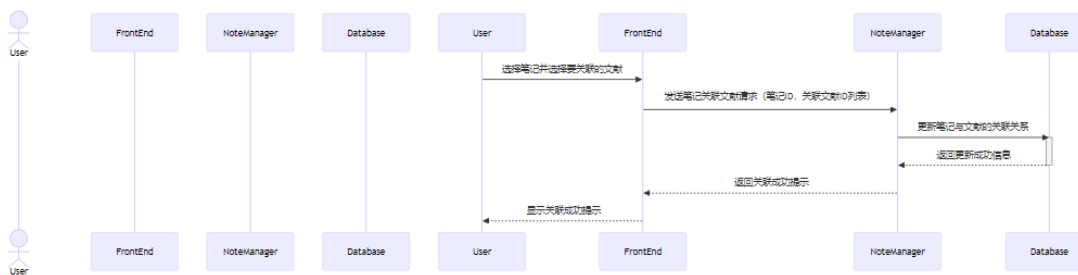


图2.2.2.5.3.4 笔记关联文献时序图

2.2.2.6 知识图谱

1、简介

知识图谱功能模块是医疗文献检索系统的重要组成部分，旨在以结构化的方式呈现医疗领域的知识，将各种医疗实体（如疾病、药物、症状、检查等）及其之间的关系清晰地展现出来。通过构建知识图谱，系统能够理解用户的问题意图，提供更精准、全面的检索结果，不仅能返回相关文献，还能展示这些文献中所涉及的实体之间的内在联系，帮助用户快速把握复杂的医疗知识体系，辅助医疗研究、临床决策等工作。

2、功能列表

知识集成与构建

可视化展示

2.2.2.6.1 知识集成与构建

1 功能设计描述

数据采集与整合：能从医学数据库、学术文献、临床指南等多源采集数据，自动处理不同格式数据，定期或手动触发采集。若数据源不可达或数据格式错误，记录日志并提示用户，提供重试或修复指引。

实体与关系处理：精准识别并分类医疗实体，抽取实体间关系。对无法识别的实体或模糊关系，标记待审；输入无效文本时提示用户。

知识融合与存储：融合多源知识，去重并解决冲突，存储于稳定可扩展的数据库（待定类型）。出现格式不一致或数据库故障，记录错误，提示用户并尝试恢复，提供人工干预途径。

2 类

1) DataCollector（数据采集器）

负责从各种数据源采集医疗相关数据，例如医疗文献数据库、临床记录、医学术语表等。它可以有不同的采集方法来适应不同类型的数据来源。

2) DataIntegrator (数据整合器)

将采集到的不同格式和结构的数据进行整合，使其具有统一的格式和规范，便于后续处理。

3) EntityAndRelationProcessor (实体与关系处理器)

对整合后的数据进行分析，识别出其中的实体（如疾病、药物、基因等）和它们之间的关系（如治疗关系、关联关系等）。

4) KnowledgeFuser (知识融合器)

将识别出的实体和关系与现有的知识图谱或其他知识源进行融合，以确保知识的一致性。

5) KnowledgeStorer (知识存储器)

将融合后的知识存储到知识图谱数据库中，以便后续的查询和使用。

6) MedicalData (医疗数据)

表示从各种数据源采集到的原始医疗数据，包含数据的具体内容和相关元信息。

7) IntegratedData (整合后的数据)

是经过 DataIntegrator 处理后具有统一格式的数据。

8) Entity (实体)

表示知识图谱中的实体，如疾病、药物等，具有唯一的标识和相关属性。

9) Relation (关系)

表示实体之间的关联关系，如“治疗”“关联”等，包含关系的类型和相关描述。

10) KnowledgeGraph (知识图谱)

存储融合后的知识，是一个由实体和关系构成的网络结构。

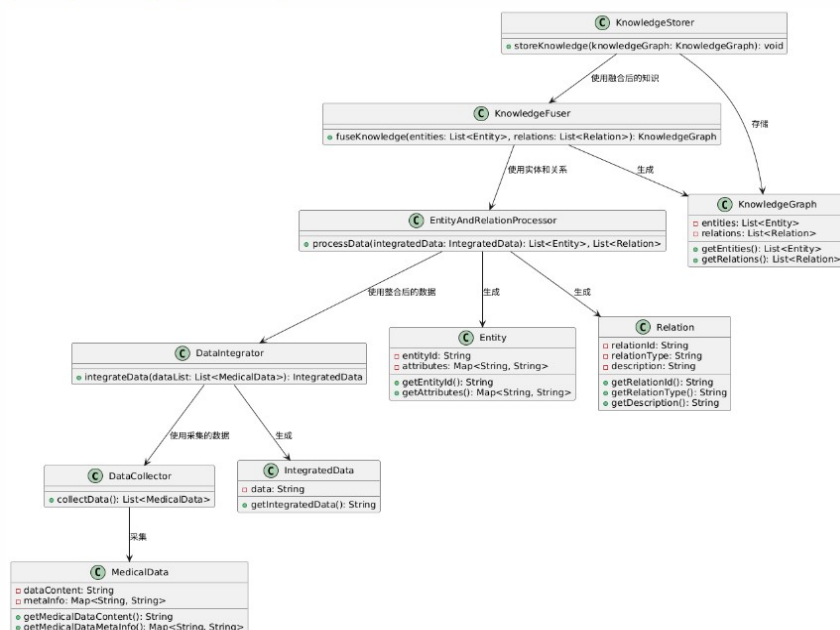


图2.2.2.6.1 知识集成与构建类图

3 文件列表

名称	类型	存放位置	说明
raw_medical_data.csv	CSV	/data/raw_data/	存储从各种数据源采集到的原始医疗数据，数据可能以表格形式（CSV）或 JSON 格式存储，包含文献文本、临床记录等内容
integrated_data.csv	CSV	/data/integrated_data/	保存经过数据整合器处理后具有统一格式的数据，便于后续的实体与关系处理
knowledge_graph.db	database	/data/ knowledge_graph/	存储融合后的知识图谱数据，数据库结构包含实体表、关系表等，用于持久化存储知识图谱中的实体和关系信息
data_collection.py	python	/app/ knowledge_graph/ data_collection/	实现从不同数据源采集医疗数据的功能，可能包括从数据库查询、文件读取、网络爬虫等操作。文件中包含针对不同数据源的采集函数，如从医疗文献数据库采集文

			献数据的函数
data_integratio n.py	python	/app/ knowledge_graph/ data_integratio n/	负责将采集到的原始数据进行整合，使其具有统一的格式。文件中包含数据清洗、格式转换、数据标准化等函数，以确保数据的一致性和可用性
entity_relation_ processing.py	python	/app/ knowledge_graph/ entity_relation_proce ssing/	对整合后的数据进行分析，识别其中的实体和关系。可能会使用自然语言处理技术、机器学习算法等。文件中包含实体识别函数、关系抽取函数等，以及相关的模型加载和配置代码
knowledge_fus ion.py	python	/app/ knowledge_graph/ knowledge_fusion/	将识别出的实体和关系与现有的知识图谱或其他知识源进行融合。文件中包含知识融合的算法实现，如冲突解决、知识更新等功能，确保知识的一致性和完整性
knowledge_sto rage.py	python	/app/ knowledge_graph/ knowledge_storage/	将融合后的知识存储到知识图谱数据库中。文件中包含数据库连接、数据插入、更新等操作的函数，根据知识图谱的结构将实体和关系数据存储到相应的表中
data_source_co nfig.json	JSON	/config/	存储各种数据源的配置信息，如数据库连接字符串、文件路径、API 接口地址等。数据采集文件通过读取该配置文件来获取数据源的相关信息，以便进行数据采集
entity_relation_ config.json	JSON	/config/	包含实体与关系处理过程中使用

config.json			的模型配置、参数设置等信息。例如，实体识别模型的路径、关系抽取算法的参数等。实体与关系处理文件通过读取该配置文件来初始化模型和设置参数
kg_db_config.json	JSON	/config/	存储知识图谱数据库的连接配置信息，如数据库类型（SQLite、Neo4j 等）、服务器地址、端口、用户名、密码等。知识存储文件通过读取该配置文件来建立与知识图谱数据库的连接

4 时序图（顺序图）

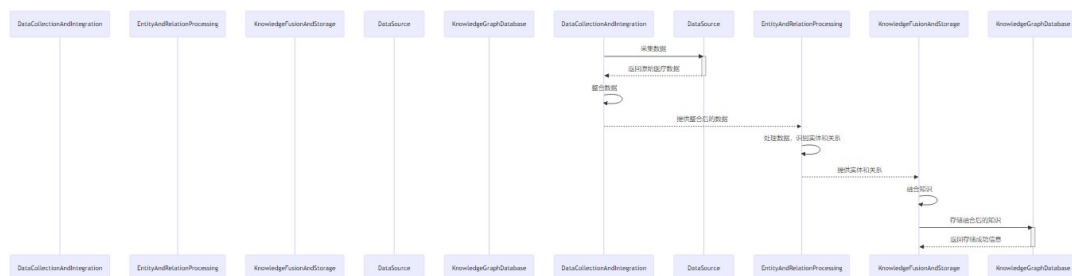


图2.2.2.6.1.1 知识集成与构建时序图

2.2.2.6.2 可视化展示

1 功能设计描述

图谱呈现：以图形化展示知识图谱，用不同颜色和样式区分实体与关系，支持缩放操作。数据量过大时采用合理策略保障性能（待定策略）。

交互操作：点击实体显示详细信息，可按实体或关系类型筛选。输入非法筛选条件或点击无效实体时，提示用户。

2 类

1) KnowledgeGraphVisualizer (知识图谱可视化器)

这是整个可视化展示功能的核心类，负责协调图谱呈现和交互操作相关的功能。它与其他类进行交互，以实现知识图谱的可视化展示。

2) GraphRenderer (图谱呈现器)

负责将知识图谱的数据转换为可视化的图形表示，例如使用图形绘制库（如 D3.js、Cytoscape.js 等）将实体和关系以图形的形式展示在界面上。

3) InteractionHandler (交互操作处理器)

处理用户与可视化图谱的交互操作，如点击实体查看详细信息、缩放图谱、拖动实体等操作。它接收用户的交互事件，并根据这些事件更新图谱的显示或执行其他相关操作。

4) KnowledgeGraph (知识图谱)

表示知识图谱的数据结构，包含实体和关系的信息。它提供方法来获取图谱的相关数据，供可视化展示部分使用。

5) VisualizationSettings (可视化设置)

存储知识图谱可视化的各种设置，如节点（实体）的颜色、大小、形状，边（关系）的样式等。这些设置可以影响图谱的外观和呈现效果。

6) UserInteraction (用户交互)

封装用户与可视化图谱的交互信息，例如点击事件、鼠标移动事件等。InteractionHandler 会根据这些交互信息来执行相应的操作。

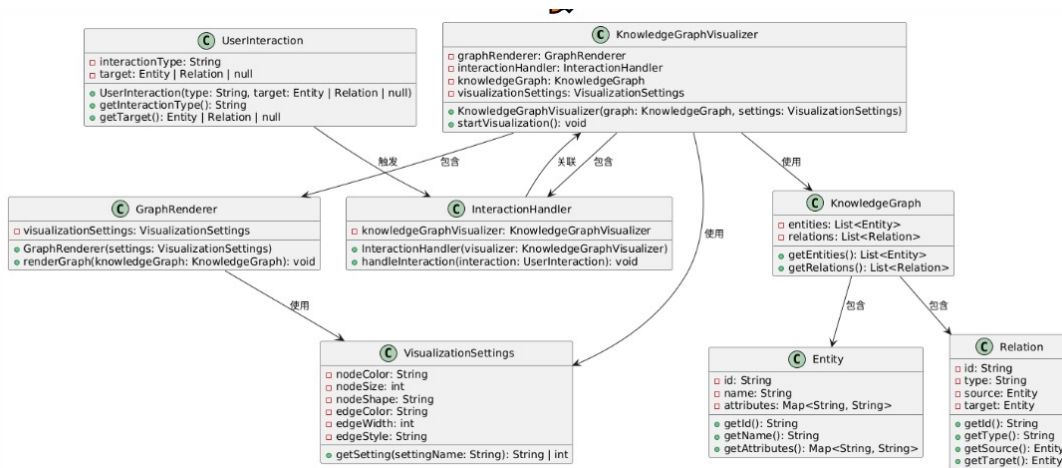


图2.2.2.6.2 可视化展示类图

3 文件列表

名称	类型	存放位置	说明
----	----	------	----

knowledge_graph_data.json	JSON	/data/ knowledge_graph/	存储知识图谱的实体和关系数据。 如果是 JSON 格式，可能以对象数组的形式存储每个实体及其属性，以及关系的描述；如果是 CSV 格式，则可能通过不同的列来表示实体 ID、属性、关系类型等信息
visualization.html		/web/templates/ knowledge_graph/	这是知识图谱可视化展示的前端页面，包含 HTML 结构用于展示可视化区域，以及与后端交互获取数据和处理用户交互的代码。通过引入相关的 JavaScript 和 CSS 文件，实现知识图谱的可视化呈现和交互操作
visualization_service.py		/app/knowledge_graph/	提供后端服务，负责处理前端的请求，从知识图谱数据文件中读取数据，并将其处理成适合前端可视化展示的格式。它还可以处理一些与可视化相关的业务逻辑，如根据用户权限返回不同的图谱数据
visualization_settings.py		/app/knowledge_graph/	存储知识图谱可视化的各种设置参数，如节点的默认颜色、大小、形状，边的样式等。这些设置可以在运行时进行调整，以满足不同的可视化需求

4 时序图（顺序图）

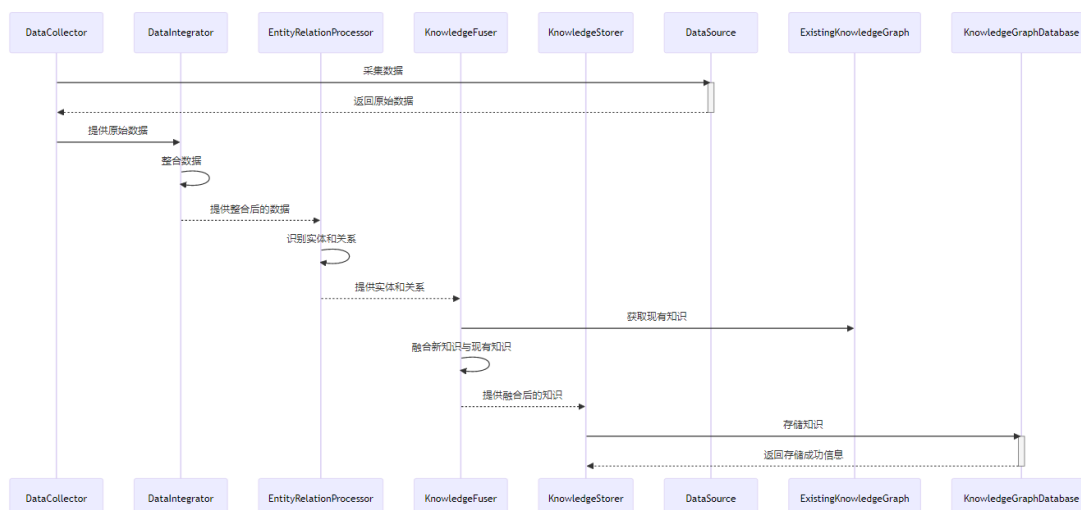


图2.2.2.6.2.1 可视化展示时序图

2.2.2.7 文献展示与阅读模块

1、简介

该模块旨在为用户提供便捷、清晰的医学文献阅读体验，涵盖文献基本信息展示、引用次数与阅读量统计、支持 PDF/HTML 格式的文献展示及下载打印等操作，帮助用户获取文献详细信息并进行阅读和进一步分析。

2、功能列表

显示文献基本信息：展示文献标题、作者、发表时间、期刊名称、DOI、摘要等基本信息，支持查看详细信息，包括关键词、来源、文献类别等。

显示文献引用次数和阅读量等：实时显示文献被引用次数、阅读量（页面浏览量、PDF 下载量等），并实时更新统计数据。

PDF/HTML 格式的文献展示：用户可选择以 PDF 或 HTML 格式查看文献，支持逐页浏览、缩放、文本搜索等功能，页面可自动切换格式。

PDF/HTML 格式的文献下载：提供文献全文 PDF 文件下载功能和打印功能，页面设有“下载全文”和“打印”按钮。

2.2.2.7.1 显示文献基本信息功能

1 功能设计描述

此功能主要通过用户输入文献 ID，系统从数据库中提取相应文献的基本信息并展示。涉及的主要类有文献信息提取类（负责从数据库获取文献数据）、信息展示类（负责将文献信息以清晰结构展示在页面上）。类图中，文献信息提取类与信息展示类相互协作，文献信息提取类将获取到的数据传递给信息展示类进行展示。

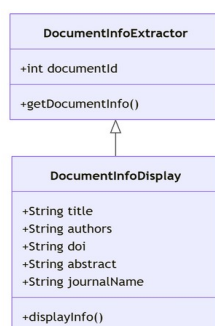
2 类

1) DocumentInfoExtractor

负责根据用户提供的文献 ID，从文献数据库中准确提取文献的各项基本信息，如标题、作者、DOI、期刊、摘要等。

2) DocumentInfoDisplay

接收文献信息提取类传来的文献数据，并将其以合适的布局和格式展示在用户界面上，确保信息清晰可读。

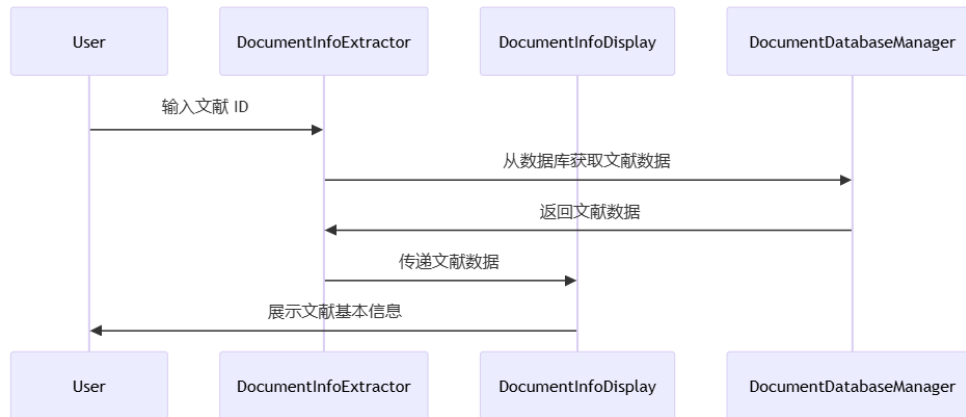


3 文件列表

名称	类型	存放位置	说明
app.py	python	Wenmai-web /app/app.py	Flask 应用的入口文件，包含了整个应用的配置、路由注册以及启动逻辑
models.py	python	Wenmai-web/app/ models.py	定义了与文献数据相关的数据库模型类
views.py	python	Wenmai-web/ app/views.py	包含了处理文献展示和阅读功能的视图函数

show_info.html	html	Wenmai-web/ templates/document/ show_info.html	用于展示文献基本信息的模板文件，展示从视图函数传递过来的文献标题、作者、发表时间、摘要等信息。
----------------	------	--	---

4 时序图（顺序图）



2.2.2.7.2 显示文献引用次数和阅读量功能

1 功能设计描述

系统根据用户输入的文献 ID，分别从第三方数据源（如 Google Scholar、Web of Science 等）获取引用次数数据，从文献库或后台统计数据获取阅读量数据，然后进行展示和实时更新。主要涉及引用数据获取类、阅读量数据获取类、数据整合展示类。引用数据获取类和阅读量数据获取类分别从不同数据源获取数据后，由数据整合展示类将数据整合并展示在相应页面。

2 类

1) CitationDataFetcher

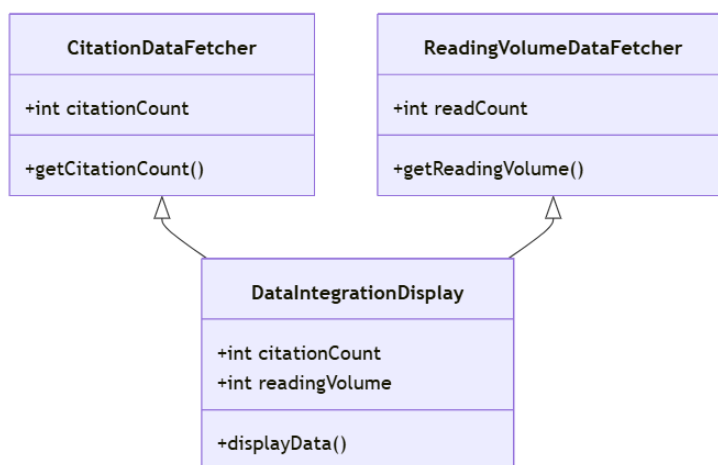
负责与第三方引用数据源建立连接，按照文献 ID 查询并获取准确的引用次数数据。

2) ReadingVolumeDataFetcher

从文献库或后台统计模块中获取文献的阅读量相关数据，如页面浏览量、PDF 下载量等。

3) DataIntegrationDisplay

接收引用数据获取类和阅读量数据获取类获取到的数据，进行整合处理后，展示在文献展示页面或分析报告中。

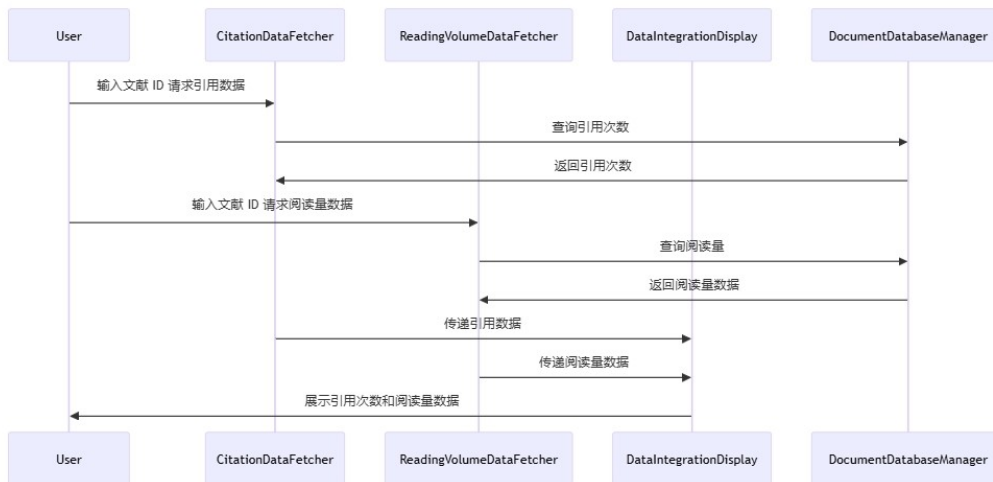


3 文件列表

名称	类型	存放位置	说明
app.py	python	Wenmai-web /app/app.py	Flask 应用的入口文件，包含了整个应用的配置、路由注册以及启动逻辑
models.py	python	Wenmai-web/app/ models.py	定义了与文献数据相关的数据库模型类
views.py	python	Wenmai-web/ app/views.py	包含了处理文献展示和阅读功能的视图函数
stats.html	html	Wenmai-web/	展示文献引用次数和阅读量统计

		templates/document/ stats.html	信息的模板，以直观的方式呈现数据，展示引用次数的趋势、阅读量的分布。
--	--	-----------------------------------	------------------------------------

4 时序图（顺序图）



2.2.2.7.3 PDF/HTML 格式的文獻展示功能

1 功能设计描述

用户选择文献查看格式（PDF 或 HTML）后，系统根据格式选择加载相应格式的文獻内容。涉及格式选择处理类、PDF 文獻展示类、HTML 文獻展示类。格式选择处理类根据用户选择调用相应的文獻展示类，PDF 文獻展示类通过嵌入 PDF 阅读器展示 PDF 文獻，HTML 文獻直接在浏览器中呈现。

2 类

1) FormatSelectionHandler

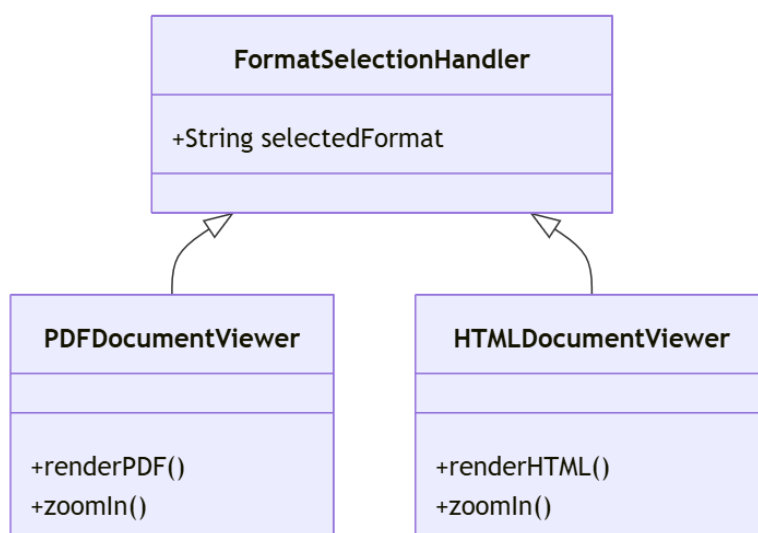
接收用户的格式选择请求，判断选择的格式并调用相应的文獻展示类进行处理。

2) PDFDocumentViewer

负责加载 PDF 格式的文献内容，利用 PDF 阅读器（如 PDF.js）实现文献的逐页浏览、缩放、搜索等功能展示。

3) HTMLDocumentViewer

将 HTML 格式的文献内容在浏览器中正确显示，确保文本、图片等元素正常展示，支持相关交互操作。

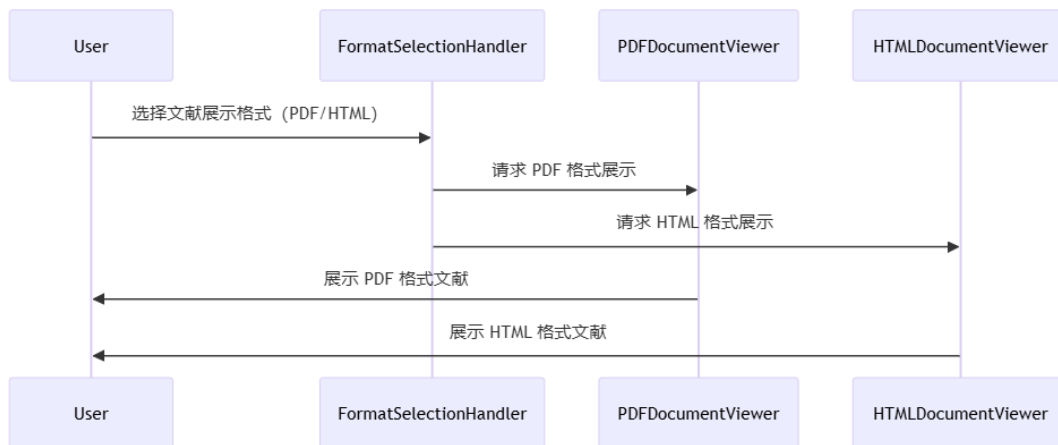


3 文件列表

名称	类型	存放位置	说明
app.py	python	Wenmai-web /app/app.py	Flask 应用的入口文件，包含了整个应用的配置、路由注册以及启动逻辑
models.py	python	Wenmai-web/app/ models.py	定义了与文献数据相关的数据库模型类
views.py	python	Wenmai-web/ app/views.py	包含了处理文献展示和阅读功能的视图函数
utils.py	python	Wenmai-web/ app/utils.py	包含一些用于文献处理的工具函数
view_pdf.html	html	Wenmai-web/	分别用于展示 PDF 格式和HTML

		templates/document/ view_pdf.html	格式文献的模板文件。
PDF.js	js	Wenmai-web/ static/pdf.js	存放 PDF.js 库的相关文件

4 时序图（顺序图）



2.2.2.7.4 PDF/HTML 格式的文献下载功能

1 功能设计描述

用户点击“下载全文”按钮或“打印”按钮时，系统根据用户请求生成 PDF 文件（若为下载则提供下载链接，若为打印则弹出打印预览）。涉及下载请求处理类、PDF 生成类、打印处理类。下载请求处理类处理用户下载请求，调用 PDF 生成类生成 PDF 文件，打印处理类处理用户打印请求并调用相关打印功能。

2 类

1) DownloadRequestHandler

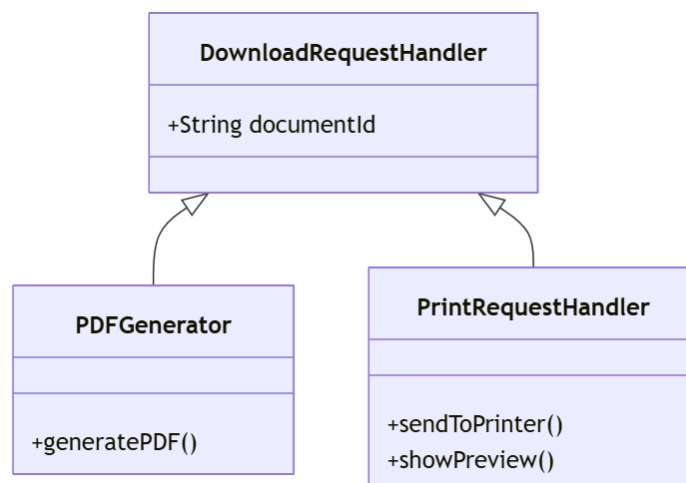
接收用户的下载请求，验证文献 ID 和下载格式的有效性，触发 PDF 生成类生成 PDF 文件并提供下载链接。

2) PDFGenerator

根据用户请求和文献内容，使用 PDF 生成库将文献转换为 PDF 格式文件，确保文件完整可下载。

3) PrintRequestHandler

接收用户的打印请求，调用浏览器的打印功能或外部打印 API，将文献内容转换为适合打印的格式并弹出打印预览。

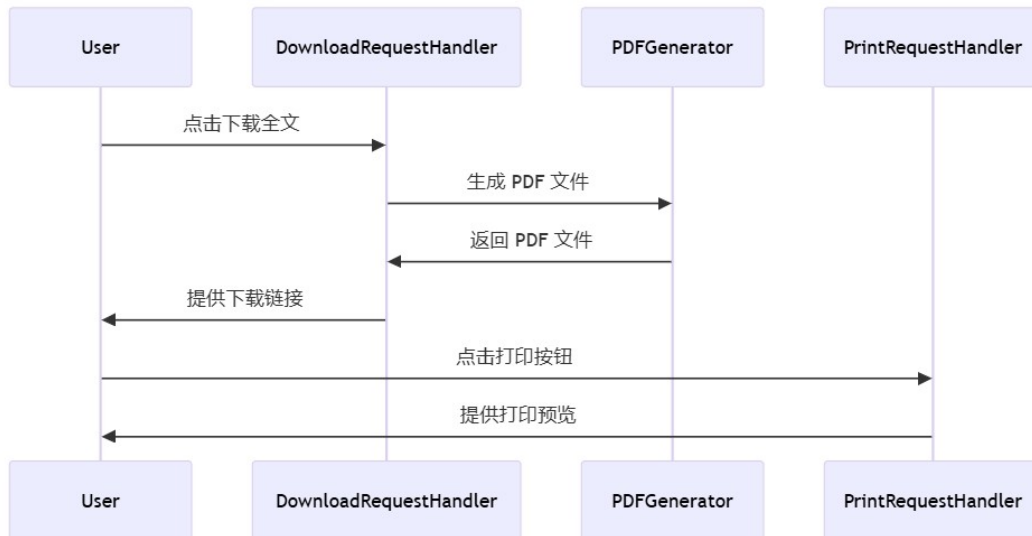


3 文件列表

名称	类型	存放位置	说明
app.py	python	Wenmai-web /app/app.py	Flask 应用的入口文件，包含了整个应用的配置、路由注册以及启动逻辑
models.py	python	Wenmai-web/app/ models.py	定义了与文献数据相关的数据库模型类
views.py	python	Wenmai-web/ app/views.py	包含了处理文献展示和阅读功能的视图函数
utils.py	python	Wenmai-web/ app/utils.py	包含一些用于文献处理的工具函数
view_pdf.html	html	Wenmai-web/	分别用于展示 PDF 格式和HTML

		templates/document/ view_pdf.html	格式文献的模板文件。
PDF.js	js	Wenmai-web/ static/pdf.js	存放 PDF.js 库的相关文件

4 时序图 (顺序图)



2.2.2.8 后台运行 - 用户管理模块

1、简介

该模块是文脉医典医学文献智能识别与检索系统的后台运维核心模块之一，主要负责系统用户的账户创建、资料更新和权限管理，确保不同用户拥有适当访问权限，维护平台安全性和操作合规性。

2、功能列表

创建 / 删除用户账户：管理员创建新用户账户并配置权限，可删除现有用户账户，系统存储或移除用户信息。

更新用户资料：管理员修改用户基本信息或权限设置，系统及时更新并保存，记录权限变更。

用户权限管理：管理员配置用户角色权限、分配用户权限，系统自动记录权限变更日志，确保权限管理的审计可追溯性。

2.2.2.8.1 创建 / 删除用户账户功能

1 功能设计描述

创建用户账户时，管理员填写用户信息和角色权限，系统生成新账户并发送激活邮件，将信息存入数据库。删除用户账户时，管理员筛选并确认删除，系统从数据库移除用户信息。涉及用户信息录入类、账户生成类、邮件发送类、用户信息删除类。用户信息录入类收集管理员输入信息，账户生成类创建账户，邮件发送类发送激活邮件，用户信息删除类执行删除操作。

2 类

1) UserInfoInputHandler

负责收集管理员输入的用户基本信息（如姓名、邮箱、联系方式、角色等）和权限设置信息，进行初步格式验证。

2) AccountCreationHandler

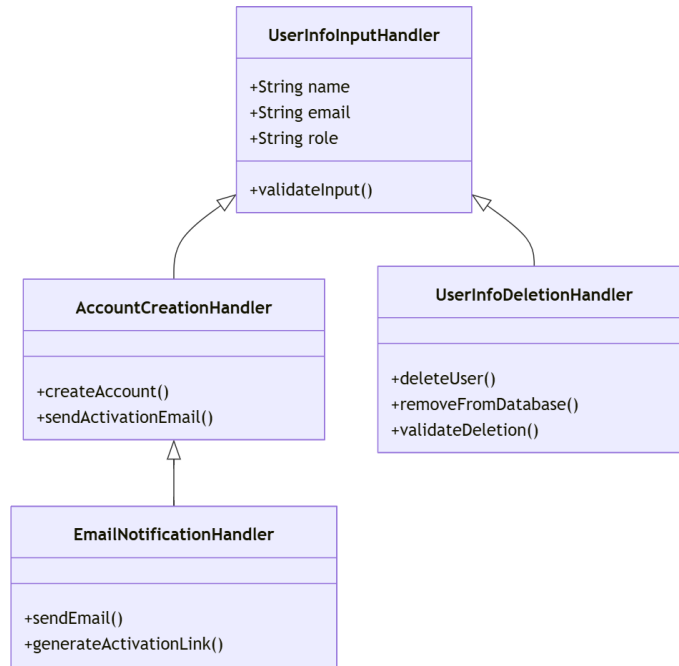
接收文献信息提取类传来的文献数据，并将其以合适的布局和格式展示在用户界面上，确保信息清晰可读。

3) EmailNotificationHandler

使用邮件发送服务，向新用户的邮箱发送激活邮件，通知用户激活账户。

4) UserInfoDeletionHandler

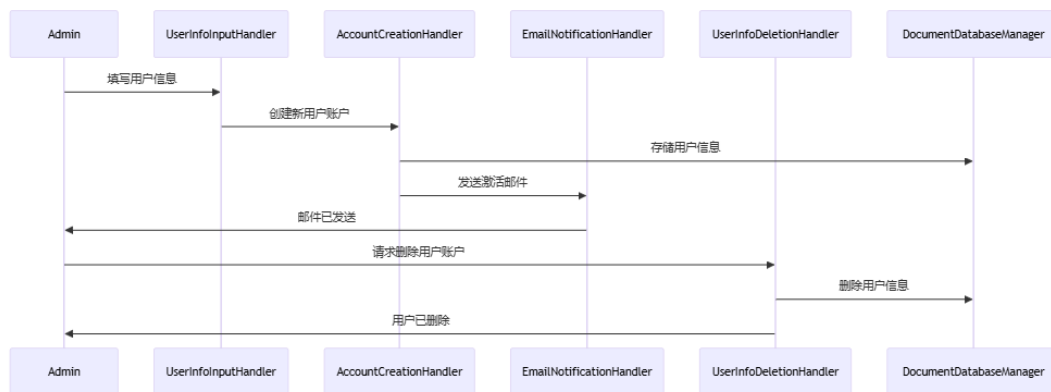
接收管理员的删除请求，在数据库中查找并彻底删除指定用户的账户信息、角色、权限等相关数据。



3 文件列表

名称	类型	存放位置	说明
app.py	python	Wenmai-web /app/app.py	Flask 应用的入口文件，包含了整个应用的配置、路由注册以及启动逻辑
models.py	python	Wenmai-web/app/models.py	定义用户相关的数据模型类
views.py	python	Wenmai-web/ app/views.py	包含处理用户管理功能的视图函数
utils.py	python	Wenmai-web/ app/utils.py	包含一些用于用户管理功能的工具函数
create.html	html	Wenmai-web/templates/use/create.html	用于创建用户的模板文件，包含用户名、密码、邮箱、用户类型等输入字段，通过表单提交数据到后端创建用户的视图函数

4 时序图 (顺序图)



2.2.2.8.2 更新用户资料功能

1 功能设计描述

管理员选择要修改资料的用户，修改基本信息或权限后保存，系统更新数据库并记录权限变更。涉及用户信息查询类、信息修改类、数据库更新类、权限记录类。用户信息查询类查找用户信息，信息修改类处理修改内容，数据库更新类更新数据库，权限记录类记录权限变更。

2 类

1) UserInfoQueryHandler

根据管理员提供的用户 ID，在数据库中查找并获取用户的当前信息，供管理员查看和修改。

2) UserInfoModificationHandler

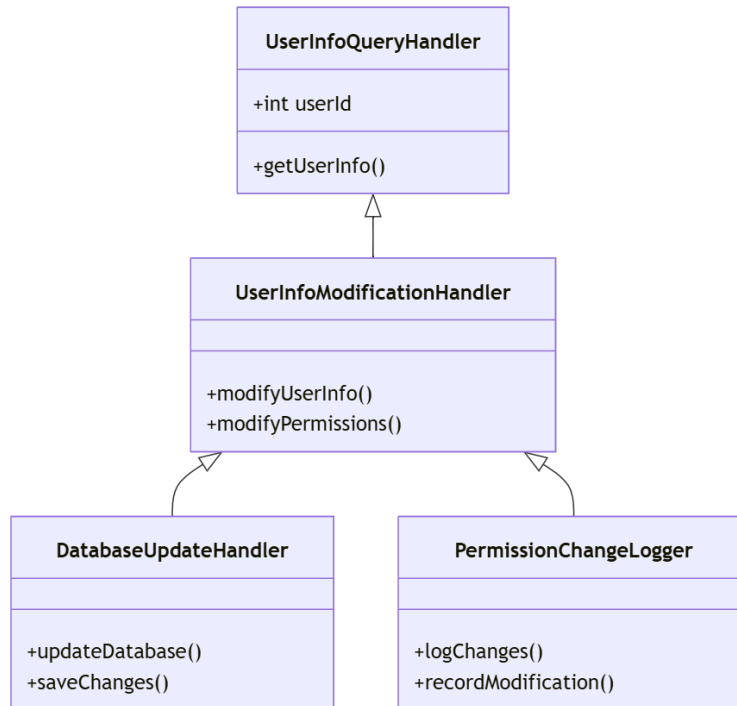
接收管理员输入的修改内容（如邮箱、姓名、联系方式、权限变更等），进行格式验证和逻辑处理。

3) DatabaseUpdateHandler

根据信息修改类提供的修改后信息，使用数据库更新语句将新信息更新到数据库中，确保数据一致性。

4) PermissionChangeLogger

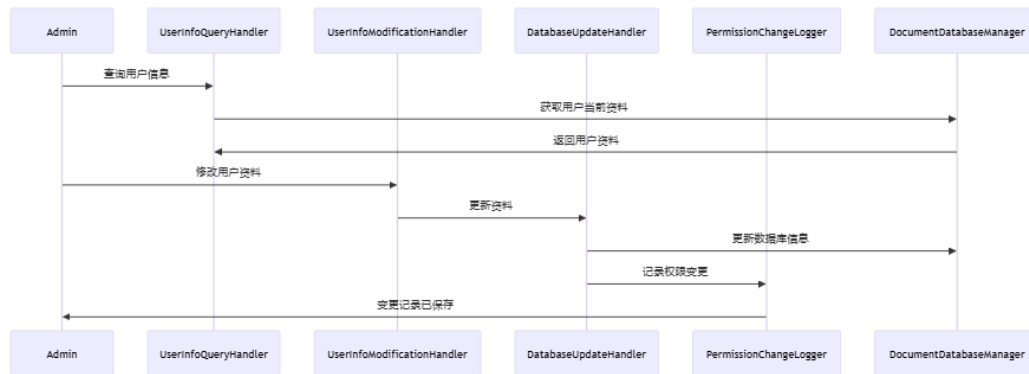
记录每次用户权限的变更情况，包括变更时间、变更内容、操作人员等信息，用于审计和追溯。



3 文件列表

名称	类型	存放位置	说明
app.py	python	Wenmai-web/ /app/app.py	Flask 应用的入口文件，包含了整个应用的配置、路由注册以及启动逻辑
models.py	python	Wenmai-web/app/ models.py	定义用户相关的数据模型类
views.py	python	Wenmai-web/ app/views.py	包含处理用户管理功能的视图函数
utils.py	python	Wenmai-web/ app/utils.py	可能包含一些用于用户管理功能的工具函数
permission_configuration.html	html	Wenmai-web/ templates/use/ permission_configuration.html	分别用于配置用户角色权限和为用户分配权限的模板

4 时序图（顺序图）



2.2.2.8.3 用户权限管理功能

1 功能设计描述

管理员配置用户角色权限和分配用户权限，系统保存配置并生效，记录权限变更日志。涉及角色权限配置类、用户权限分配类、权限同步类、日志记录类。角色权限配置类设置角色权限，用户权限分配类为用户分配权限，权限同步类确保权限同步更新，日志记录类记录权限变更。

2 类

1) RolePermissionConfigHandler

提供界面让管理员设置各个角色的权限，如管理员、研究人员、访客等的访问、修改、删除等权限，进行权限冲突检测。

2) UserPermissionAssignmentHandler

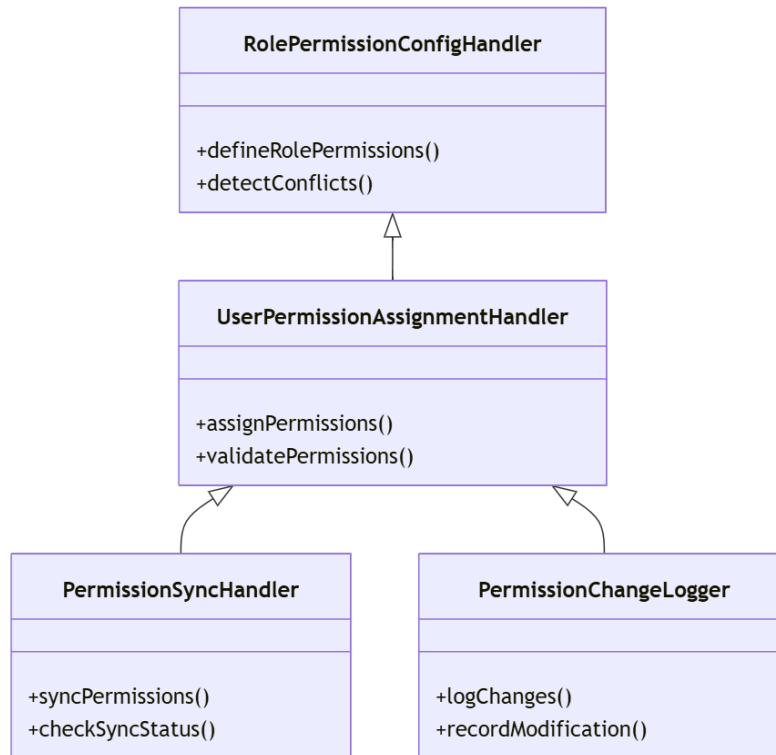
根据管理员指定，为具体用户分配相应的权限，将权限信息存储到数据库中，并确保用户下次登录时加载最新权限。

3) PermissionSyncHandler

保证用户权限在系统各模块中的实时同步，避免因权限更新不及时导致的安全问题。

4) PermissionChangeLogger

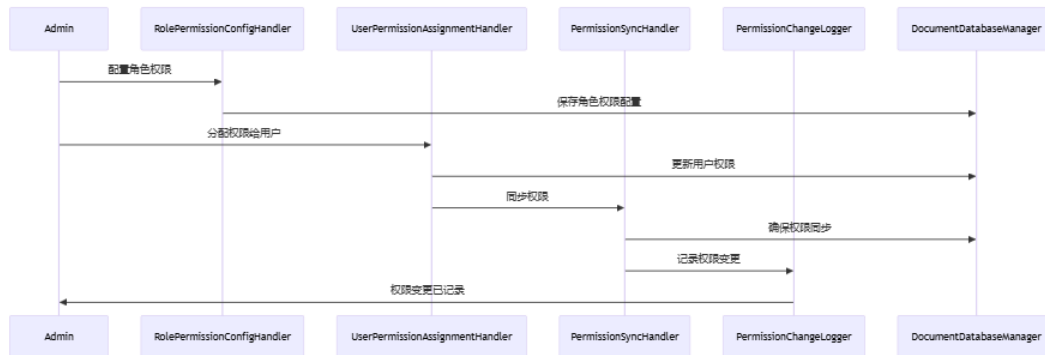
详细记录每次权限变更的操作，包括操作时间、操作人员、变更前权限、变更后权限等信息，生成审计日志。



3 文件列表

名称	类型	存放位置	说明
app.py	python	Wenmai-web/ /app/app.py	Flask 应用的入口文件，包含了整个应用的配置、路由注册以及启动逻辑
models.py	python	Wenmai-web/app/ models.py	定义用户相关的数据模型类
views.py	python	Wenmai-web/ app/views.py	包含处理用户管理功能的视图函数
utils.py	python	Wenmai-web/ app/utils.py	可能包含一些用于用户管理功能的工具函数
permission_config.html	html	Wenmai-web/ templates/use/ permission_config.html	分别用于配置用户角色权限和为用户分配权限的模板

4 时序图（顺序图）



2.2.3 接口描述

2.2.3.1 注册接口

Name名称: /api/user/register

Description说明: 用户注册接口，用户可以通过该接口注册新账户。

Definition定义:

- 请求方法: POST
- @param Object类型的一个结构体userInfo
- userInfo: {
 - o username: 用户名 (必填)
 - o email: 邮箱 (必填)
 - o password: 密码 (必填) }
- @return: 200 操作成功, 400 操作失败

2.2.3.2 登录接口

Name名称: /api/user/login

Description说明: 用户登录接口，用户通过用户名和密码进行登录。

Definition定义:

- 请求方法: POST
- @param Object类型的一个结构体loginInfo
- loginInfo: {
 - o username: 用户名 (必填)
 - o password: 密码 (必填) }
- @return: 200 登录成功, 400 登录失败

2.2.3.3 密码找回

Name名称: /api/user/password/recover

Description说明: 用户密码找回接口, 通过邮箱发送找回密码的链接或验证码。

Definition定义:

- 请求方法: POST
- @param String类型 email: 用户邮箱 (必填)
- @return: 200 操作成功, 400 操作失败

2.2.3.4 个人信息修改

Name名称: /api/user/update

Description说明: 用户个人信息修改接口, 允许用户更新个人信息。

Definition定义:

- 请求方法: POST
- @param Object类型的一个结构体updateInfo
- updateInfo: {
 - o user_id: 用户ID (必填)
 - o new_info: 用户新的信息 (必填, 可能包含 username, email, avatar 等字段) }
- @return: 200 更新成功, 400 更新失败

2.2.3.5 关键词、短语和自然语言搜索

Name名称: /api/literature/search

Description说明: 关键词、短语和自然语言搜索接口。

Definition定义:

- 请求方法: POST
- @param Object类型的一个结构体searchQuery
- searchQuery: {
 - o query: 搜索内容 (必填)
 - o search_type: 搜索类型 (必填, 'keyword', 'phrase', 'natural_language') }
- @return: 200 操作成功, 返回搜索结果, 400 操作失败

2.2.3.6 基于时间、作者、引用等条件对搜索结果进行排序

Name名称: /api/literature/sort

Description说明: 基于时间、作者、引用等条件对搜索结果进行排序。

Definition定义:

- 请求方法: POST
- @param Object类型的一个结构体sortParams
- sortParams: {
 - o results: 搜索结果列表 (必填)
 - o sort_by: 排序条件 (可选, 'time', 'author', 'citations'等) }
- @return: 200 排序成功, 返回排序后的结果, 400 操作失败

2.2.3.7 搜索结果过滤接口，支持按日期、类型等条件过滤

Name名称: /api/literature/filter

Description说明: 搜索结果过滤接口，支持按日期、类型等条件过滤。

Definition定义:

- 请求方法: POST
- @param Object类型的一个结构体filterParams
- filterParams: {
 - o results: 搜索结果列表 (必填)
 - o filters: 过滤条件 (如按日期、类型等, 必填) }
- @return: 200 过滤成功, 返回过滤后的结果, 400 操作失败

2.2.3.8 个性化推荐接口

Name名称: /api/literature/recommend

Description说明: 个性化推荐接口，根据用户的历史数据或兴趣推荐文献。

Definition定义:

- 请求方法: POST
- @param Integer类型 user_id: 用户ID (必填)
- @return: 200 推荐成功, 返回推荐的文献列表, 400 操作失败

2.2.3.9 文献书签功能接口

Description说明: 文献书签功能接口，允许用户为文献添加书签。

Definition定义:

- 请求方法: POST
- @param Integer类型 user_id: 用户ID (必填)
- @param Integer类型 doc_id: 文献ID (必填)
- @return: 200 添加成功, 400 操作失败

2.2.3.10 文献标注功能接口

Name名称: /api/document/annotate

Description说明: 文献标注功能接口，允许用户为文献添加标注。

Definition定义:

- 请求方法: POST
- @param Integer类型 user_id: 用户ID (必填)
- @param Integer类型 doc_id: 文献ID (必填)
- @param String类型 annotation: 标注内容 (必填)
- @return: 200 添加成功, 400 操作失败

2.2.3.11 文献笔记功能接口

Name名称: /api/document/note

Description说明: 文献笔记功能接口，允许用户为文献添加笔记。

Definition定义:

- 请求方法: POST
- @param Integer类型 user_id: 用户ID (必填)
- @param Integer类型 doc_id: 文献ID (必填)
- @param String类型 note: 笔记内容 (必填)
- @return: 200 添加成功, 400 操作失败

2.2.3.12 获取文献的知识图谱数据接口

Name名称: /api/knowledge/graph

Description说明: 获取文献的知识图谱数据接口，返回与文献相关的关键词、作者、引用关系等。

Definition定义:

- 请求方法: POST
- @param Integer类型 doc_id: 文献ID (必填)
- @return: 200 获取成功, 返回文献的知识图谱, 400 操作失败

2.2.3.13 获取文献的基本信息接口

Name名称: /api/document/info

Description说明: 获取文献的基本信息接口，包括标题、作者、出版时间等。

Definition定义:

- 请求方法: POST
- @param Integer类型 doc_id: 文献ID (必填)
- @return: 200 获取成功，返回文献的基本信息，400 操作失败

2.2.3.14 获取文献的展示格式接口

Name名称: /api/document/display

Description说明: 获取文献的展示格式接口，支持 PDF 和 HTML 格式。

Definition定义:

- 请求方法: POST
- @param Integer类型 doc_id: 文献ID (必填)
- @param String类型 format: 显示格式 (必填，支持 'pdf', 'html')
- @return: 200 获取成功，返回文献内容，400 操作失败

2.2.3.15 文献全文下载接口

Name名称: /api/document/download

Description说明: 文献全文下载接口，允许用户下载文献的 PDF 或 HTML 格式。

Definition定义:

- 请求方法: POST
- @param Integer类型 doc_id: 文献ID (必填)
- @param String类型 format: 下载格式 (必填，支持 'pdf', 'html')
- @return: 200 下载成功，返回文献文件，400 操作失败

2.2.3.16 文献全文打印接口

Name名称: /api/document/print

Description说明: 文献全文打印接口，允许用户打印文献的内容。

Definition定义:

- 请求方法: POST
- @param Integer类型 doc_id: 文献ID (必填)
- @param String类型 format: 打印格式 (必填，支持 'pdf', 'html')

- @return: 200 打印成功, 400 操作失败

2.2.3.17创建新用户功能接口

Name名称: /api/admin/user/create

Description说明: 创建新用户接口, 管理员可以通过此接口创建新账户。

Definition定义:

- 请求方法: POST
- @param Object类型的一个结构体userInfo
- userInfo: {
 - o username: 用户名 (必填)
 - o email: 邮箱 (必填)
 - o password: 密码 (必填)
 - o role: 角色 (必填, 'admin', 'user' 等) }
- @return: 200 创建成功, 400 创建失败

2.2.3.18删除用户账户功能接口

Name名称: /api/admin/user/delete

Description说明: 删除用户账户接口, 管理员可以删除某个用户账户。

Definition定义:

- 请求方法: POST
- @param Integer类型 user_id: 用户ID (必填)
- @return: 200 删除成功, 400 删除失败

2.2.3.19更新用户资料功能接口

Name名称: /api/admin/user/update

Description说明: 更新用户资料接口, 管理员可以更新用户的个人信息。

Definition定义:

- 请求方法: POST
- @param Integer类型 user_id: 用户ID (必填)
- @param Object类型的一个结构体updateInfo: 用户新资料 (必填)
- updateInfo: {
 - o username: 新用户名 (选配)
 - o email: 新邮箱 (选配)
 - o role: 新角色 (选配) }
- @return: 200 更新成功, 400 更新失败

2.2.3.20 获取用户日活跃报表接口

Name名称: /api/admin/analytics/daily_active

Description说明: 获取用户日活跃报表接口，提供每日用户活跃度的统计数据。

Definition定义:

- 请求方法: POST
- @param String类型 date: 日期 (必填, 格式 'YYYY-MM-DD')
- @return: 200 获取成功, 返回日活跃数据, 400 操作失败

python

复制代码

2.2.3.21 功能页面分析报告接口

Name名称: /api/admin/analytics/page_activity

Description说明: 获取功能页面分析报告接口，统计各个功能页面的访问量、行为数据等。

Definition定义:

- 请求方法: POST
- @param String类型 start_date: 起始日期 (必填)
- @param String类型 end_date: 结束日期 (必填)
- @param String类型 page_name: 页面名称 (必填)
- @return: 200 获取成功, 返回页面访问统计, 400 操作失败

2.2.3.22 用户点击率分析接口

Name名称: /api/admin/analytics/click_rate

Description说明: 获取用户点击率分析接口，分析各个功能模块的点击率。

Definition定义:

- 请求方法: POST
- @param String类型 element_id: 元素ID (必填, 如按钮ID、文献ID)
- @param String类型 start_date: 起始日期 (必填)
- @param String类型 end_date: 结束日期 (必填)
- @return: 200 获取成功, 返回点击率数据, 400 操作失败

2.2.3.23数据爬取功能接口

Name名称: /api/admin/data/crawl

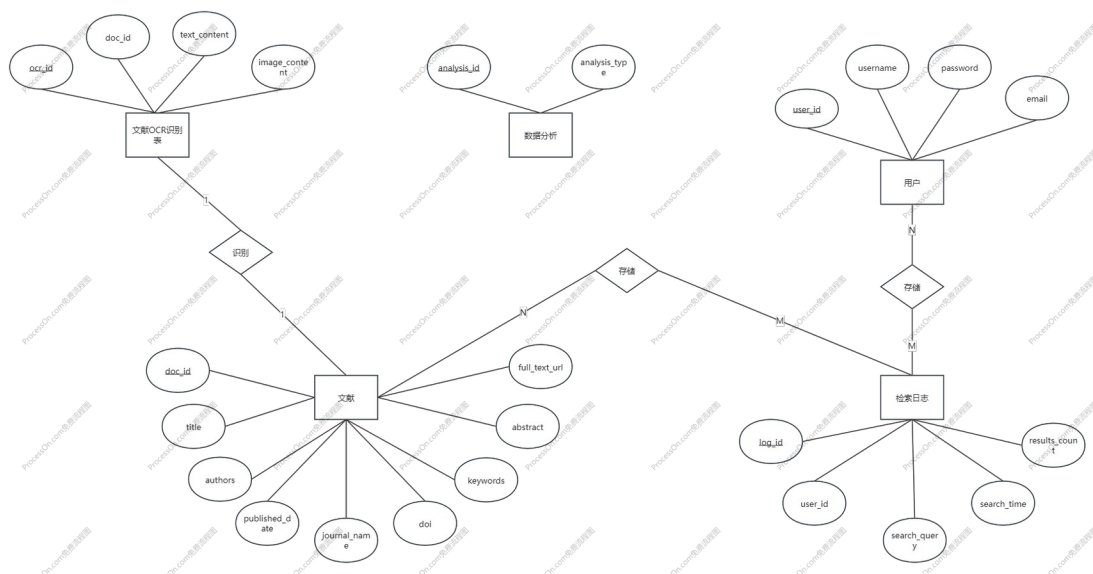
Description说明: 数据爬取接口，爬取指定的外部数据源。

Definition定义:

- 请求方法: POST
- @param String类型 source_url: 外部

3 数据结构/数据库设计

3.1 概念模型



3.2 数据库表设计

3.2.1 用户表

存储用户信息，包括注册、登录等相关数据

字段	类型	约束条件	描述
user_id	INT	主键 自动增长	用户ID
username	VARCHAR (50)	唯一 非空	用户名
User_type	VARCHAR (50)	非空	用户类型
password	VARCHAR (255)	非空	用户密码
email	VARCHAR	唯一 非空	用户邮箱

文献表

存储医学文献的基本信息，包括标题、作者、出版时间等

字段	类型	约束条件	描述
doc_id	INT	主键 自动增长	文献ID
title	VARCHAR (255)	非空	文献标题
authors	TEXT	非空	文献作者列表
published_date		非空	发表日期
journal_name	VARCHAR (255)	非空	期刊名称
doi	VARCHAR (100)	唯一	DOI链接
keywords	TEXT	非空	文献关键词
abstract	TEXT	可空	文献摘要
full_text_url	VARCHAR (255)	可空	文献全文链接

文献OCR识别结果表

存储通过OCR技术识别出来的文献内容，包括正文、图像中的诊断信息等。

字段	类型	约束条件	描述
ocr_id	INT	主键 自动增长	OCR识别结果ID
doc_id	INT	外键 关联文献表	文献ID
text_content	TEXT	非空	OCR识别出的文献内容
image_content	TEXT	非空	OCR识别出的图像描述（如诊断结果）

检索日志表

记录用户每次检索的相关信息，以便进行分析与优化

字段	类型	约束条件	描述
log_id	INT	主键 自动增长	日志ID
user_id	INT	外键 关联用户表	用户ID
search_query	TEXT	非空	用户搜索的查询内容
search_time	TIMESTAMP	默认当前时间	检索时间
results_count	INT	非空	返回的文献数目

数据分析表

用于存储后台数据分析结果，用于决策支持和系统优化

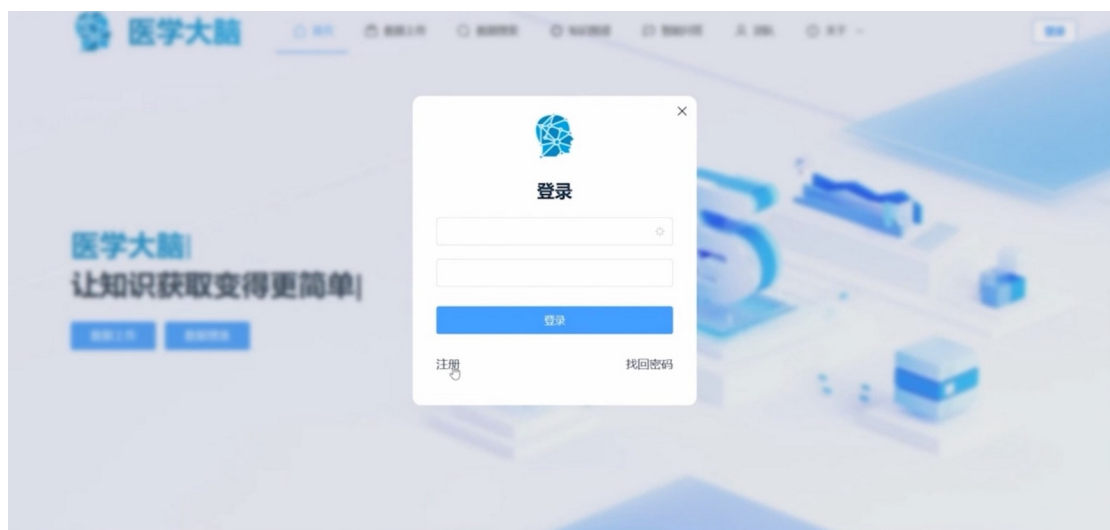
字段	类型	约束条件	描述
analysis_id	INT	主键 自动增长	分析ID
analysis_type	VARCHAR (100)	非空	分析类型 (如关键词趋势、文献分布等)

4 界面设计

4.1 首页



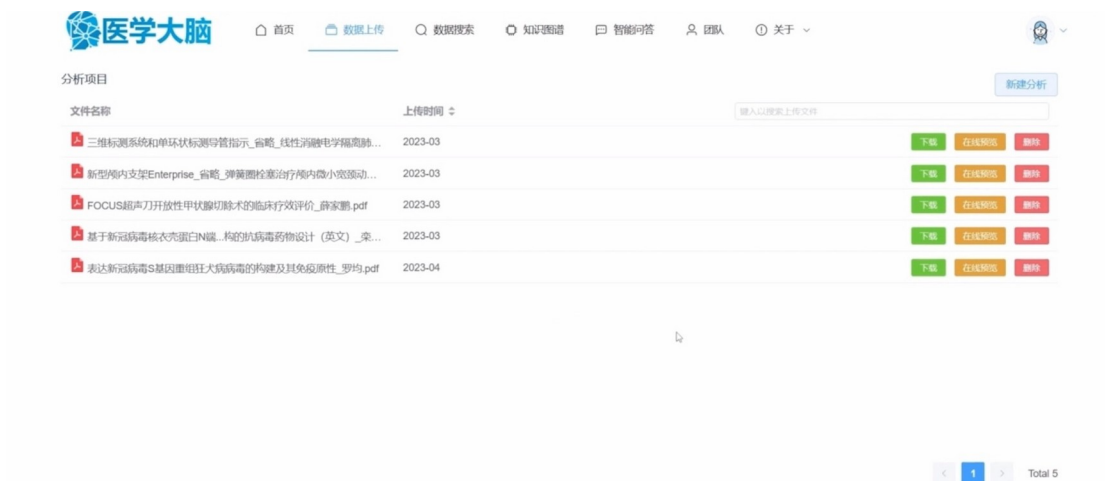
4.2 登录

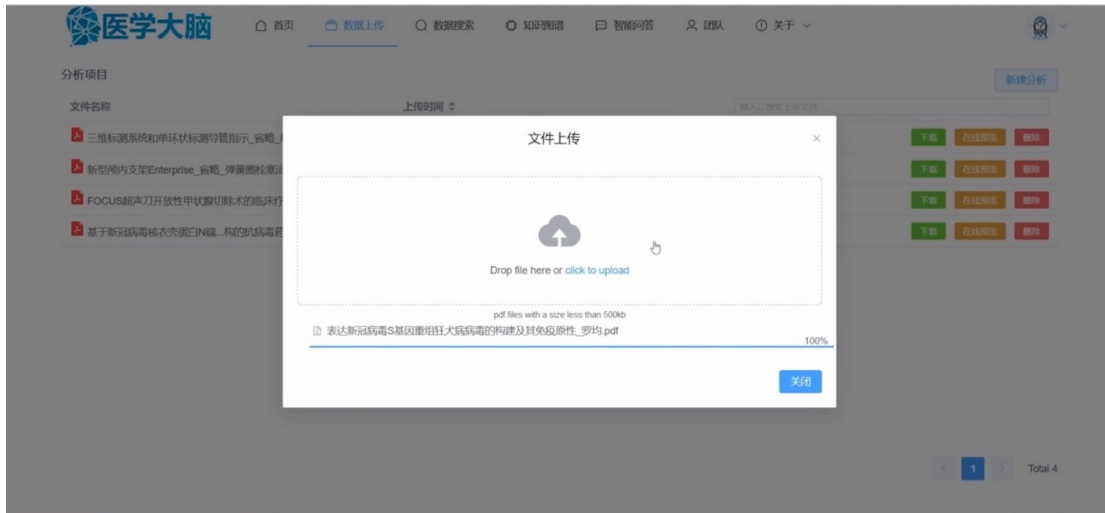


4.2 注册



4.3 pdf格式文献上传

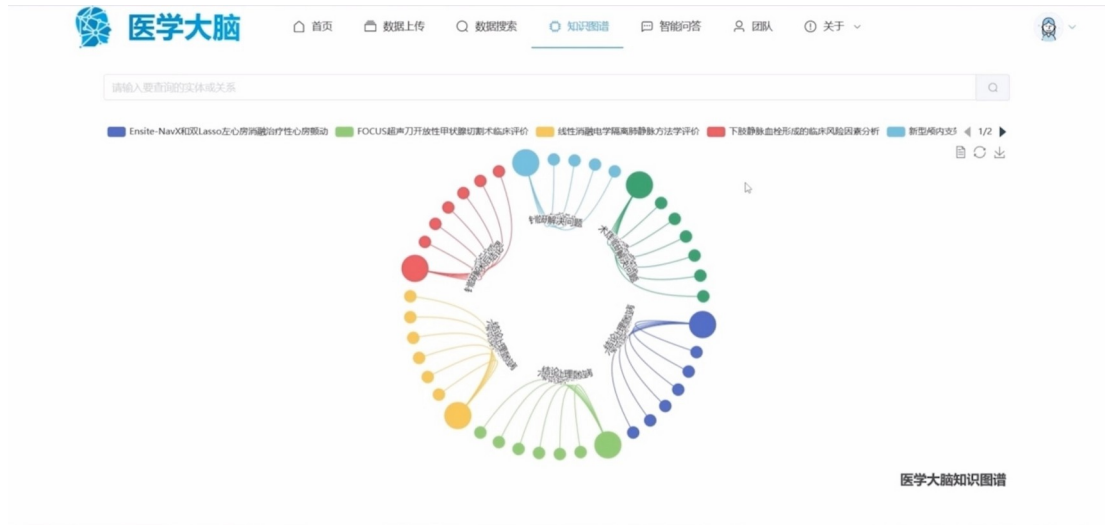




4.4 智能文献检索



4.5 知识图谱



5 Error Design 出错处理设计

1. 状态码为 200 操作成功
2. 状态码为 500 服务器错误
3. 状态码为 400 操作失败
4. 状态码为 401 信息不正确