

律

镜

2023-01-01

过程总结报告

AI + 法律领域垂直搜索引擎

《软件工程管理》G04 小组

组长：徐毕颖

组员：章越、沈书豪、黄文杰、王梓轩



过程总结报告

1. 文档介绍

1.1 引言

本文档是2023年秋冬学期软件工程管理 G04 小组的过程总结报告，旨在用文档的形式实时记录并总结项目开发实况，为开发过程进行可追溯的留档，方便进度管理；并以一定的评估方法及时评估项目工作情况是否与预期相符、是否需要调整开发计划等。此外，本文档还记录了小组成员对每个过程节点完成情况的评价，我们将在此部分灵活地使用软件工程管理中的各类评估方法。

总之，本过程性文档服务于“律镜”软件开发的全过程，监控与评估开发过程，以期帮助开发人员对工程情况快速建立更全面与良好的认识。

1.2 总体内容

本文档的内容应至少包含以下几个部分：

- **里程碑与开发计划表：**即项目计划阶段确定的项目里程碑与依此形成的项目开发计划表，这两份材料来自本项目已编写的项目计划书，为指导项目正式开发而设计。因此我们在本文档中也需要引用这两份材料作为对照，比较实际开发过程与计划是否吻合或作出相应调整。
- **评估标准参考：**对于如何以较为恰当的方式评估，需要提供一定的统一参考标准与评价维度，以避免各组员间因个人标准不同而对项目理解产生偏差。
- **节点式的项目管理纪实：**我们将在本文档中提供一份较为完整的节点式项目管理纪实，记录项目开发过程中的每个关键节点、会议情况及项目计划的调整等，真实展现项目开发过程。这份纪实由所有小组成员共同维护，“节点式”可理解为类似账本式的逐条记录模式（每个节点的详细组织方式见 4.项目管理纪实 部分），方便组员共同添加和更新内容。
- **项目过程总结：**虽然在纪实部分的每个节点中我们也会提供一定量的简短评价，但本文档还应在项目基本完成后，对项目情况（成果如何）与开发过程（整个开发过程是否组织有序、合理）进行总结与评价，并从中吸取经验/教训，实现自我能力的提升。

2. 里程碑&开发计划表

在此处贴出本项目初期设计的项目里程碑（图示）与开发计划表，这两份材料来自本项目已编写的项目计划书，在此作为本文档过程评价的依据和参考。

项目里程碑：

项目里程碑示意图中规定了本软件开发过程中的几个大的关键节点，将软件开发总过程分为了总体规划、需求分析、系统设计、开发（即代码编写）、测试和部署六个大的阶段，这适用于一般的软件开发流程。



项目阶段划分&里程碑示意图

初版项目开发计划表：

项目里程碑提供的阶段划分是非常粗粒度的，因此我们基于项目里程碑的大阶段划分，设计了更为细粒度的项目开发计划表。计划表将项目里程碑图中的大块阶段进一步细分为几个具体任务事项，并规定了预期成果和预期开始与结束时间，方便进度检查与对照。

但是，由于本计划表是在项目正式设计与编写前制定的，可能对项目各部分情况的预期与实际情况略有偏差，我们认为，根据实际情况进行对开发计划进行合理调整是必需的，因此本计划表仅提供参考和对照作用。

事项	预期成果	预期开始时间	预期结束时间
第一阶段：总体规划			
撰写项目计划书	产出项目计划书	2023/10/27	2023/10/30
立项报告展示	完成立项展示ppt	2023/10/28	2023/10/31
第二阶段：需求分析			
市场调研	调研报告	2023/10/31	2023/11/03
用户意见收集	用户意见报告	2023/11/03	2023/11/05
形成需求文档	需求文档	2023/11/06	2023/11/08
明确定义系统功能和特性	功能文档	2023/11/09	2023/11/12
第三阶段：系统设计			

设计系统架构	系统架构设计文档	2023/11/13	2023/11/16
数据库设计	数据库设计文档	2023/11/16	2023/11/18
后端接口设计	API文档	2023/11/18	2023/11/20
前端界面设计	前端设计文档	2023/11/17	2023/11/20
第四阶段：开发			
前端代码编写	前端子系统	2023/11/20	2023/12/13
后端代码编写	后端子系统	2023/11/20	2023/12/13
前后端整合	项目初步模型	2023/12/14	2023/12/16
第五阶段：测试			
单元测试&集成测试	单元测试报告&集成测试报告	2023/12/16	2023/12/17
功能测试	功能测试报告	2023/12/17	2023/12/18
性能测试&安全测试	性能测试报告&安全测试报告	2023/12/18	2023/12/19
用户体验测试	用户体验测试报告	2023/12/19	2023/12/20
第六阶段：部署			
将项目部署到生产环境	部署文档	项目完成优化之后	项目完成优化之后
验收测试&性能优化	验收测试报告	部署上线之后	部署上线之后

3. 评估标准

我们综合查阅了多种软件评估模型或标准，包括 CMMI、ISO 9001 标准、SPICE等，总结出软件开发过程一般可以通过以下评价维度进行评估：

- 质量：**软件开发过程的质量评价维度涵盖了产品质量和过程质量。**产品质量**包括功能完整性、性能、可靠性、可维护性和安全性等方面，这可通过本组开发人员的代码审查与功能测试进行评估；**过程质量**则包括了开发过程中所采用的方法、工具、标准以及质量保证活动等，侧重与开发过程规范性的控制。

2. **效率**：评估软件开发过程的效率包括了资源利用效率、时间效率和成本效率等方面，这包括了开发过程中所需的人力资源、时间成本和财务成本等。在本项目的此项评估中，我们可以综合考虑开发过程中各组员的发展压力是否均衡、工作量分配是否合理等。
3. **创新性**：评价软件开发过程的创新性包括了对新技术、新方法的采用情况，以及对解决问题的创造性思维和方法。我们在开发过程中应及时捕捉创新思路，并在评估后决定是否依此调整现有规划。
4. **过程成熟度**：评估软件开发的过程成熟度需要考虑多个方面，如过程管理能力、过程质量能力、过程改进能力。在这一维度上，我们可善用成熟的评估模型：例如CMMI，它提供了一套结构化的框架，使用了不同的能力级别来评估软件开发过程的成熟度，包括了初始级、已管理级、已定义级、已量化管理级和优化级，方便组织通过对这些能力级别的评估，确定其软件开发过程的成熟度，并识别出改进的方向和重点。
5. **适应性**：软件开发过程的适应性评价维度应考量开发过程对需求变更、现实条件和技术储备的适应能力，例如对于已经指出的问题与开发新需求，本团队能否快速调整并顺利过渡。
6. **沟通与协作**：评价软件开发过程中团队成员之间沟通协作的有效性，包括了信息流畅性、团队合作和决策协调等方面。

4. 项目管理纪实

小组全部成员将共同记录整个项目的实时情况，作为项目管理的一部分。如上文所述，我们将在本部分维护一份节点式的项目管理纪实列表，记录项目中的“里程碑”、每个组内成员负责的任务事项完成情况、开发计划调整等开发过程中各类反映项目过程的动态，这不仅有助于反映项目实况，而且便于在完成开发任务后，根据此节点列表中记录的时间线进行有效的项目复盘。

该份列表注重体现实时性、简洁性。

4.1 节点规范

为保证多人合作完成的项目纪实，能在格式与表达上达到统一性，我们对节点的添加/更新规范做如下规定：


- **节点添加/更新时间：**
 - 每次小组会议后
 - 里程碑记录
 - 完成进度计划表中的某项事务时（相对于上面的计划可以调整）
 - 完成单元性的某个开发任务或者管理任务时
 - 开发计划发生变动时
- **节点关键内容：**
 - 节点创建者

- 创建时间
 - 节点类型：里程碑/会后记录/事项完成/计划调整
 - 具体内容：对节点记录事件的更详细描述，记录清楚即可，原则上不超过100字
 - 技术难点（可选）：对于某个功能开发/实施过程中遇到的问题（难点）、对应的解决方案及心得进行记录
 - 简短评价（总结）：是否按时完成（如果没有的话怎么处理）？完成质量如何？
 - 提供一些证明材料（比如关联的文档链接、git commit 截图、和共同开发者的讨论记录截图等）
- 节点由谁创建：
 - 里程碑/事项完成：完成该事项的人，如相关几个 Commit 的提交者之一
 - 会后记录：由各组员轮流记录
 - 计划变动：提出并实行变动的人

4.2 实时管理记录

10.26 团队会议

 节点创建者：


 徐毕颖


 创建时间：

2023/10/26

 类型：

会后记录

 **具体内容：**作为团队的第一次会议，主要讨论了项目选题、大致功能需求等较为全局性的问题，最终确定了初步以“法律领域垂直搜索引擎”作为本组选题。同时，小组成员确定了项目开发过程中使用的技术栈以及项目管理工具、组内通讯工具、任务完成形式等。

 **总结与评价：**在会议过程中大家以项目本身为主要内容，围绕此主题展开了较为全面的探讨，涉及到之后实际开发中的多个方面。但本次会议中没有对各成员自身的情况（包括知识面、开发能力）进行详细的了解，而这一点对于分工来说是需要。


 记录材料：[📖 10.26 会议记录](#)

10.28 团队会议

 节点创建者：

 创建时间：

 类型：

 沈书豪

2023/10/28

会后记录



具体内容：讨论了软件的具体功能细节与实现思路，并商定了立项展示的具体内容，同时对项目计划书中的部分模块进行了分工以及完善。



总结与评价：在讨论过后明确了软件的功能、创新点等，并对所开发的系统有了更为清晰的总体认识，进一步推动了项目的前期设计进度。

 记录材料：[📖 10.28 会议记录](#)

项目计划书编写完成



节点创建者：

 徐毕颖



创建时间：

2023/10/28



类型：

事项完成



具体内容：通过再次讨论初步确定了本项目的特色以及主要功能的实现次序，在此基础上，小组成员共同完成了项目计划书格式与内容的编写。



总结与评价：在讨论与编写过程中充分调动了各组员对“垂直搜索引擎”这一选题的思考，显著推进了项目进度，沟通协作过程顺利、执行效率高。

 记录材料：[📖 项目计划书](#)

立项完成



节点创建者：

 徐毕颖



创建时间：

2023/11/1



类型：


里程碑



具体内容：小组成员共同讨论立项答辩的重点内容：即本小组项目的特色、重点及采用的项目管理方法；然后由大家共同准备并完成立项答辩的ppt。


同时，完成立项答辩后，项目启动及计划阶段即告一段落，之后就是实际的项目开发阶段，因此此节点作里程碑记。

- ✓ **总结与评价：**顺利完成了小组的立项展示，作为第一组上台展示的小组，尽力体现了本项目特点及开发规划，并回答了老师提出的较多问题（个人感觉回答得不算太坏）；不足之处是由于对展示时间（8分钟）的把握不够精准，导致有一定的超时情况，应在下一次展示时尽可能避免。


 **记录材料：**本小组立项答辩的 ppt（已提交给助教和老师，在此不做展示）

11.12 团队会议

 **节点创建者：**
 文杰 黄文杰

 **创建时间：**
2023/11/12

 **类型：**
会后记录


-  **具体内容：**小组成员讨论了中期展示前要完成的工作（两份中期文档+过程总结报告）、讨论了小组的后续工作模式（如何体现“过程管理”），并对接下来要进行的工作进行了分工总结。

- ✓ **总结与评价：**合理地规划了中期展示以及未来的小组工作，并讨论了后续的管理模式以及过程总结报告的呈现形式，为后来按时有序地完成相应的开发工作以及过程总结报告的撰写工作做好了铺垫。

 **记录材料：** [📄 11.12 会议记录](#)

11.21 团队会议

 **节点创建者：**
 文杰 黄文杰

 **创建时间：**
2023/11/21

 **类型：**
会后记录

-  **具体内容：**小组成员讨论了设计报告和需求规格说明书主要部分的分工以及法律文献数据来源获取的相关问题（爬虫）

✅ **总结与评价：**本次会议的内容较为简单，主要讨论了两篇中期文档的内容划分以及对未来开发计划的进一步设想与细化

📁 记录材料：📄 [11.21 会议记录](#)

11.29 团队会议

😊 **节点创建者：**
👤 文杰 黄文杰

🕒 **创建时间：**
2023/11/29

📌 **类型：**
会后记录

📋 **具体内容：**1. 小组成员商量的中期展示前项目要实现那些功能、展示哪些内容，然后制定了前后端的开发进度任务。2. 检查了两篇中期文档中缺漏和不足的部分

✅ **总结与评价：**在中期展示前期，小组成员有序地开展了开发计划，得益于此次计划，最终在中期展示前很好地实现了前后端整合，产出了一个较为美观和使用的半成品项目。

📁 记录材料：📄 [11.29 会议记录](#)

ElasticSearch 与 Kibana 部署完成

😄 **节点创建者：**
👤 书豪 沈书豪

🕒 **创建时间：**
2023/12/1

📌 **类型：**
事项完成

📋 **具体内容：**在阿里云租用的云服务器上部署了项目所需的 ElasticSearch 服务器以及 Kibana 服务器，并为 ElasticSearch 配置了 ik 中文分词器。

🔑 **技术难点：**ElasticSearch 运行所需的内存较大，需要考虑到服务器的性能，否则会直接崩溃，此外最新版的 ElasticSearch 暂无支持的服务器插件，因此选用了上一版的 8.11.0。



总结与评价：实现了项目所需的底层搜索引擎服务的部署，通过在云服务器上共享ElasticSearch 服务，后端获得了更好的协同开发条件，能够随时测试数据查询相关接口的正确性。



记录材料：Kibana中ElasticSearch页的截图：

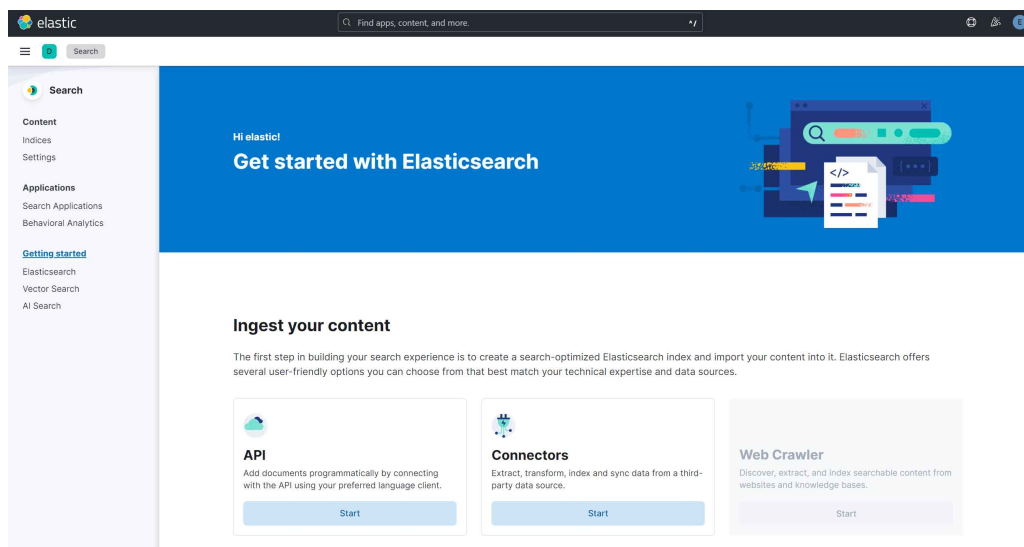


图1

12.3 团队会议



节点创建者：

文杰 黄文杰



创建时间：

2023/12/3



类型：

会后记录



具体内容：小组成员就以下内容进行了讨论：中期答辩准备+进度统计与协调+报告检查与修改



总结与评价：在会议中小组成员充分讨论了中期答辩的准备工作,协调了前后端的进度，制定了下一步计划（包括一些功能的增加与推进以及前后端整合）的ddl，同时还讨论了两份报告需要修改的内容。最后，小组成员统一协商制定了相关任务并进行了分锅（展示、PPT、前后端进度推荐、报告修改）



记录材料： 12.3 会议记录

添加法律法规数据库爬虫



节点创建者：



沈书豪



创建时间：

2023/12/3



类型：

事项完成



具体内容：实现了法律法规数据库爬虫，对国家法律法规数据库进行了自动化的数据爬取。



技术难点：国家法律法规数据库存在复杂的反爬机制，如前端js混淆、challenge code、验证码等，且文档的格式也存在一定的差异，对自动化爬取和信息提取造成了相当的困难。



总结与评价：绕过了国家法律法规数据库全部的反爬机制，实现了全自动、不间断的数据爬取，获取到了包括地方性法规、行政法规、司法解释在内的共计2w+条法律法规数据，为项目提供了早期的数据支持。



记录材料：相关 github 提交记录截图：

feat(crawler): 添加法律法规数据库爬虫

main (#5)

parameterN committed 3 weeks ago

说明：截图补充于12.22

软件需求规格书编写完成（需求确认）



节点创建者：



徐毕颖



创建时间：

2023/12/3



类型：

里程碑



具体内容：小组首先通过讨论明确了本项目要实现的各模块及模块包含的功能，然后合作完成了软件需求规格书的编写，其中重点是各功能需求的使用流程设计。

✅ **总结与评价：**顺利按预期完成了需求分析和需求规格书的编写，在这个过程中，组内各成员都在先前的基础上更加明确了之后要开发哪些内容、要如何开发、开发顺序如何这几个问题，对于项目开发而言是大有裨益的。

📁 记录材料：[📄 软件需求规格书](#)

系统设计报告编写完成（设计完成）

😊 **节点创建者：**
👤 徐毕颖

🕒 **创建时间：**
2023/12/3

📌 **类型：**
里程碑

📄 **具体内容：**小组合作完成了系统设计报告的撰写，明确了系统的整体设计，同时对各部分实现方式做出了大概的构想，并进行了记录。

🔑 **难点：**在系统设计部分，前后端交互的接口设计及数据存储设计是重点，而又因为前者同时涉及到前后端代码的编写而需要考虑多种因素（前后端基于此接口的代码实现难易程度是否均衡、性能是否符合要求），所以写出优质、助力开发的接口设计是不容易的。

✅ **总结与评价：**完成了系统各部分的设计，尽可能呈现了一份能够较好地指导实际开发的系统设计报告，系统设计报告编写完成标志着系统设计阶段结束。

📁 记录材料：[📄 系统设计报告](#)

12.5 中期集成

👍 **节点创建者：**
👤 章越

🕒 **创建时间：**
2023/12/5

📌 **类型：**
里程碑

📄 **具体内容：**前后端集成与联合测试，准备中期答辩。



技术难点：由于前后端独立开发，集成过程中存在许多不确定因素；并且在此之前前后端开发人员并没有相互运行过对方的代码，可能在运行对方程序时遇到依赖包安装问题。



总结与评价：得益于项目前期良好的规划，前后端集成较为顺利，在修改了部分集成过程中发现的 bug 之后，圆满完成了先前制订的中期开发目标，再一次体现了规范的开发流程对开发的助力作用。



记录材料：选取部分相关 github 提交记录截图展示如下（修正集成过程中发现的bug）



基本搜索引擎功能开发完成



节点创建者：



徐毕颖



创建时间：

2023/12/5



类型：

事项完成



具体内容：到目前为止，根据先前分析的需求优先级，已基本实现优先级最高的几个搜索引擎基本功能：输入内容-->跳转到搜索结果页并分页展示搜索结果-->进入详情页查看某一条搜索结果的详细内容（目前由于爬取数据的限制，搜索内容中仅含法律法规）。可以认为已经实现了搜索引擎最基本功能。




技术难点：

1. **前端：**实现搜索功能时，前端请求过程较为复杂，涉及路由更新与传参、触发 swr 请求、页面与组件重新渲染等过程，开发时思路容易混乱。
2. **后端：**需要快速上手及应用 elastic search 与 springboot 的集成，并结合调试结果与目前所得的数据源，制定合适的搜索权重计算。



总结与评价：在团队成员的共同努力下，实现了这些基本功能的开发，可以完成一套流畅的搜索流程，使得项目进度达到了项目计划的预期。

 **记录材料：** 选取部分相关 github 提交记录截图展示如下（按提交时间排列为图1~图4）

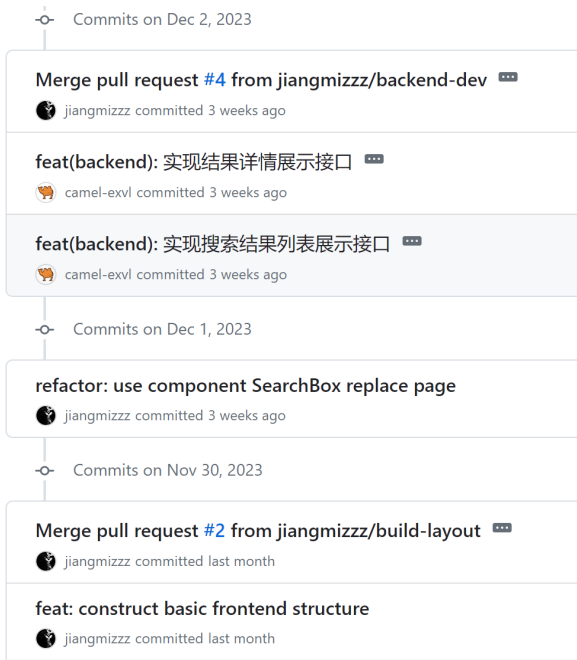


图1

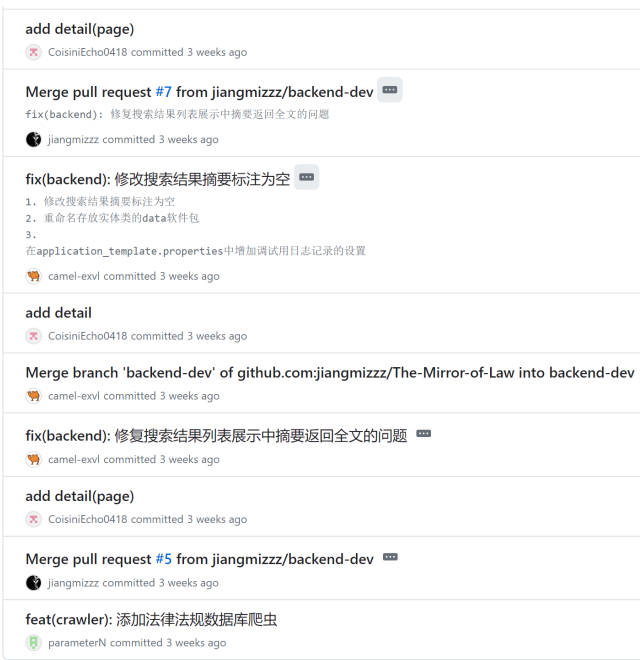


图2

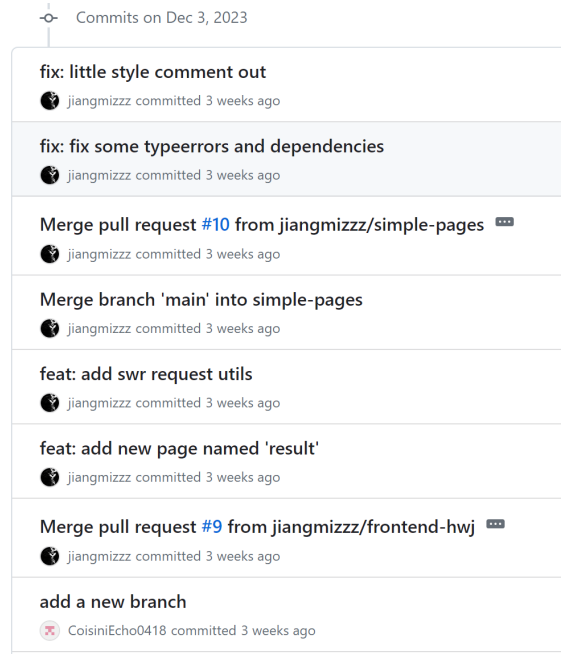


图3

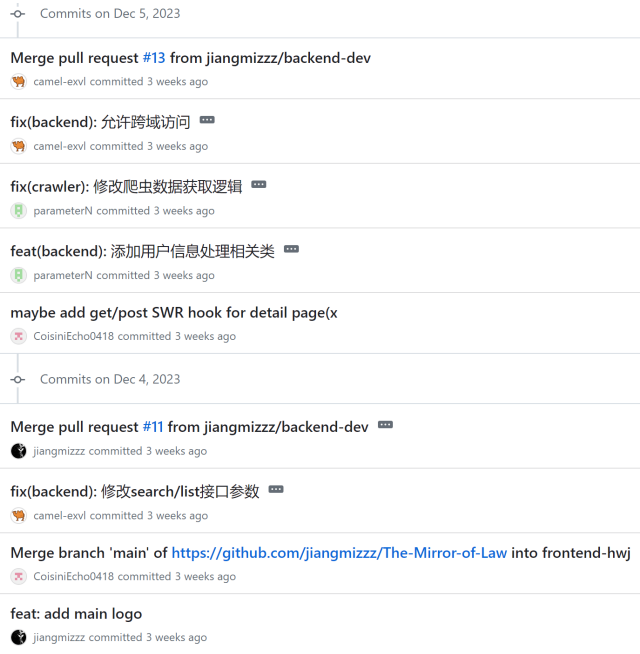






图4

12.16 团队会议

 **节点创建者：**
 **章越**

 **创建时间：**
2023/12/16

 **类型：**
会后记录



具体内容：安排下一步开发进度，进一步完善过程总结报告。



总结与评价：在会议中小组成员充分讨论下一步的开发计划，根据当前进度对计划进行一定的调整，同时进一步规范了过程总结报告格式，完善小组过程管理。



记录材料： [12.16 会议记录](#)

实现AI 接口调用



节点创建者：



沈书豪



创建时间：

2023/12/19



类型：

事项完成



具体内容：完成了对星火AI大模型的接口调用以及相关接口的实现，并设计了AI辅助模块相关功能的prompts。



技术难点：星火大模型的接口对输入文本长度有限制，同时对于部分包含敏感信息的prompts无法作出正确的响应。



总结与评价：确定了系统中AI辅助模块各个功能的可行性，为AI关键词提取、AI文档总结等功能的实现打下了基础。



记录材料：部分相关 github 提交记录截图展示如下



Commits on Dec 19, 2023

feat(backend): 添加AI关键词提取接口 ...

1. 添加连接星火大模型相关依赖
2. 添加星火大模型连接类
3. 添加连接星火大模型所需apiKey等相关配置项
4. 实现AI关键词提取接口，返回结果为以";"分隔的3-5个关键词的字符串
5. 修改druid连接池配置，在后端启动时初始化一个连接，优化数据库访问速度



parameterN committed 5 days ago

用户模块设计修改

 **节点创建者:**
 **章越**

 **创建时间:**
2023/12/20

 **类型:**
计划变动

 **具体内容:** 根据实际开发需要, 修改用户数据表设计和接口设计。

 **技术难点:** 需要充分考虑用户各个字段的结构化存储方式, 关注用户模块增删改查的需求。

 **总结与评价:** 进一步完善了用户模块增删改查的接口设计, 丰富了用户数据表的设计, 为下一步的开发打下基础。

 **记录材料:** 部分此模块相关 github PR记录截图如下

feat(backend): 实现用户登录模块和用户增删改查 #21

Merged

by jiangmizz

main

←

backend-dev

5 days ago

Conversation 2

Commits 1

Checks 0

Files changed 15



camel-exvi commented last week

注意：后端配置文件有修改，新增security配置，测试时无需额外修改内容，直接复制即可。

在开发中针对用户相关API做了如下增改：

用户注册

后端同样会对新用户信息格式进行检查，从安全角度来说仅在前端做检查是没有意义的。
后端对参数的格式限制如下：

- 1. 用户名和密码长度在5-20位，只能包含大小写字母、数字、下划线和减号；
- 2. 邮箱格式会进行正则检查；
- 3. 用户名与邮箱不能重复。

用户登录

参数为 { "email":string, "password":string, "rememberMe":boolean }，新增的 rememberMe 参数可以无视，默认为 true。该参数为真时登录信息将保存7天，否则仅本会话有效。

修改用户信息

API为 PUT /api/user/modify/info，参数为 { "username":string, "email":string }，表示将要修改成的信息，注意若某个参数无需改变则需要传入原始值。（即仅需修改邮箱时传入参数的 username 应为原 username，不能为空）

修改用户密码

API为 PUT /api/user/modify/password，参数为 { "oldPassword":string, "newPassword":string }，表示原密码和新密码，修改成功后会删除cookie退出登录。

删除用户

API为 DELETE /api/user/delete，无参数，删除成功后会删除cookie退出登录。

图1

用户模块基础功能开发完成



节点创建者：

 徐毕颖



创建时间：

2023/12/21



类型：

事项完成



具体内容：根据开发计划，完成用户模块基本功能（用户注册、登录（7天/当前会话有效）、登出、根据 cookie 获取用户信息）的前后端开发。



技术难点：

16

1. **前端**：部分浏览器如 Chrome 等对用户自定义的 cookie 有严格的拦截规则（对于跨域的请求，只允许 `SameSite = Zone` 且使用 https 证书时才能携带 cookie），导致当前开发环境下难以使用 cookie 实现无感知的登录态获取。
2. **后端**：Spring Security 框架结构复杂，需要多个 Filter 相互配合以实现基本的登录登出功能。此外，如何保存用户的登录状态以及根据保存的状态自动登录也是一大难点。



总结与评价：克服了相当的困难，实现了较为严谨安全的用户功能模块，为之后基于用户身份实现的系统功能开发（如用户反馈、搜索历史记录获取、个性化推荐等）打下了牢靠的基础。



记录材料：部分此模块相关 github 提交记录截图如下（按提交时间先后为图1~图3）



图1



图2

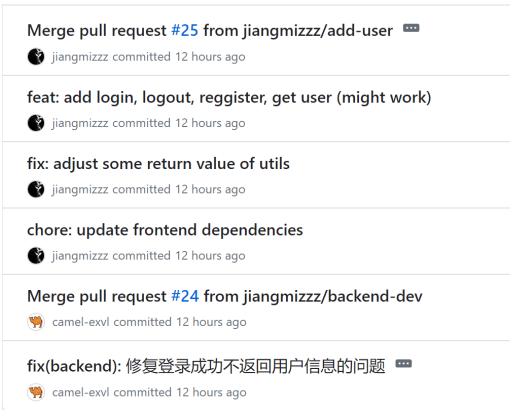


图3

知识图谱设计修改

**节点创建者:**

章越

**创建时间:**

2023/12/22

**类型:**

计划变动

**具体内容:** 根据实际开发需要, 修改知识图谱表格设计。**技术难点:** 使用什么数据结构来表示知识图谱这一复杂网络以及如何在数据库中结构化存储这一信息——原先的节点和边的存储结构在每次寻找子节点时需线性遍历全部的边, 在稠密图中性能不佳。**总结与评价:** 由先前的节点和边存在不同表中改为存储每个节点的邻居, 从而提升在广度优先搜索时的性能。**记录材料:** 部分此模块相关 github 提交记录截图如下。

Commits on Dec 21, 2023

feat(backend): 实现知识图谱获取与新增接口 ...

1. 实现知识图谱获取与新增的接口。
2. 新增知识图谱相关实体类、服务类等, 重新设计知识图谱表结构。
3. 修复用户注册接口返回类型不正确的bug。

camel-exvl committed 3 days ago

fix(backend): 修复知识图谱接口的若干bug ...

1. 修复知识图谱获取接口需要传id的bug。
2. 修复知识图谱新增节点时指定邻居id不存在不报错的bug。

camel-exvl committed 3 days ago

fix(backend): 修复知识图谱接口需要登录才能访问的bug ...

1. 修复知识图谱接口需授权的bug。

camel-exvl committed 3 days ago

图1

AI 辅助模块与词云数据获取接口开发完成

**节点创建者:**

沈书豪

**创建时间:**

2023/12/23

**类型:**

事项完成



具体内容：完成了AI辅助模块原定计划中的全部功能，包括AI关键词提取接口以及AI文档总结接口，并实现了新增的功能词云图数据的获取接口。



技术难点：对于文档进行词频分析，获取词云图所需的数据时，需要去除其中的停用词，防止对生成词云图的质量造成影响。此外AI模块需要考虑到文本过长、敏感信息等特殊情况，并给前端相应的反馈。



总结与评价：为前端界面的词云图渲染、AI助手等功能提供了切实可用的接口，有效推动了相关模块的开发进度。



记录材料：选取部分相关 github 提交记录截图展示如下（按提交时间排列）

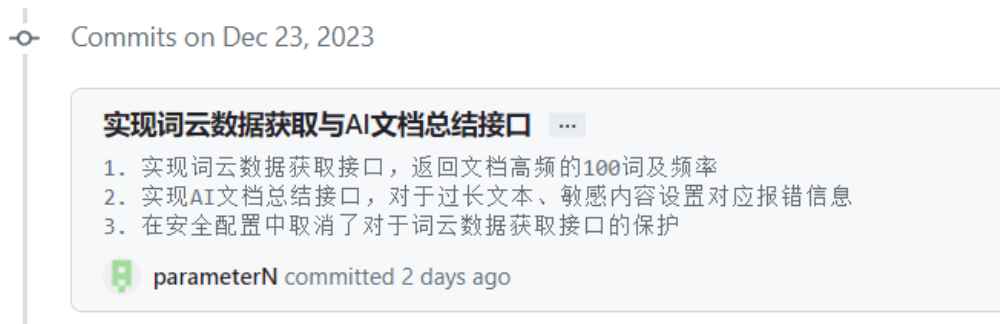


图1

词云模块开发完成



节点创建者：

文杰 黄文杰



创建时间：

2023/12/23



类型：

事项完成



具体内容：在词云图数据结构商定和后端接口实现的此基础上，完成了前端词云模块的开发工作



技术难点：如何正确理解官方文档中的代码示例以及组件相关属性的含义及使用方法，并正常地在前端显示。



总结与评价：参考了CSDN上某篇资料（但是图片遮罩的功能貌似没有实现，后续有待完善）



记录材料：选取部分相关 github 提交记录截图&前端界面截图展示如下

大致实现了词云图、鼠标点击动效和一些请求逻辑并修改了详情页的页面布局 #34

Merged CoisiniEcho0418 merged 1 commit into main from frontend-hwj 3 days ago

Conversation 1 Commits 1 Checks 0 Files changed 17

CoisiniEcho0418 commented 3 days ago

TODO: 1. 试验请求接口 2. 实现鼠标点击特效的关闭逻辑 3. 美化详情页的left-box布局, 使其始终规定在视窗中

be61304

CoisiniEcho0418 commented 3 days ago

增加了新的依赖, 前端启动前需要更新依赖

西藏自治区第九届人民代表大会常务委员会关于强烈谴责达赖集团策划煽动极少数分裂主义分子打砸抢烧的罪恶行径, 坚决维护祖国统一, 反对分裂破坏活动, 促进社会和谐稳定的决议

2008-03-29 来源: 西藏自治区人民代表大会常务委员会

(2008年3月29日西藏自治区第九届人民代表大会常务委员会第二次会议通过)

在举世瞩目的全国两会召开期间, 2008年北京奥运会日益临近之时, 3月10日以来, 在达赖集团精心策划煽动下, 拉萨少数僧人连续聚集闹事, 14日极少数不法分子在拉萨打砸抢烧, 给人民群众生命财产造成极大损失, 严重破坏了社会秩序和社会稳定。这是一起由境内外“藏独”分裂势力策划煽动的严重破坏社会秩序的暴力犯罪事件, 是我们同达赖集团长期尖锐斗争的集中反映, 有着深刻的政治背景和复杂的社会背景。为了坚决维护祖国统一, 反对分裂破坏活动, 促进社会和谐稳定, 特作如下决议:

一、旗帜鲜明地维护祖国统一, 坚决反对分裂破坏活动。西藏是我国神圣领土不可分割的一部分, 是中华人民共和国的一个自治区, 这早已成为国际社会的共识。是世界各国政府公认的历史事实。这次拉萨发生的极少数不法分子打砸抢烧破坏活动, 是达赖集团有组织、有预谋、精心策划的。其险恶用心, 就是企图在敏感时期挑起事端, 蓄意把事情搞大甚至造成流血事件, 干扰北京奥运会, 破坏我区安定和谐的社会政治局面, 最终目的是妄图依靠国际反华势力的支持, 在西藏推翻中国共产党的领导, 颠覆社会主义制度, 否定民族区域自治制度, 把西藏从祖国分裂出去, 恢复最黑暗、最反动、最残酷、最落后的封建农奴制度。这是包括西藏各族群众在内的13亿中国人民绝对不允许的, 是永远不能得逞的, 是注定要失败的。我们同达赖集团的斗争, 不是什么民族、宗教和文化问题, 而是分裂与反分裂的政治斗争。我们一定要深刻认识这场斗争的艰巨性、复杂性和长期性, 在这场斗争中, 在正义与邪恶的较量中, 切实把思想和行动统一到中央关于西藏反分裂斗争的

AI 辅助模块前端界面设计完成



节点创建者：

文杰 黄文杰



创建时间：

2023/12/23



类型：

事项完成



具体内容：完成了 AI 辅助模块前端界面设计原定计划中的全部功能, 利用 ant design 的抽屉组件实现了一个类似于 AI 问答的功能, 同时完成了前端请求的设计并成功与后端接

口实现联调整合。



技术难点：在 ant design 抽屉组件的 body 部分实现 AI 问答界面中的头像+不定长文本的对话 CSS 效果有点困难，具体难点包括1. 头像要位于信息框的左上角，不能因为消息框的拉长而位置变动 2. 用户的消息要和 AI 的消息呈现左右对称的效果 3. 要区分 AI 和用户的消息，并添加了两个不同的消息容器中 4. 实现 AI 消息的获取与答复，并呈现在前端界面中



总结与评价：初步实现了AI辅助模块的大部分功能，不过对于AI加载消息的动态效果有待完善



记录材料：前端界面截图& github 提交记录



Merge pull request #30 from jiangmizzz/frontend-hwj

CoisiniEcho0418 committed 4 days ago

增加了主搜索页和详情页的AI智能助手（但请求方面似乎还有点问题）

CoisiniEcho0418 committed 4 days ago

Merge pull request #36 from jiangmizzz/frontend-hwj

CoisiniEcho0418 committed 2 days ago

修复了AI功能的一点BUG

CoisiniEcho0418 committed 2 days ago

知识图谱模块开发完成



节点创建者:



徐毕颖



创建时间:

2023/12/23



类型:

事项完成



具体内容: 配合完成了知识图谱模块的功能开发, 实现了根据输入词获取相关的知识图谱、短按切换图谱中心节点、长按实现节点值搜索、图谱辐射与收缩、全屏预览等功能。



技术难点: 前端需要学习并运用 ant-design-charts 中提供的辐射树图的操作 API, 以实现对知识图谱的合适呈现及各种附加操作, 过程中需要注意页面渲染问题; 后端需要实现图结构数据的高性能搜索。



总结与评价: 在完成了图谱基本显示功能外又基于现有接口开发了多项附加功能, 大大提升了知识图谱的功能及用户使用体验。



记录材料: 部分此模块相关 github 提交记录截图如下 (按提交时间先后为图1~图3)

Commits on Dec 21, 2023

Merge pull request #29 from jiangmizzz/backend-dev

camel-exvl committed 5 days ago

fix(backend): 修复知识图谱接口需要登录才能访问的bug

camel-exvl committed 5 days ago

fix(backend): 修复知识图谱接口的若干bug

camel-exvl committed 5 days ago

feat(backend): 实现知识图谱获取与新增接口

camel-exvl committed 5 days ago

图1

Commits on Dec 22, 2023

Merge pull request #32 from jiangmizzz/add-user

jiangmizzz committed 4 days ago

feat: build a basic knowledge map

jiangmizzz committed 4 days ago

Merge pull request #31 from jiangmizzz/backend-dev

camel-exvl committed 4 days ago

fix(backend): 获取知识图谱时若无children则不包含该参数

camel-exvl committed 4 days ago

图2

feat: finish knowledge map

jiangmizzz committed 3 days ago

图3

12.23 团队会议



节点创建者:



章越



创建时间:

2023/12/23



类型:

会后记录



具体内容：统计当前开发进度，讨论下一步开发内容和测试报告结构。



总结与评价：项目进度符合预期，组员充分交换意见，讨论了下一步开发中的技术细节，体现良好的沟通。



记录材料： [目 12.23 会议记录](#)

用户反馈和搜索历史功能开发完成



节点创建者：

章越



创建时间：

2023/12/25



类型：

事项完成



具体内容：根据开发计划，完成用户反馈和搜索历史功能的前后端开发。



技术难点：

1. **前端：**前端需要考虑到后端数据更新时间延迟问题，设置一定的数据刷新时间间隔；同时，为避免过于频繁的点击造成的后端数据读写的一致性被破坏，需要在前端对反馈频率进行控制。
2. **后端：**用户反馈模块需要考虑高并发情况下带来的数据不一致问题；搜索历史模块需要考虑对历史记录的去重和重排问题。



总结与评价：经过前后端的通力协作，实现了较为完善的用户反馈和搜索历史功能，进一步提升用户体验。



记录材料：部分此模块相关 github 提交记录截图如下（按提交时间先后为图1~图6）

Commits on Dec 24, 2023

fix: fix the bug that no initialValue when editing info



jiangmizz committed 8 hours ago

feat: add search history




jiangmizz committed 7 hours ago

图2

Commits on Dec 24, 2023

feat(backend): 实现搜索历史与用户反馈接口 ...

1. 优化日志记录等级，修改配置文件，增加调试输出。
2. 修复获取知识图谱接口children可能返回空数组的问题。
3. 修复获取知识图谱接口depth深度过大空循环不退出bug。
4. 搜索结果显示接口返回内容新增score值（前端可忽略此值），并且结果列表将根据score、date排序，为后续搜索类型扩展准备。
5. 实现搜索历史记录功能，搜索时若用户已登录则记录历史。
6. 修改搜索结果展示接口参数，新增type参数表示文档类型，为后续搜索类型扩展做准备。
7. 新增用户反馈接口，与设计时的接口略有不同，会返回整数代表操作状态。
8. 修改获取用户信息接口与登录接口，返回的用户信息内容中新增history、like、dislike。
9. 修改User表设计，实现history、feedbackLike、feedbackDislike的序列化和反序列化。
10. 新增获取个人历史记录接口，返回指定的历史记录数量。

 camel-exvl committed 11 hours ago

feat(backend): 修复用户反馈可以同时点赞点踩的bug ...

1. 修复用户反馈可以同时点赞点踩的bug。


 camel-exvl committed 11 hours ago

图1

Commits on Dec 24, 2023

fix(backend): 修改历史记录为倒序返回，默认值调整为10 ...

1. UserInfo中历史记录改为返回最近10条，且倒序返回。

 camel-exvl committed 7 hours ago

Merge branch 'main' of github.com:jiangmizzz/The-Mirror-of-Law into b... ...

...ackend-dev

 camel-exvl committed 42 minutes ago

fix(backend): 修改若干bug ...

1. 修复搜索历史记录中已有内容后不重新将该内容置于链表尾的bug。
2. 修复用户反馈接口访问过快完整性被破坏的bug。

 camel-exvl committed 30 minutes ago

图5

Merge pull request #50 from jiangmizzz/fix-bug ...

fix: 添加了用户反馈的频率控制和时区校准

 jiangmizzz committed 10 hours ago

fix: control feedback frequency and time offset


 jiangmizzz committed 10 hours ago

图6

Commits on Dec 24, 2023

fix: fix the putFetcher throw Error too fast

 jiangmizzz committed 6 hours ago

fix: history can only be added if logged in


 jiangmizzz committed 6 hours ago

图3

Commits on Dec 24, 2023

feat: add feedback in the results page

 jiangmizzz committed 1 hour ago

图4

数据调整为以单表形式存储



节点创建者：



沈书豪



创建时间：

2023/12/25



类型：

计划变动



具体内容：出于对后端搜索结果优先级排序以及分页相关功能实现的考虑，将法律法规、裁判文书等各类型数据迁移至 ES 中的同一个索引存储，并以额外添加的 type 字段用于文档类型的区分。



技术难点：ES的索引字段一经创建就很难进行修改，必须设计正确的 reindex 指令以及重建索引过程中所需的 painless 脚本，才能做到将不同字段类型的文档正确地迁移至新的索引中，且互相之间不会发生冲突。



总结与评价：通过将不同类型的文档统一储存在 ES 的同一个索引之中，实现了后端文档检索的操作与文档类型之间的解耦，让检索性能更好的 ES 服务器来完成不同类型文档的相关度排序和分页工作而非让 SpringBoot 后端来实现，简化了后端的工作量。



记录材料：Kibana中新的索引数据截图：

mirror-of-law

Overview Documents Index mappings Pipelines

Browse documents

Search documents in this index

< 1 2 3 4 5 ... 400 >

Showing 25 of 10,000. Search results maxed at 10,000 documents.

新疆兴南装饰工程有限公司、乌鲁木齐经济技术开发区卫玲陶瓷经销部买卖合同纠纷民事管辖案件管辖裁定书			①	⇅
📅 date	→	"2022-04-26"		
👍 like	→	0		
📄 docType	→	"裁定书"		
郭某1、沙湾县城给排水公司等财产损害赔偿纠纷民事二审民事判决书			①	⇅
📅 date	→	"2022-04-25"		
👍 like	→	0		
📄 docType	→	"判决书"		
三峡新能源富盛发电有限公司布尔津分公司、布尔津吉峰新能源有限公司等产品责任纠纷民事管辖案件管辖裁定书			①	⇅
📅 date	→	"2022-04-22"		
👍 like	→	0		
📄 docType	→	"裁定书"		

图1

个性化推荐和相关文档推荐模块开发完成



节点创建者：

文杰 黄文杰



创建时间：

2023/12/26



类型：

事项完成



具体内容：

1. 前端：完成了个性化推荐和相关文档推荐的前端开发工作。个性化推荐实现了登录状态下根据用户的历史记录和点赞信息实现个性化推荐，未登录状态采用系统随机推荐（热门推荐），相关文档推荐实现了根据当前文档推荐相关文档的功能。除了展示文档之外，用户点击相应的文档还可以跳转至对应文档的详情页
2. 后端：实现个性化推荐和相关文档推荐的接口，当前端请求相应接口时能够返回对应的文档数据列表



技术难点：

1. 前端：详情页实现点击跳转的时候，不能只采用navigate的方法，因为该方法只会根性浏览器的路径参数，由于父组件的isGetDetail的值没有更新，故不会在向后端发请求获取详情页的数据，所以需要父组件往子组件传递一个用于修改自身state（isGetDetail）的函数作为props，然后子组件在点击处理函数中调用这个函数来修改父组件的state（isGetDetail）来激活父组件重新发请求刷新页面。
2. 后端：个性化推荐模块需要根据用户搜索其历史浏览记录和点赞点踩记录，然后根据其喜好来确定相关推荐的文档。相关文档推荐模块需要根据这篇文档去搜索相关的其他文档并返回相关文档列表（Elastic Search有相应的功能实现）

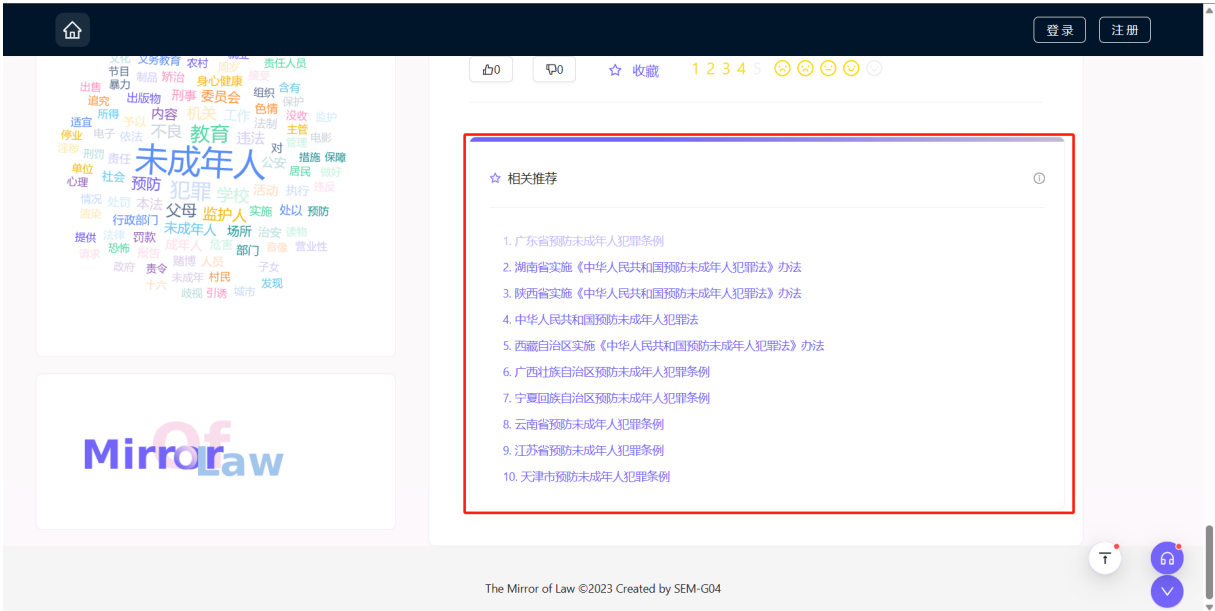


总结与评价：个性化推荐的前后端整合基本完成，功能基本得以完善实现



记录材料：github提交记录截图和前端界面截图：





Merge pull request #60 from jiangmizzz/backend-dev

🐼 camel-exvl committed yesterday

feat(backend): 实现个性化推荐接口 ...

🐼 camel-exvl committed yesterday

Merge pull request #57 from jiangmizzz/backend-dev

🐼 camel-exvl committed 2 days ago

fix(backend): 修复相关文档推荐接口返回null ...

🐼 camel-exvl committed 2 days ago

Merge pull request #51 from jiangmizzz/backend-dev

🐼 camel-exvl committed 3 days ago

feat(backend): 实现相关内容推荐接口 ...

🐼 camel-exvl committed 3 days ago

12.27 团队会议



节点创建者:



徐毕颖



创建时间:

2023/12/27



类型:

会后记录



具体内容: 进行最后一次开发进度统计，确定系统测试报告的工作分配，并且完成末期答辩的准备工作探讨。




总结与评价: 确定了之后需要完成的几点剩余开发工作，并且完成了系统测试报告中最关键部分：测试用例编写的人员分工，基本分工与开发吻合，便于根据实际编写的功能设计合适的测试用例。



记录材料: [12.27 会议记录](#)

项目编码完成

**节点创建者:**


 徐毕颖

**创建时间:**


2023/12/28

**类型:**

里程碑


**具体内容:**

在小组成员的努力合作下，终于顺利完成了本项目的全部代码编写工作，标志着项目编码阶段结束（在编写代码时也完成了代码级的单元测试工作），进入整体测试阶段。

**总结与评价:**

代码编写部分的任务很重，但通过合理的任务分配和 workflows 控制，最终在规定时间内以增量式开发的模式完成了全部代码编写工作，体现了较高的项目管理水平。

 **记录材料:** 项目 github 仓库情况总览:

 The-Mirror-of-Law Private

Unwatch 3

Fork 0


Starred 5

main 7 Branches 0 Tags

Go to file

Add file

Code

 CoisiniEcho0418 Merge pull request #71 from jiangmizzz/frontend-hwj

78f29c5 · yesterday 188 Commits

backend	添加用户自定义内容总结接口	3 days ago
crawler	添加裁判文书索引脚本与法律观点爬虫	4 days ago
frontend	修改了个性推荐列表的获取逻辑	yesterday
README.md	Initial commit	last month

About

2023ZJU软件工程管理课程-G04小组项目代码仓库

Readme


Activity


5 stars


3 watching

0 forks


项目测试与修正完成

**节点创建者:**


 徐毕颖

**创建时间:**

2023/12/31

**类型:**

里程碑

**具体内容:**

在全组的努力下，进行了本项目系统测试报告的编写，完成了包括测试用例设计、测试执行和发现的 bug 修复等工作，标志着项目测试阶段结束。

- ✓ **总结与评价：**通过系统测试工作，我们对之前编写的项目代码进行了从功能模块到整体的测试，得益于严密的用例设计，我们在测试过程中发现了几个之前没有注意到的小 bug 并及时进行了修正，保证了项目总体质量，如果条件允许，已经可以部署上线。

 **记录材料：** [📄 系统测试报告](#)

5. 总结

5.1 成果总结

在本次团队开发中，我们的最终成果是完整的“律镜”法律垂直搜索引擎平台。尽管开发人员、时间和精力均受到限制，但我们仍然通过合理的开发安排与个人的努力保质保量地完成了本项目的开发预期（来自立项与需求分析阶段），配备有如下功能模块及功能条目：

- **搜索词条模块**
 - **关键词搜索：**用户可以在搜索框中输入一个或多个关键词或短语，或者一段长文本，用于搜索指定的内容。
 - **搜索类型选择：**用户可以选择搜索的类型，例如全文搜索、标题搜索、文档类型搜索、作者搜索等。
 - **高级搜索：**用户可以对搜索的方式进行一定的限定，如限制时间范围、结果类型等，来获取更精确、符合需求的搜索结果。
- **搜索结果模块**
 - **搜索结果排序：**搜索引擎可以根据相关性对搜索结果进行排序，基于用户的关键词、搜索方式等因素为结果设置不同的权重，将最相关的结果显示在前面。
 - **结果摘要：**对于每个搜索结果，生成一个摘要放在结果条目下，包含相关文档的片段，以便用户快速了解文档内容。
 - **用户反馈：**用户可以对搜索结果进行反馈（点赞点踩等），规定只有登录的用户才能点赞/点踩，防止恶意刷赞/踩。
- **历史搜索模块**
 - **搜索历史显示：**用户通过搜索引擎进行的法律领域的各种查询会被记录下来，形成用户的搜索历史，以便重新访问之前查找的法律信息，省去了重新输入搜索词的步骤。
 - **个性化推荐：**个性化推荐功能通过分析用户的搜索历史、点击行为、喜好等信息，为用户提供定制化的推荐内容。根据用户的兴趣和需求，搜索引擎会推荐相关的法律文档、案例、法规等内容，以满足用户更深层次的信息需求。

○ 详情展示模块

- **法律文档展示：**根据用户选中的搜索结果，系统可以展示具体的法律文档内容，并以合适的字体、排版来显示文档的标题、作者、导语、正文等部分的内容，使文档更易阅读和理解。
- **相关文档推荐：**在文档的最后，搜索引擎会基于该文档的内容与类型，提供相关性较高的其他文档的链接，供用户快速查阅到相关的内容。

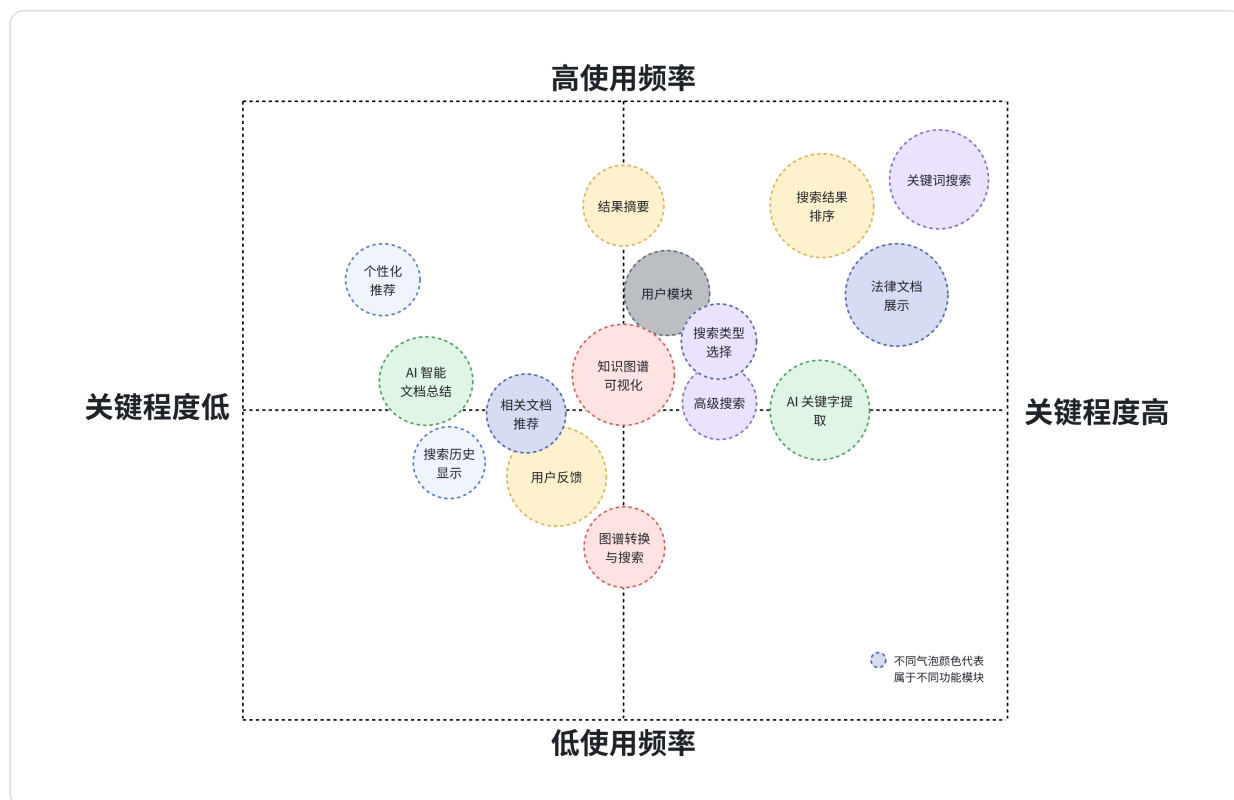
○ AI 辅助模块

- **关键字提取：**通过调用相关 API，对用户输入的查询进行分析，提取其中的关键字，用于优化搜索算法，提高搜索结果的相关性，从而更好地满足用户的信息需求。
- **文档总结：**在阅读文档详情时，可以利用 AI 对搜索结果文档进行智能总结，以提供用户更为精炼和易理解的信息，生成简明扼要的摘要，有助于用户快速了解文档内容，减少阅读时间，提高信息获取效率。

○ 知识图谱模块

- **关键词图生成：**对于任何一次搜索，可以生成它知识图谱形式的关键词图：提取得到它的1-2个关键词，作为图的中心词，然后返回其深度1-2的其他关键词节点，组成一张美观的词图。
- **图谱转换与跳转搜索：**点击词图中的各节点，可以将其转换为中心词（提供动态转换效果），并且用户可以选择是否以该节点上的词为输入值，再次搜索。

我们也可以对照需求分析阶段形成的这张四象限气泡图，图中计划开发的全部功能均按照合理的优先级与顺序完成了实际的开发：



需求分析阶段形成的四象限气泡图

值得一提的是，对照我们的实际功能开发顺序可以发现，**该顺序与气泡图中划定的优先级高低（即从右上到左下）是非常对应的**，这不仅体现了需求分析阶段对各功能需求分析的精准，也说明了需求分析对实际开发具有显著的指导意义。

总之，虽然总的开发工作量非常大，但我们最终成功实现了预期实现的全部功能，经过整体测试可知，开发完成的项目具有优质功能表现、少 bug 的优点，总的来说是一个质量上乘、具有实际应用价值的 web 平台。

5.2 开发经验总结

我们同样通过本次项目开发积累了许多不错的开发经验，尤其是在合作开发方面：

- **促成高效的沟通、清晰的目标**

- 在项目立项、需求分析等前期阶段，就制定出清晰的项目目标和里程碑，便于执行。
- 在每次会议前整理出一份会前准备，确保小组中各成员了解会议内容、提前做好准备以实现高效讨论。
- 目标的表述尽量清晰明确，会更有利于目标的达成。

- **制定合理的分工合作模式**

- 在合作时明确每个成员在团队中的角色（一开始不明晰没关系，一般会随着进度推进而逐渐清晰），每次分工后也需要再次确保每个人都清楚自己之后的责任和目标。
- 有合适的方式记录和展示每位成员的工作量，比如组内 TODO 列表、git 提交和分支合并记录等。

- **进行必要的代码管理和版本控制**

- 使用 Git 来管理团队代码，并在开发前强调规范的组内 Git 使用方式（如不要直接推送到主分支、合并通过 pr 实现、分支保护等），确保团队成员可以顺畅地协作开发，不至于产生不必要的冲突和麻烦。
- 遵循良好的分支管理策略，以便有效地管理功能开发和修复。

- **开发过程中定期审查和反馈**

- 进行定期的代码审查（组内的话可以是同端的开发者之间互相检查代码），以确保代码质量和一致性。
- 在团队会议和日常的通讯群中提供积极的反馈和及时的答疑，互帮互助，改进工作。

- **在线文档编辑和共享**

- 在项目过程中编写清晰的文档，包括API文档、功能需求规定等，以便团队成员可以共同查看并及时对照文档内容进行开发或纠错（对于“纠错”这个需求来说，在线文档非常重要）。
- 使用适当的组内工作沟通工具，如飞书，以促进实时沟通和信息共享。

- **保持开发的灵活性和适应性**

- 对于实际情况的变化和更合理的新需求保持开放的心态，及时调整开发计划和优先级。
- 鼓励团队成员提出想法和改进建议，以促进功能创新和持续改进。

上述的这些开发经验，既是我们在整个开发过程中逐渐认知和理解到的，也帮助我们使整个开发过程变得更顺畅、更少麻烦而更多获得感。我们参阅了 CMMI 中定义的软件开发过程的五个成熟度级别，认为我们组整体开发过程应当已经达到了基本的“**已量化管理级**”，即：

- 过程是量化的，通过量化的技术进行管理和控制。
- 过程的绩效和成本是可见和可度量的。
- 过程的改进是基于量化的数据和分析。

就实际开发过程中的灵活度和改进能力来说，也初步触到了“**优化级**”的门槛，即：

- 过程是持续改进的，以应对组织的目标和竞争的需要。
- 过程改进是根据定量的理解和技术创新。
- 组织寻求不断优化过程，以满足不断变化的需求和机遇。

因此总的来说，是一次高质量的开发与合作，也是一次高质量的项目管理体验。